

Analisis Sentimen Publik Terhadap Deepfake AI Menggunakan Aplikasi X Dengan Metode Support Vector Machine dan Naive Bayes Classifier

Satria Al Afif¹, Ryan Randy Suryono^{2,*}

¹Fakultas Teknik dan Ilmu Komputer, Sistem Informasi, Universitas Teknokrat Indonesia, Bandar Lampung, Indonesia

²Fakultas Teknik dan Ilmu Komputer, Magister Ilmu Komputer, Universitas Teknokrat Indonesia, Bandar Lampung, Indonesia

Email: ¹satria_al-afif@teknokrat.ac.id, ^{2,*}ryan@teknokrat.ac.id

Email Penulis Korespondensi: ryan@teknokrat.ac.id

Submitted: 29/08/2025; Accepted: 20/09/2025; Published: 21/09/2025

Abstrak—Perkembangan teknologi kecerdasan buatan (AI) yang semakin pesat mendorong meningkatnya interaksi masyarakat dengan *platform* berbasis AI, termasuk Deepfake AI. Salah satu tantangan yang muncul adalah bagaimana menilai opini publik secara objektif, khususnya di media sosial yang menjadi wadah utama penyampaian pendapat. Penelitian ini bertujuan untuk membandingkan kinerja dua algoritma *machine learning*, yaitu *Support Vector Machine* (SVM) dan *Naive Bayes* (NB), dalam analisis sentimen publik terhadap Deepfake AI pada media sosial X. Dataset penelitian terdiri dari 7.774 *tweet* yang dikumpulkan pada periode Oktober hingga November 2024. Setelah tahap *preprocessing*, jumlah data yang digunakan sebanyak 5.559 *tweet* dengan tiga kategori sentimen, yaitu positif, negatif, dan netral. Ketidakseimbangan data ditangani menggunakan *Synthetic Minority Over-sampling Technique* (SMOTE), dengan pembagian data 80% untuk pelatihan dan 20% untuk pengujian. Hasil penelitian menunjukkan bahwa sebelum penerapan SMOTE, algoritma SVM memiliki akurasi tertinggi sebesar 71%, sedangkan *Naive Bayes* hanya mencapai 62%. Setelah penerapan SMOTE, kinerja kedua algoritma meningkat, dengan SVM mencapai 77% dan *Naive Bayes* sebesar 68%. Dengan demikian, SVM terbukti sebagai algoritma terbaik dalam penelitian ini, baik sebelum maupun sesudah penerapan SMOTE, dengan hasil yang lebih seimbang pada tiap kelas sentimen. Penelitian ini menunjukkan bahwa pendekatan analisis sentimen berbasis *machine learning* dapat digunakan untuk memahami opini publik terhadap platform AI, sekaligus memberikan masukan berharga bagi pengembang untuk meningkatkan kualitas layanan dan kepercayaan masyarakat.

Kata Kunci: Analisis Sentimen; Deepfake AI; Support Vector Machine (SVM); Naive Bayes; Media Sosial; SMOTE

Abstract—The rapid development of artificial intelligence (AI) technology has driven increased public interaction with AI-based platforms, including Deepfake AI. One of the main challenges that arises is how to objectively assess public opinion, particularly on social media, which serves as a primary medium for expressing opinions. This study aims to compare the performance of two machine learning algorithms, namely Support Vector Machine (SVM) and Naive Bayes (NB), in analyzing public sentiment toward Deepfake AI on the X social media platform. The research dataset consists of 7,774 tweets collected between October and November 2024. After preprocessing, 5,559 tweets were used, categorized into three sentiment classes: positive, negative, and neutral. Data imbalance was addressed using the Synthetic Minority Over-sampling Technique (SMOTE), with 80% of the data allocated for training and 20% for testing. The results show that before applying SMOTE, the SVM algorithm achieved the highest accuracy at 71%, while Naive Bayes only reached 62%. After the application of SMOTE, the performance of both algorithms improved, with SVM achieving 77% accuracy and Naive Bayes reaching 68%. Thus, SVM proved to be the best-performing algorithm in this study, both before and after SMOTE application, delivering more balanced results across sentiment classes. This research demonstrates that sentiment analysis based on machine learning can be utilized to understand public opinion toward AI platforms, while also providing valuable insights for developers to improve service quality and strengthen public trust.

Keywords: Sentiment Analysis; Deepfake AI; Support Vector Machine (SVM); Naive Bayes; Social Media; SMOTE

1. PENDAHULUAN

Perkembangan teknologi kecerdasan buatan (*Artificial Intelligence/AI*) telah membawa perubahan besar dalam berbagai aspek kehidupan, termasuk dalam bidang analisis teks, sistem rekomendasi konten, hingga interaksi berbasis AI. Salah satu platform yang memanfaatkan teknologi ini adalah Deepfake AI, yang dikembangkan untuk membantu pengguna dalam mengelola dan menganalisis data secara otomatis [1]. Platform ini telah dimanfaatkan oleh berbagai kalangan, mulai dari individu, pelaku usaha, hingga institusi pendidikan, guna meningkatkan produktivitas dan kualitas layanan berbasis digital. Namun, penerimaan masyarakat terhadap Deepfake AI masih beragam. Sebagian memberikan opini positif karena kemudahan dan inovasi yang ditawarkan, tetapi sebagian lain menyoroti isu akurasi, privasi, serta etika penggunaan AI [2]. Permasalahan utama yang muncul adalah kurangnya pemahaman mengenai bagaimana opini publik terhadap Deepfake AI terbentuk, serta faktor-faktor yang memengaruhi persepsi tersebut. Tantangan ini semakin kompleks karena terjadi di era digital yang sarat informasi dan sangat dinamis.

Media sosial, khususnya *platform X* (sebelumnya *Twitter*), menjadi salah satu ruang utama bagi masyarakat untuk menyampaikan pendapat, berbagi pengalaman, serta mendiskusikan isu-isu terkait teknologi AI [3][4]. Hingga Oktober 2023, jumlah pengguna aktif X mencapai 666,2 juta di seluruh dunia dan menempatkannya di posisi ke-12 sebagai platform media sosial terbesar secara global. Sementara itu, di Indonesia terdapat sekitar 27,5 juta pengguna X, sehingga menjadikannya sebagai salah satu pasar potensial untuk memantau opini publik terkait teknologi seperti Deepfake AI [5]. Dengan sifatnya yang *real-time*, terbuka, dan interaktif, X menghadirkan sumber data yang berharga untuk memahami pola penyebaran informasi, tren perbincangan, serta sentimen publik terhadap suatu produk atau teknologi [6][7][8]. Akan tetapi, banyaknya informasi yang tersebar di media sosial juga meningkatkan risiko terjadinya bias, misinformasi, maupun disinformasi yang dapat memengaruhi persepsi publik. Oleh karena itu,

dibutuhkan sebuah pendekatan analisis yang tepat untuk memperoleh gambaran yang lebih objektif mengenai kecenderungan sentimen publik terhadap Deepfake AI.

Solusi yang diusulkan dalam penelitian ini adalah menggunakan analisis sentimen berbasis kecerdasan buatan untuk mengidentifikasi pola opini masyarakat terhadap Deepfake AI di media sosial [9]. Analisis ini memanfaatkan dua metode pembelajaran mesin yang umum digunakan, yaitu *Support Vector Machine* (SVM) dan *Naive Bayes Classifier* (NBC). Metode SVM bekerja dengan mencari *hyperplane* optimal yang dapat memisahkan data berdasarkan kelasnya, sehingga sangat cocok untuk menangani data teks berdimensi tinggi. Sementara itu, NBC memanfaatkan prinsip *probabilitas Bayes* dengan asumsi independensi antar fitur, yang menjadikannya cepat dan efisien dalam melakukan klasifikasi teks [10]. Dengan menerapkan kedua metode ini, penelitian diharapkan dapat memberikan pemahaman mendalam mengenai sentimen publik, sekaligus mendukung strategi komunikasi, pengembangan produk, serta mitigasi potensi risiko reputasi [11].

Sejumlah penelitian sebelumnya telah membuktikan relevansi penerapan analisis sentimen dengan metode SVM dan *Naive Bayes*. Penelitian oleh Alharbi & de Doncker (2020) menemukan bahwa SVM unggul dalam menganalisis opini publik di *X* terkait isu kesehatan, terutama dalam menangani dataset besar dengan kompleksitas tinggi. Selanjutnya, Sharma et al. (2021) memanfaatkan kombinasi *Naive Bayes* dan *Random Forest* untuk menganalisis ulasan produk *e-commerce*, dengan hasil bahwa *Naive Bayes* lebih efisien pada dataset kecil hingga menengah. Penelitian lain oleh Jang & Kim (2022) menyoroiti sentimen publik terkait etika AI di media sosial, dan menunjukkan pentingnya mengintegrasikan metode pembelajaran mesin untuk memahami persepsi masyarakat. Li et al. (2022) membandingkan SVM dan *Deep Learning* dalam menganalisis opini publik terhadap kebijakan pemerintah; meskipun *Deep Learning* menunjukkan keunggulan, SVM tetap memberikan hasil yang stabil dan efisien pada dataset tertentu. Terakhir, Nugroho et al. (2023) di Indonesia menerapkan metode *Naive Bayes* untuk menganalisis sentimen publik terhadap layanan digital pemerintah, dan hasilnya menunjukkan tingkat akurasi yang cukup tinggi serta relevan dalam konteks lokal.

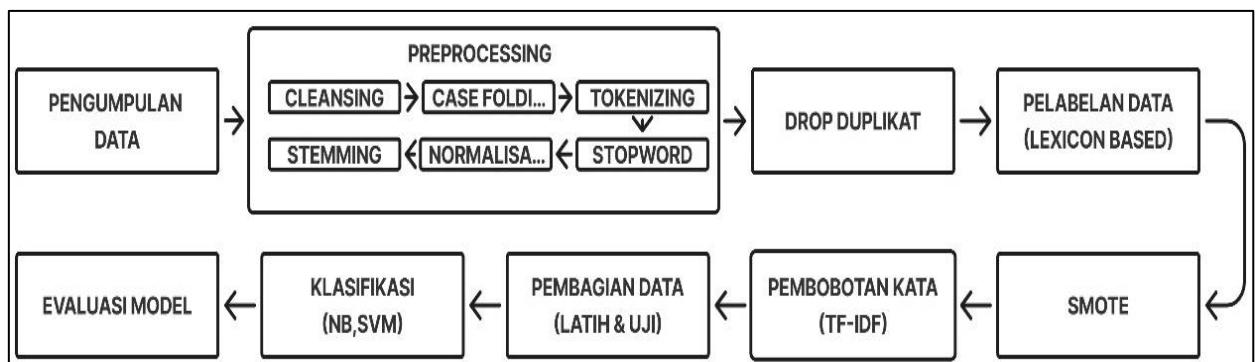
Berdasarkan penelitian-penelitian tersebut, terlihat bahwa baik SVM maupun *Naive Bayes* telah terbukti efektif untuk analisis sentimen di berbagai bidang, seperti isu kesehatan, ulasan produk, kebijakan publik, hingga etika AI [12][13]. Namun, gap penelitian masih jelas terlihat dalam konteks penerapannya pada analisis sentimen publik terhadap Deepfake AI. Hingga saat ini, studi khusus yang membahas perbandingan performa SVM dan *Naive Bayes* dalam menganalisis opini publik terhadap platform AI tertentu, khususnya di Indonesia, masih sangat terbatas. Padahal, memahami persepsi masyarakat dalam konteks lokal penting untuk mendukung pengembangan produk, strategi komunikasi, serta penyusunan kebijakan etika AI yang lebih komprehensif.

Oleh karena itu, penelitian ini bertujuan untuk menganalisis sentimen publik terhadap Deepfake AI di platform *X* menggunakan metode *Support Vector Machine* dan *Naive Bayes Classifier*. Penelitian ini juga akan membandingkan kinerja kedua metode dalam aspek akurasi, presisi, dan recall. Hasil penelitian diharapkan dapat memberikan gambaran yang lebih jelas mengenai persepsi publik terhadap Deepfake AI, menjadi dasar dalam penyusunan strategi komunikasi yang lebih efektif, serta memberikan kontribusi bagi literatur akademik mengenai penerapan analisis sentimen pada teknologi AI [14][15]. Dengan demikian, penelitian ini berupaya tidak hanya menjawab tantangan akademis, tetapi juga memberikan manfaat praktis bagi pengembangan teknologi AI yang lebih etis, transparan, dan sesuai kebutuhan masyarakat.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Penelitian ini dilakukan dengan mengikuti beberapa tahapan yang sistematis untuk memastikan hasil yang valid dan dapat dipertanggungjawabkan. Setiap langkah penelitian dirancang untuk mendalami topik secara menyeluruh dan memberikan pemahaman yang lebih baik mengenai permasalahan yang diangkat. Keseluruhan tahapan penelitian ini divisualisasikan dalam bentuk skema pada Gambar 1.



Gambar 1. Skema Penelitian

Berdasarkan Gambar 1 tahap penelitian dimulai dengan proses pengumpulan data adalah awalan dari penelitian ini, dengan menggunakan metode *crawling* melalui *library Tweet Harvest* di *Google Colab* dengan memanfaatkan Application Programming Interface (API). [16]. Data berupa *tweet* dengan kata kunci tertentu dikumpulkan dalam rentang waktu yang telah ditentukan, selanjutnya data disimpan dalam format CSV (*Comma Separated Values*) untuk diproses lebih lanjut. Tahap berikutnya adalah pra-pemrosesan data elemen yang tidak relevan, akan di bersihkan dan dihilangkan teksnya. Setelah itu, data diberi label sesuai kategori tertentu, seperti sentimen positif, netral dan negatif. Untuk mengatasi ketidakseimbangan data, dilakukan teknik *Synthetic Minority Oversampling Technique* (SMOTE) agar distribusi kelas lebih merata. Selanjutnya, fitur teks diekstraksi menggunakan *Term Frequency-Inverse Document Frequency* (TF-IDF) agar menghasilkan gambaran dalam bentuk angka. Dataset kemudian dipisahkan antara data latih dan data uji untuk membangun model klasifikasi dengan algoritma seperti *Support Vector Machine* dan *Naive Bayes Classifier*. Terakhir, performa model dievaluasi menggunakan metrik seperti *accuracy*, *precision*, *recall*, dan *F1-score* untuk mengevaluasi kinerja model dalam pengklasifikasian data yang tepat.

2.2 Pengumpulan Data

Proses pengumpulan data menggunakan metode *crawling*, yaitu pengambilan informasi dari situs web secara otomatis dengan bantuan perangkat lunak. Data dikumpulkan menggunakan Teknik *crawling* melalui *library tweet-harvest* dari *Node.js*[17] Salah satu kelebihan *Tweet-Harvest* adalah kemampuannya untuk mengumpulkan data dalam jumlah besar. Proses ini dapat dijalankan melalui *Command Line Interface* (CLI) dengan hanya memanfaatkan *auth_token*. [18] Data dikumpulkan dalam *tweet* berbahasa Indonesia dengan kata kunci "Deepfake AI". Sebanyak sebanyak 7.000 *tweet* berhasil dikumpulkandari periode juli 2025 hingga Agustus 2025.

2.3 Preprocessing

Preprocessing data merupakan langkah awal dalam pengolahan data yang bertujuan untuk mempermudah analisis dan pemrosesan lebih lanjut. Proses ini mencakup pembersihan data dari gangguan, penyederhanaan dimensi data, serta penyusunan data agar lebih terstruktur [19]. Tahap *preprocessing* penting untuk meningkatkan kualitas hasil analisis. Proses ini mencakup pembersihan data, normalisasi teks melalui *case folding*, pemecahan teks menjadi token (*tokenizing*), penghapusan kata tidak penting (*stopwords removal*), serta reduksi kata ke bentuk dasar (*stemming*). Berikut Penjelasannya :

- a. Pembersihan Data (*Data Cleaning*) dilakukan untuk menghapus elemen yang tidak relevan dalam teks, seperti tanda baca, angka, URL, dan karakter khusus, sehingga data menjadi lebih bersih dan siap untuk diproses lebih lanjut. Sebagai contoh, teks "Halo! Selamat datang di #Event2024. Info lengkap: www.event.com" setelah melalui proses pembersihan akan menjadi "Halo Selamat datang di Event".
- b. Normalisasi Teks (*Case Folding*) bertujuan untuk mengubah seluruh huruf dalam teks menjadi huruf kecil guna menjaga konsistensi dan menghindari perbedaan akibat penggunaan huruf kapital. Sebagai contoh, teks "Baik" dan "baik" tanpa normalisasi dianggap berbeda, tetapi setelah normalisasi keduanya menjadi "baik".
- c. Tokenisasi (*Tokenizing*) merupakan proses pemecahan teks menjadi unit-unit kecil atau token agar dapat diolah lebih mudah dalam analisis lebih lanjut. Sebagai contoh, teks "Saya suka belajar data science" setelah tokenisasi akan menjadi ["Saya", "suka", "belajar", "data", "science"].
- d. Penghapusan Kata Tidak Penting (*Stopwords Removal*) dilakukan dengan menghilangkan kata-kata umum yang tidak memiliki makna signifikan dalam analisis, seperti dan, di, ke, dan yang. Sebagai contoh, teks "Saya sedang belajar di universitas" setelah melalui proses *stopwords removal* akan menjadi ["belajar", "universitas"].
- e. Normalisasi (*normalized*) proses standarisasi kata dengan mengubah huruf menjadi kecil, mengganti kata tidak baku ke bentuk standar, serta menghapus karakter khusus, misalnya teks "Aku lg di stasiun, nunggu kereta. Gpp sih telat dikit, yg pntg aman :)" setelah normalisasi menjadi "aku lagi di stasiun menunggu kereta tidak apa-apa sih telat sedikit yang penting aman".
- f. *Stemming* bertujuan untuk mereduksi kata ke bentuk dasarnya dengan menghilangkan imbuhan (*prefix*, *suffix*, *infix*), sehingga kata-kata yang memiliki arti serupa dapat dikenali sebagai satu entitas yang sama. Sebagai contoh, kata "berlari", "berjalan", dan "menyanyikan" setelah *stemming* akan menjadi ["lari", "jalan", "nyanyi"].

2.4 Drop Duplicate

Drop Duplicate adalah proses menghapus data yang memiliki nilai ganda atau berulang dalam sebuah dataset, sehingga hanya tersisa data unik yang relevan untuk dianalisis. Tahap ini penting dilakukan karena data duplikat dapat menyebabkan distorsi hasil analisis, menurunkan akurasi model, serta memperbesar ukuran data secara tidak efisien. Dengan menerapkan *drop duplicate*, kualitas data menjadi lebih bersih, terstruktur, dan representatif sehingga dapat meningkatkan keandalan hasil penelitian maupun kinerja algoritma analisis data.

2.4 Labeling

Tahap pelabelan menggunakan metode berbasis *lexicon* dilakukan dengan memanfaatkan kamus yang berisi katakata positif dan negatif. Proses ini didasarkan pada skor sentimen: jika skor lebih dari nol, maka sentimen dikategorikan sebagai positif; jika skor kurang nol, sentimen dikategorikan sebagai negatif. Sementara itu, jika skor tidak memenuhi kedua kriteria tersebut, sentimen akan dimasukkan ke dalam kategori netral.[20]

2.5 SMOTE

Proses pelabelan menghasilkan data yang tidak seimbang, sehingga penanganan lebih lanjut dibutuhkan menggunakan metode SMOTE (*Synthetic Minority Over-sampling Technique*). SMOTE adalah teknik pembelajaran mesin yang dirancang untuk mengatasi masalah ketidakseimbangan kelas dalam dataset. Teknik ini bekerja dengan membuat sampel baru dari kelas *minoritas* untuk menyeimbangkan dataset. Dengan menambahkan contoh dari kelas *minoritas*, dataset pelatihan menjadi lebih seimbang, dan data tambahan tersebut digunakan untuk melatih model pengklasifikasi.

2.6 Feature Extraction

TF-IDF adalah metode yang menggabungkan dua konsep, yaitu *term frequency* (TF) dan *inverse document frequency* (IDF). Metode ini digunakan untuk mengukur pentingnya suatu kata dalam sebuah data. TF menggambarkan seberapa sering suatu kata muncul dalam data tertentu, yang menandakan seberapa relevan kata tersebut dalam data tersebut [21]. Secara matematis algoritma ini dapat dituliskan seperti pada persamaan 1,2 dan 3.

$$tf = 0,5 + 0,5 \times \frac{tf}{\max(tf)} \quad (1)$$

$$idf_t = \log\left(\frac{D}{df_t}\right) \quad (2)$$

$$W_{d,t} = tf_{d,t} \times idf_{d,t} \quad (3)$$

Rumus 1,2 dan 3 merupakan perhitungan TF-IDF (*Term Frequency–Inverse Document Frequency*) yang digunakan untuk menentukan bobot suatu term dalam dokumen. Nilai *tf* menggambarkan jumlah kemunculan *term t* pada dokumen ke-*d*, sedangkan *idf* berfungsi mengukur tingkat kepentingan term dengan mempertimbangkan jumlah dokumen (*D*) yang mengandung term tersebut (*df*). Bobot akhir (*W*) dari *term ke-t* pada dokumen ke-*d* diperoleh dari hasil perkalian antara nilai *tf* dan *idf*, sehingga semakin sering suatu *term* muncul dalam dokumen tertentu namun jarang muncul di dokumen lain, maka bobotnya semakin tinggi. Pendekatan ini umum digunakan dalam analisis teks untuk mengidentifikasi kata-kata yang paling representatif dari suatu dokumen.

2.7 Data Splitting

Tahap berikutnya dalam penelitian ini adalah membagi data menjadi dua bagian: data latih (*training data*) dan data uji (*testing data*), dengan pembagian sebesar 80% untuk pelatihan dan 20% untuk pengujian [22]. Data latih digunakan untuk mengembangkan model dengan menyesuaikan parameter agar mampu melakukan prediksi atau klasifikasi secara akurat. Sementara itu, data uji berfungsi untuk mengevaluasi kinerja model terhadap data yang belum pernah diproses sebelumnya. Pendekatan ini bertujuan untuk memastikan model dapat menggeneralisasi data dengan baik, menghindari *overfitting*, serta tetap efektif dalam menghadapi data baru.

2.8 Classification Model Support Vector Mechine

Support Vector Machine (SVM) adalah algoritma machine learning yang memanfaatkan hyperplane untuk memisahkan area kelas pada data. *Hyperplane* berfungsi sebagai pemisah antara kelas-kelas tersebut. SVM memprediksi kelas sebuah data dengan memberi label berdasarkan area kelas di mana data tersebut berada [23]. Rumus perhitungannya adalah sebagai berikut disajikan pada persamaan .4,5 dan 6.

$$\{(x_i, y_i)\}_{i=1}^N \quad (4)$$

Menghitung nilai *w* dan *b*:

$$w = \sum_{i=1}^N a_i, y_i, x_i \quad b = -\frac{1}{2}(w \cdot x^+ + w \cdot x^-) \quad (5)$$

Fungsi Keputusan klasifikasi *sign* (*f*(*x*))

$$f(x) = w \cdot x + b \quad \text{atau} \quad f(x) = \sum_{i=1}^m a_i, y_i K(x, x_i) + b \quad (6)$$

N adalah jumlah data dalam dataset, *n* adalah jumlah fitur, dan *m* adalah jumlah support vector dengan $a_i > x$, yang menentukan margin optimal. $K(x, x_i)$ adalah fungsi kernel yang menghitung kedekatan titik data dalam ruang fitur lebih tinggi, memungkinkan SVM bekerja efektif tanpa menghitung koordinat eksplisit.

2.9 Classification Naive Bayes Classifier

Naive Bayes, yang didasarkan pada asumsi kuat tentang *independensi* setiap fitur atau kata dalam kelas yang diberikan. Algoritma *Naive Bayes* bertujuan untuk memisahkan data ke dalam kelas-kelas yang spesifik. Kinerja pengklasifikasi dievaluasi berdasarkan akurasi prediksi [24]. Algoritme ini memungkinkan sistem untuk secara akurat memahami dan memprediksi sentimen ulasan melalui analisis teks, menawarkan dasar yang kuat untuk menilai kualitas layanan aplikasi dan umpan balik pengguna. Metodologi penelitian melibatkan perincian masalah yang diteliti setepat mungkin (secara matematis, jika memungkinkan) dan menguraikan pendekatan yang diusulkan. Algoritma *Naive Bayes* didasarkan pada *Teorema Bayes* dengan asumsi bahwa setiap fitur bersifat independen. Rumus *Teorema Bayes* dapat ditulis sebagai berikut:

$$P(H|X) = \frac{P(H|X)P(H)}{P(X)} \tag{7}$$

Pada Rumus 2 merupakan sebuah rumus *Teorema Bayes* dimana rumus yang digunakan untuk menghitung *probabilitas* terjadinya suatu peristiwa berdasarkan informasi atau bukti yang ada. Dalam rumus tersebut, $(P(H|X))$ merepresentasikan probabilitas suatu hipotesis (H) benar setelah adanya bukti (X), dikenal sebagai *posterior probability*. Probabilitas ini dihitung dengan mengalikan $(P(X|H))$, yaitu peluang bukti (X) muncul jika hipotesis (H) benar (disebut *likelihood*), dengan $(P(H))$, yaitu probabilitas awal dari hipotesis (H) sebelum melihat bukti (disebut *prior probability*). Hasilnya kemudian dibagi dengan $(P(X))$, probabilitas keseluruhan dari bukti (X) terjadi tanpa memperhatikan hipotesis (H), yang dikenal sebagai *marginal likelihood*. Dengan teorema ini, kita dapat memperbarui keyakinan terhadap suatu hipotesis seiring dengan adanya bukti baru.

2.10 Evaluasi Model

Selanjutnya adalah melakukan evaluasi untuk menilai kinerja model klasifikasi yang digunakan. Evaluasi ini dilakukan menggunakan *confusion matrix*, dengan beberapa metrik utama. Akurasi (*Accuracy*) digunakan untuk mengukur sejauh mana prediksi model sesuai dengan data secara keseluruhan, baik untuk kategori positif maupun negatif. *Precision* membandingkan jumlah prediksi yang benar terhadap total prediksi positif yang dibuat oleh model. *Recall* mengukur seberapa banyak data positif yang berhasil diprediksi dengan benar dibandingkan dengan total data positif yang sebenarnya. Selain itu, F1-Score dihitung sebagai nilai rata-rata harmonis antara *Precision* dan *Recall* untuk memberikan gambaran seimbang mengenai kinerja model [25]. Rumus perhitungannya adalah sebagai berikut disajikan pada persamaan 8,9,10 dan 11.

$$Akurasi = \frac{TP+TN}{TP+TN+FP+FN} \times 100\% \tag{8}$$

$$Presisi = \frac{TP}{TP+FP} \tag{9}$$

$$Recall = \frac{TP}{TP+FN} \tag{10}$$

$$f_1 - score = 2x = \frac{Presisi \times Recall}{Presisi + Recall} \tag{11}$$

3. HASIL DAN PEMBAHASAN

3.1 Data Collection

Data dikumpulkan melalui metode crawling dengan memanfaatkan *library Tweet Harvest* di *Google Colab* dan API Token *Twitter*. API Token digunakan untuk mengautentikasi aplikasi atau skrip yang mengakses data dari *Twitter* melalui API resmi. Hasil pengumpulan data dirangkum pada Tabel 1.

Tabel 1. Hasil Pengumpulan Data

No	Tweet
1	@jessepinkman_71 @nec358 Ya video ini dihasilkan oleh AI. Tanda-tandanya termasuk deformasi tangan wanita (seperti 6 jari atau tangan kanan di lengan kiri) dan cairan tidak wajar dari kucing yang merupakan kesalahan khas AI. Kebiasaan kucing memang nyata tapi ini palsu.
7.000	@royalvin2018 @txtkonoha Tidak video itu palsu. Tidak ada pelatih orca bernama Jessica Radcliffe yang pernah ada dan insiden tersebut adalah hoaks viral kemungkinan dibuat dengan AI. Kejadian serupa pernah terjadi secara nyata seperti pada Dawn Brancheau tahun 2010 tapi ini bukan.

Bedasarkan Tabel 1, yang menunjukkan hasil pengumpulan data, yang diakumulasikan berupa *tweet* berbahasa Indonesia yang memiliki kata kunci "DeeoKafeAi" selama periode 20 Juli 2025 hingga 15 Agustus 2025. Sebanyak 7.000 *tweet* berhasil dikumpulkan dan disimpan dalam format CSV untuk tahap preprocessing selanjutnya.

3.2 Preprocessing

Dari periode Juli 2025 hingga Agustus 2025, berhasil dikumpulkan sebanyak 7.000 *tweet* yang menjadi dasar analisis dalam penelitian ini. Data yang diperoleh dari media sosial tersebut kemudian melalui serangkaian proses preprocessing untuk memastikan kualitasnya sebelum dilakukan analisis lebih lanjut. Tahap preprocessing meliputi beberapa langkah penting, antara lain pembersihan teks dari karakter khusus, tanda baca, URL, serta stopwords yang tidak memiliki kontribusi signifikan terhadap makna kalimat. Selain itu, dilakukan juga proses normalisasi kata, penghapusan duplikasi, serta tokenisasi agar data lebih terstruktur dan siap diolah menggunakan metode analisis sentimen. Proses ini sangat krusial karena data mentah yang diperoleh dari media sosial biasanya masih berantakan, mengandung banyak noise, serta tidak langsung dapat digunakan untuk pemodelan. Setelah melewati seluruh tahapan preprocessing, data menjadi lebih terorganisir, bersih, dan siap dianalisis secara sistematis, sehingga hasil yang

diperoleh dapat lebih akurat dan reliabel. Gambaran hasil akhir dari tahap preprocessing ini dapat dilihat pada Tabel 2 yang menyajikan ringkasan data setelah dilakukan pembersihan dan persiapan.

Tabel 2. Hasil *Preprocessing*

Tahapan	Tweet
<i>Data Tweet</i>	AKPS memaklumkan video dimuat naik oleh pemilik akaun Hermano Kedutaan itu adalah palsu dan dihasilkan melalui teknik Deepfake menggunakan teknologi kecerdasan buatan (AI). #AWANInews #AWANI745 https://t.co/4ZponlXuXN
<i>Data Cleaning</i>	AKPS memaklumkan video dimuat naik oleh pemilik akaun Hermano Kedutaan itu adalah palsu dan dihasilkan melalui teknik Deepfake menggunakan teknologi kecerdasan buatan AI AWANInews AWANI httpstcoZponlXuXN
<i>Case Folding</i>	akps memaklumkan video dimuat naik oleh pemilik akaun hermano kedutaan itu adalah palsu dan dihasilkan melalui teknik Deepfake menggunakan teknologi kecerdasan buatan ai awaninews awani httpstcozponlxun
<i>Normalizex</i>	[akps memaklumkan video dimuat naik oleh pemilik akaun hermano kedutaan itu adalah palsu dan dihasilkan melalui teknik Deepfake menggunakan teknologi kecerdasan buatan ai awaninews awani httpstcozponlxun]
<i>Tokenizing</i>	['akps', 'memaklumkan', 'video', 'dimuat', 'naik', 'oleh', 'pemilik', 'akaun', 'hermano', 'kedutaan', 'itu', 'adalah', 'palsu', 'dan', 'dihasilkan', 'melalui', 'teknik', 'Deepfake', 'menggunakan', 'teknologi', 'kecerdasan', 'buatan', 'ai', 'awaninews', 'awani', 'httpstcozponlxun']
<i>StopWords</i>	['akps', 'memaklumkan', 'video', 'dimuat', 'pemilik', 'akaun', 'hermano', 'kedutaan', 'palsu', 'dihasilkan', 'teknik', 'Deepfake', 'teknologi', 'kecerdasan', 'buatan', 'ai', 'awaninews', 'awani', 'httpstcozponlxun']
<i>Stemming</i>	akps maklum video muat milik akaun hermano duta palsu hasil teknik Deepfake teknologi cerdas buat ai awaninews awan httpstcozponlxun

Berdasarkan Tabel 2, proses *preprocessing* dilakukan untuk menyiapkan teks agar lebih mudah dianalisis. Tahap pertama yaitu Data X, teks masih berupa data mentah yang mengandung *hashtag*, tautan, dan simbol. Pada tahap *Data Cleaning*, elemen-elemen tersebut dihapus sehingga teks lebih sederhana. Selanjutnya *Case Folding* mengubah seluruh huruf menjadi kecil agar konsisten. Setelah itu, *Tokenizing* memecah teks menjadi kata-kata individu, lalu pada tahap *Stopwords Removal* kata-kata umum yang tidak bermakna dihapus. Terakhir, *Stemming* mengubah kata ke bentuk dasar sehingga teks menjadi lebih ringkas, misalnya “dimuat” menjadi “muat” dan “dihasilkan” menjadi “hasil”. Hasil akhir menunjukkan teks bersih dan siap digunakan untuk analisis sentimen lebih lanjut.

3.3 Labeling

Setelah melalui tahap pemrosesan data, sebanyak 5.916 *tweet* berhasil disaring menjadi data yang bersih dan siap untuk dianalisis. Selanjutnya, setiap *tweet* dilabeli menggunakan *lexicon based* berdasarkan analisis sentimen Dapat dilihat pada Tabel 3 berikut untuk hasil pelabelan:

Tabel 3. Hasil Labeling

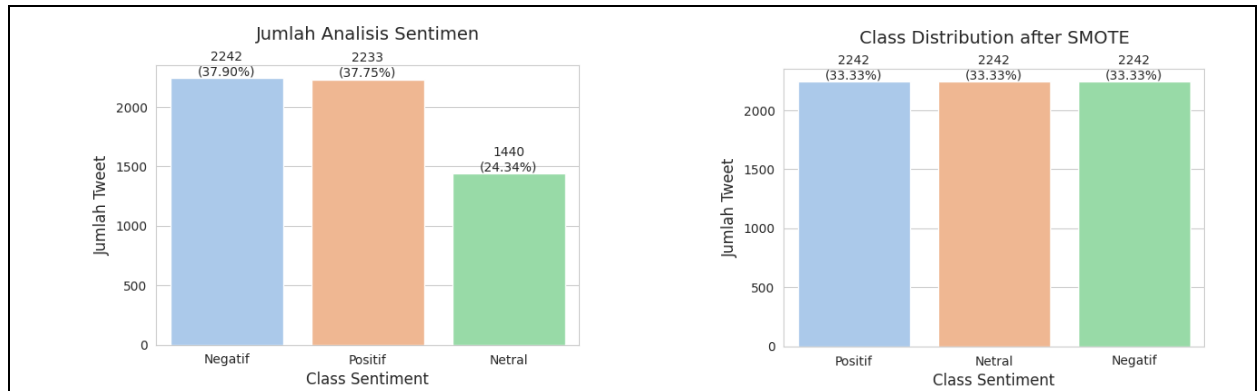
No	Tweet	Sentimen Score	Label
1	haha goda banget serius ai patuh etika hukum surat dokter palsu bikin konsultasi dokter asli sehat bantu	1	Positif
2	video palsu gambar ular raksasa sungai amazon ai sora rekam asli bukti ular alam liar sumber snopes	0	Netral
5.916	ai ubah wajah video palsu potensi fitnah rugi perintah atur tekan risiko httpstcoazzocz	-2	Negatif

Berdasarkan tabel 3, proses pelabelan dilakukan dengan menghitung nilai polaritas dalam rentang -1 hingga 1. Nilai di atas 0 dianggap positif, sementara di bawah 0 dianggap negative sedangkan nilai 0 dianggap netral.

3.4 Handle Imbalance (SMOTE)

Hasil awal analisis sentimen menunjukkan bahwa jumlah data tidak seimbang, dengan 2.233 *tweet* positif, 2.242 *tweet* negatif, dan hanya 1.440 *tweet* netral. Kondisi ini menandakan adanya ketidakseimbangan kelas (*class imbalance*) antara data mayoritas (positif dan negatif) dan data minoritas (netral). Ketidakseimbangan ini berpotensi menyebabkan model cenderung lebih “berpihak” pada kelas mayoritas, sehingga akurasi untuk kelas minoritas menurun. Untuk mengatasi masalah tersebut, digunakan teknik Synthetic Minority Oversampling Technique (SMOTE). Metode ini bekerja dengan cara membuat sampel sintesis baru untuk kelas minoritas, bukan sekadar melakukan duplikasi data yang sudah ada. Proses SMOTE membentuk data baru dengan menghitung jarak antar data minoritas yang ada, kemudian menghasilkan titik data baru di antara mereka. Dengan demikian, distribusi data menjadi lebih seimbang,

sekaligus membantu model pembelajaran mesin mengenali pola pada kelas minoritas secara lebih baik. Hasil SMOTE data dirangkum pada gambar 2.



Gambar 2. Perbandingan sentiment penggunaan Teknik SMOTE

Gambar 2 menunjukkan perbandingan sebelum dan setelah penerapan SMOTE. Sebelum penerapan metode SMOTE, diagram batang memperlihatkan perbedaan signifikan dalam distribusi sentimen, dengan 2.233 *tweet* positif, 2.242 *tweet* negative, dan 1.440 *tweet* netral. Ketidakseimbangan ini dapat menyebabkan bias pada model klasifikasi, di mana model lebih cenderung memprediksi kelas mayoritas, dengan akurasi yang lebih tinggi, namun kurang efektif dalam memprediksi kelas minoritas. Setelah penerapan SMOTE, jumlah *tweet* dalam kedua kategori menjadi seimbang, masing-masing 2.242. Penyeimbangan ini penting untuk meningkatkan akurasi analisis sentimen secara keseluruhan, karena model dapat belajar secara adil dari kedua kelas, mengurangi bias, dan meningkatkan kinerja prediksi pada data minoritas. Penyesuaian ini memastikan bahwa model klasifikasi memberikan hasil yang lebih andal dan representatif.

3.5 Hasil Feature Extraction (TF-IDF)

TF-IDF (*Term Frequency - Inverse Document Frequency*) merupakan metode untuk menilai kepentingan suatu kata dalam sebuah dokumen relatif terhadap kumpulan dokumen lainnya. TF mengukur seberapa sering kata muncul dalam dokumen, sedangkan IDF menunjukkan seberapa jarang kata tersebut muncul di seluruh dokumen, sehingga kata-kata umum memiliki bobot lebih rendah. Teknik ini sering digunakan dalam analisis teks, seperti pencarian informasi dan pemrosesan bahasa alami (NLP). Hasil TF-IDF data dirangkum pada Tabel 4.

Tabel 4. Hasil TF-IDF

Term	TF D1	TF D2	TF D3	IDF	TF-IDF D1	TF-IDF D2	TF-IDF D3
about	0	0	1	1.7	0.0	0.0	0.2
ai	1	1	0	1.3	0.3	0.2	0.0
and	0	0	1	1.7	0.0	0.0	0.2
army	1	0	0	1.7	0.3	0.0	0.0
bagus	1	0	0	1.7	0.3	0.0	0.0
banget	1	0	0	1.7	0.3	0.0	0.0
big	0	0	1	1.7	0.0	0.0	0.2
bilang	1	0	0	1.7	0.3	0.0	0.0
billie	0	0	1	1.7	0.0	0.0	0.4
But	0	0	1	1.7	0.0	0.0	0.2

Tabel 4 menampilkan hasil perhitungan TF-IDF (*Term Frequency - Inverse Document Frequency*) untuk sebuah teks yang dianalisis dari kumpulan *tweet*. Nilai TF menunjukkan seberapa sering suatu kata muncul dalam setiap dokumen (D1, D2, dan D3), sedangkan IDF mengukur tingkat kepentingan kata berdasarkan jumlah dokumen tempat kata tersebut muncul. Hasil perkalian TF dan IDF menghasilkan bobot TF-IDF, yang menunjukkan relevansi suatu kata terhadap dokumen tertentu. Kata dengan nilai TF-IDF tertinggi, seperti *army* (0.330), *bagus* (0.330), *banget* (0.330), dan *bilang* (0.330) pada dokumen D1, dianggap lebih signifikan dalam merepresentasikan isi dokumen tersebut. Sementara itu, kata seperti *ai* (0.251 di D1 dan 0.178 di D2) serta *about* (0.185 di D3) memiliki tingkat kepentingan menengah. Sebaliknya, kata-kata umum yang muncul di lebih banyak dokumen, seperti *and* atau *big*, memiliki bobot TF-IDF yang lebih rendah, sehingga dianggap kurang penting dalam membedakan isi antar dokumen.

3.6 Hasil Klasifikasi Model

Pada tahap ini dilakukan pengujian model klasifikasi dengan menggunakan dua algoritma, yaitu Support Vector Machine (SVM) dan Naive Bayes Classifier (NBC). Kedua algoritma ini dipilih karena memiliki karakteristik yang berbeda dalam proses klasifikasi, sehingga hasil yang diperoleh dapat dibandingkan untuk mengetahui model mana

yang lebih optimal dalam analisis sentimen. Pembagian data dilakukan dengan proporsi 80% untuk data training dan 20% untuk data testing. Skema ini bertujuan agar sebagian besar data digunakan untuk melatih model, sedangkan sisanya digunakan untuk menguji sejauh mana model mampu melakukan prediksi terhadap data baru yang belum pernah dilihat sebelumnya. Tabel 5 dan tabel 6 berikut ini menunjukkan skor *precision*, *accuracy*, *recall*, dan *F1*.

Tabel 5. Perbandingan Klasifikasi algoritma *Support Vector Mechine* sebelum dan Setelah SMOTE

<i>Matriks</i>	Sebelum SMOTE	Setelah SMOTE
<i>Accuracy</i>	71%	77%
Sentimen Negatif		
<i>Precision</i>	70%	74%
<i>Recall</i>	86%	84%
<i>F1-Score</i>	77%	78%
Sentimen Netral		
<i>Precision</i>	72%	87%
<i>Recall</i>	27%	61%
<i>F1-Score</i>	39%	72%
Sentimen Positif		
<i>Precision</i>	72%	74%
<i>Recall</i>	87%	88%
<i>F1-Score</i>	80%	80%

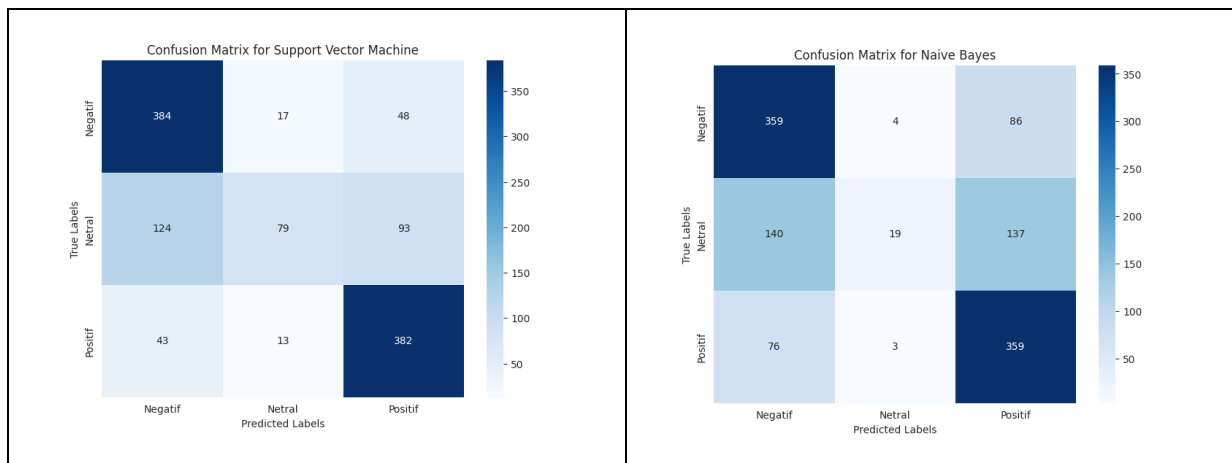
Hasil pengujian menunjukkan bahwa penerapan SMOTE memberikan dampak positif terhadap kinerja model secara keseluruhan, dengan akurasi meningkat dari 71% menjadi 77% yang menandakan model lebih mampu melakukan klasifikasi dengan benar setelah distribusi data diseimbangkan. Pada sentimen negatif, *precision* naik dari 70% menjadi 74% yang menunjukkan prediksi semakin tepat, meskipun *recall* sedikit menurun dari 86% menjadi 84%, namun *F1-score* tetap mengalami peningkatan dari 77% menjadi 78%. Peningkatan paling signifikan terlihat pada sentimen netral, di mana *precision* melonjak dari 72% menjadi 87%, *recall* meningkat drastis dari 27% menjadi 61%, dan *F1-score* naik tajam dari 39% menjadi 72%, yang membuktikan bahwa SMOTE berhasil memperbaiki kelemahan model dalam mengenali kelas minoritas. Sementara pada sentimen positif, *precision* sedikit meningkat dari 72% menjadi 74%, *recall* naik dari 87% menjadi 88%, dan *F1-score* tetap konsisten di angka 80%, menunjukkan performa yang stabil dan andal. Secara keseluruhan, penerapan SMOTE terbukti efektif dalam meningkatkan kualitas klasifikasi sentimen, terutama dalam menyeimbangkan performa antar kelas dan memperbaiki kelemahan pada sentimen netral tanpa mengurangi kinerja pada kelas mayoritas. Sedangkan klasifikasi algoritma *Naïve Bayes* bisa dilihat pada tabel 6.

Tabel 6. Perbandingan Klasifikasi algoritma *Naïve Bayes* sebelum dan Setelah SMOTE

<i>Matriks</i>	Sebelum SMOTE	Setelah SMOTE
<i>Accuracy</i>	62%	68%
Sentimen Negatif		
<i>Precision</i>	62%	66%
<i>Recall</i>	80%	69%
<i>F1-Score</i>	70%	67%
Sentimen Netral		
<i>Precision</i>	73%	71%
<i>Recall</i>	06%	55%
<i>F1-Score</i>	12%	62%
Sentimen Positif		
<i>Precision</i>	62%	66%
<i>Recall</i>	82%	81%
<i>F1-Score</i>	70%	73%

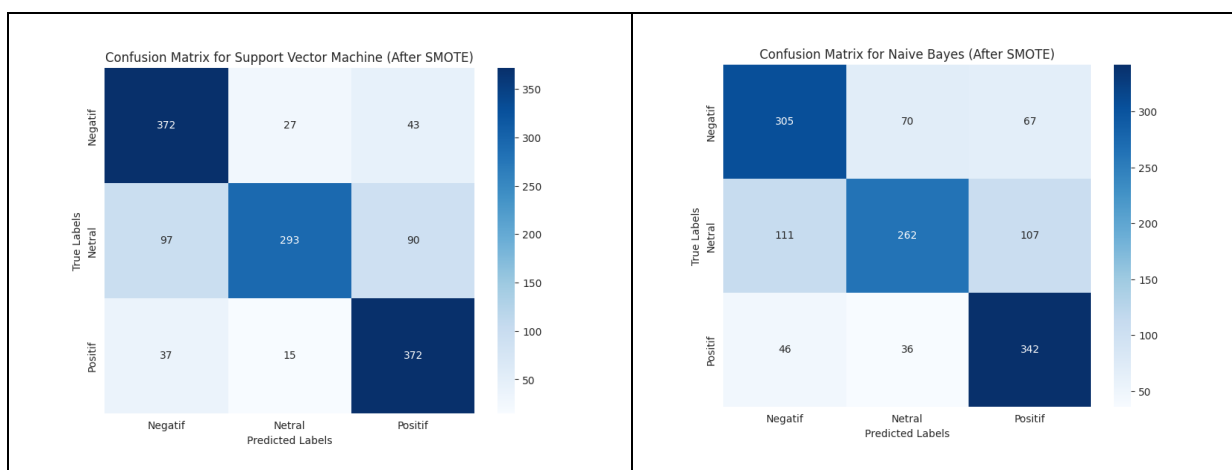
Hasil pengujian menunjukkan bahwa penerapan SMOTE mampu meningkatkan performa model secara keseluruhan, ditandai dengan kenaikan akurasi dari 62% menjadi 68% yang berarti model lebih mampu mengenali pola data setelah distribusi kelas diseimbangkan. Pada sentimen negatif, *precision* meningkat dari 62% menjadi 66%, yang menunjukkan prediksi negatif menjadi lebih tepat, namun *recall* mengalami penurunan dari 80% menjadi 69% sehingga ada sebagian data negatif yang tidak teridentifikasi, dan hal ini menyebabkan *F1-score* ikut turun dari 70% menjadi 67%. Perubahan paling signifikan terlihat pada sentimen netral, di mana *precision* sedikit menurun dari 73% menjadi 71%, tetapi *recall* mengalami lonjakan besar dari hanya 6% menjadi 55%, sehingga *F1-score* juga meningkat tajam dari 12% menjadi 62%; kondisi ini menegaskan bahwa SMOTE sangat membantu model dalam mengenali data netral yang sebelumnya sulit terdeteksi akibat dominasi kelas mayoritas. Sementara itu, pada sentimen positif, performa model relatif stabil dengan peningkatan *precision* dari 62% menjadi 66%, penurunan *recall* yang sangat kecil dari 82% menjadi 81%, serta kenaikan *F1-score* dari 70% menjadi 73%, yang menandakan bahwa model tetap

konsisten dalam mengklasifikasikan data positif meskipun terjadi perubahan distribusi data. Secara keseluruhan, hasil ini menunjukkan bahwa SMOTE efektif dalam memperbaiki kelemahan model terutama pada kelas netral, sekaligus menjaga kinerja pada kelas positif, meskipun terdapat sedikit kompromi pada kelas negatif.



Gambar 3. Confusion Matrix Before SMOTE

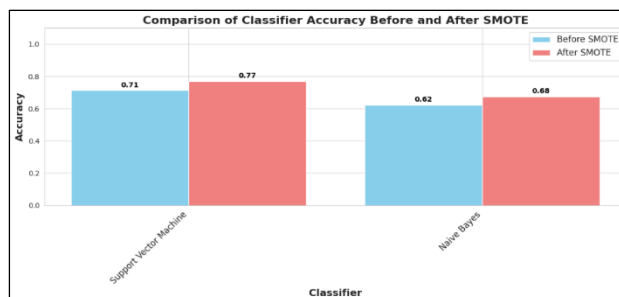
Berdasarkan Gambar 3 yang menampilkan confusion matrix sebelum penerapan SMOTE, terlihat bahwa algoritma SVM memiliki kinerja yang cukup baik dalam mendeteksi kelas mayoritas, khususnya pada kategori negatif dengan jumlah prediksi benar sebanyak 384 dan pada kategori positif sebanyak 382. Namun demikian, SVM masih menunjukkan kelemahan yang cukup mencolok pada kategori netral, di mana banyak data salah diklasifikasikan ke kelas negatif (124 data) maupun ke kelas positif (93 data), sehingga akurasi untuk kelas netral menjadi rendah. Sementara itu, algoritma Naive Bayes Classifier (NBC) juga memperlihatkan performa yang relatif baik pada kelas mayoritas, dengan prediksi benar sebanyak 359 data negatif dan 359 data positif. Akan tetapi, kelemahan NBC dalam mendeteksi data netral jauh lebih besar dibandingkan SVM, dengan kesalahan prediksi yang lebih tinggi yaitu 140 data netral diklasifikasikan sebagai negatif dan 137 data netral diklasifikasikan sebagai positif. Hal ini menandakan bahwa kedua algoritma sama-sama mengalami kesulitan dalam mengenali pola pada kelas netral akibat distribusi data yang tidak seimbang, tetapi SVM menunjukkan tingkat kesalahan yang lebih rendah dibandingkan NBC, sehingga dapat dikatakan memiliki kemampuan yang lebih baik dalam menjaga keseimbangan klasifikasi antar kelas meskipun masih perlu perbaikan, khususnya pada kategori netral.



Gambar 4. Confusion Matrix After SMOTE

Berdasarkan Gambar 4 yang menampilkan confusion matrix setelah penerapan SMOTE, terlihat bahwa algoritma SVM menunjukkan kinerja yang lebih seimbang dengan jumlah prediksi benar sebesar 372 pada kelas negatif, 293 pada kelas netral, dan 372 pada kelas positif; meskipun masih terdapat kesalahan pada kelas netral dengan 97 data salah diprediksi sebagai negatif dan 90 data salah diprediksi sebagai positif, hasil ini jauh lebih baik dibandingkan sebelum SMOTE ketika data netral banyak salah klasifikasi. Sementara itu, algoritma Naive Bayes Classifier (NBC) juga mengalami peningkatan performa dengan menghasilkan 305 prediksi benar pada kelas negatif, 262 pada kelas netral, dan 342 pada kelas positif, namun kesalahan prediksi pada kelas netral masih cukup tinggi, yaitu 111 data diprediksi negatif dan 107 data diprediksi positif, ditambah 70 data negatif yang salah diklasifikasikan sebagai netral, sehingga menunjukkan bahwa meskipun SMOTE membantu memperbaiki kinerja NBC, algoritma ini masih lebih lemah dibandingkan SVM dalam mendeteksi pola kelas minoritas. Secara keseluruhan, penerapan

SMOTE terbukti mampu meningkatkan deteksi kelas netral pada kedua algoritma, namun SVM tetap konsisten menunjukkan performa yang lebih unggul dengan distribusi hasil klasifikasi yang lebih proporsional dibandingkan NBC.

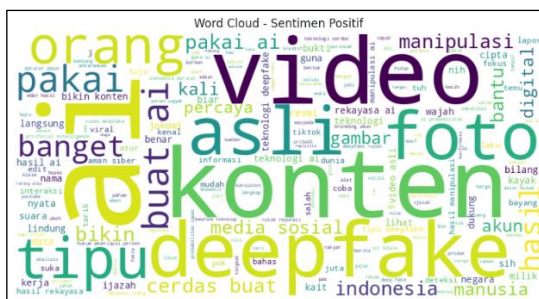


Gambar 5. Perbandingan Klasifikasi

Gambar 5 menunjukkan perbandingan akurasi klasifikasi sebelum dan sesudah penerapan SMOTE, di mana hasilnya konsisten meningkatkan performa pada kedua metode pembelajaran mesin yang digunakan. Pada metode *Support Vector Machine* (SVM), akurasi meningkat dari 71% menjadi 77%, sedangkan pada metode *Naive Bayes* akurasi naik dari 62% menjadi 68%. Hasil ini membuktikan bahwa penerapan SMOTE efektif dalam menyeimbangkan data sehingga model dapat belajar lebih optimal, dengan SVM mencatat akurasi yang lebih tinggi dibandingkan *Naive Bayes* baik sebelum maupun sesudah SMOTE diterapkan.

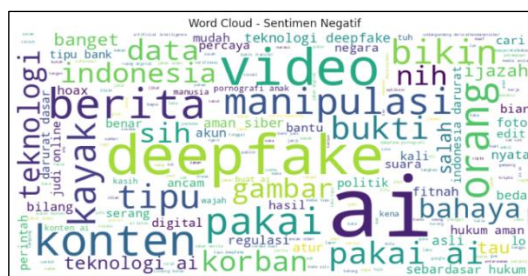
3.7 Hasil Visualisasi WordCloud

Visualisasi WordCloud digunakan untuk menggambarkan kata-kata yang paling sering muncul pada sentimen positif, negatif, dan netral terkait Deepfake AI, di mana ukuran kata mencerminkan frekuensinya. Pada sentimen positif, kata dominan berkaitan dengan manfaat dan harapan, sedangkan pada sentimen negatif lebih banyak menyoroti risiko dan dampak buruk. Sementara itu, pada sentimen netral, kata-kata bersifat deskriptif dan informatif tanpa penilaian emosional. Dengan demikian, WordCloud tidak hanya memvisualisasikan data, tetapi juga memperkuat hasil analisis sentimen dengan menunjukkan kata kunci yang mendominasi tiap kategori. Dapat dilihat pada Gambar 6,7 dan Gambar 8.



Gambar 6. Wordcloud Positif

Gambar 6 menunjukkan *wordcloud* untuk sentimen positif, di mana kata-kata seperti "video," "konten," "foto," "asli," dan "Deepfake" muncul dengan ukuran lebih besar, menandakan bahwa kata-kata ini paling sering digunakan dalam opini positif. Hal ini menunjukkan apresiasi masyarakat terhadap penggunaan teknologi kecerdasan buatan dalam menghasilkan konten yang kreatif, realistis, dan inovatif. Selain itu, kata-kata seperti "orang," "pakai," "buat," dan "cerdas" memperlihatkan pandangan positif mengenai pemanfaatan AI yang dinilai mampu memberikan nilai tambah dalam pembuatan konten digital. Secara keseluruhan, word cloud ini memberikan gambaran bahwa masyarakat menilai teknologi Deepfake dan kecerdasan buatan secara positif, terutama dalam hal kreativitas, orisinalitas, serta peluang pemanfaatannya di dunia digital.



Gambar 7. Wordcloud Negatif



- [4] D. Smith-Mutegi, Y. Mamo, J. Kim, H. Crompton, and M. McConnell, “Perceptions of STEM education and artificial intelligence: a Twitter (X) sentiment analysis,” *Int J STEM Educ*, vol. 12, no. 9, 2025, doi: 10.1186/s40594-025-00527-5.
- [5] Katadata, “pengguna x di indonesia,” Katadata.co.id. [Online]. Available: <https://search.katadata.co.id/search?q=pengguna+x+di+indonesia&source=databoks>
- [6] R. Alsini, “Analysis of Real Time Twitter Sentiments using Deep Learning Models,” *Journal of Applied Data Sciences*, vol. 4, no. 4, pp. 480–489, 2023, doi: 10.47738/jads.v4i4.146.
- [7] Y. Akbar and T. Sugiharto, “Analisis Sentimen Pengguna Twitter di Indonesia Terhadap ChatGPT Menggunakan Algoritma C4.5 dan Naïve Bayes,” *Jurnal Sains dan Teknologi*, vol. 5, no. 1, pp. 115–122, 2023, doi: <https://doi.org/10.55338/saintek.v4i3.1368>.
- [8] Ali Ahmad and W. Gata, “Sentimen Analisis Masyarakat Indonesia di Twitter Terkait Metaverse dengan Algoritma Support Vector Machine,” *Ali aJurnal JTik (Jurnal Teknologi Informasi dan Komunikasi)*, vol. 6, no. 4, pp. 548–555, 2022, doi: <https://doi.org/10.35870/jti.k.v6i4.569>.
- [9] A. Fauziah and Y. Salim, “Analisis Sentimen Pengguna X Terhadap Perkembangan Artificial Intelligence (AI) Menggunakan Algoritma Machine Learning,” *LINER (Literatur Informatika & Komputer)*, vol. 1, no. 4, pp. 347–357, 2024, doi: doi.org/10.33096/linier.v1i4.2534.
- [10] J. Saputra, L. Maryani, D. Wulandari, and W. Eka, “Analisis Performa Naive Bayes dan SVM terhadap Sentimen Teks Media Sosial dengan Word2Vec dan SMOTE,” *Jurnal INSTEK*, vol. 10, no. 1, pp. 143–155, 2025, doi: doi.org/10.24269/mtkind.v17i1.7092
- [11] T. A. Zuraiyah, M. M. Mulyati, and G. H. F. Harahap, “Perbandingan Metode Naïve Bayes, Support Vector Machine Dan Recurrent Neural Network Pada Analisis Sentimen Ulasan Produk E-Commerce,” *Multitek Indonesia*, vol. 17, no. 1, pp. 27–43, 2023, doi: 10.24269/mtkind.v17i1.7092.
- [12] S. Styawati, A. R. Isnain, N. Hendrastuty, and L. Andraini, “Comparison of Support Vector Machine and Naïve Bayes on Twitter Data Sentiment Analysis,” *Jurnal Informatika: Jurnal Pengembangan IT*, vol. 6, no. 1, pp. 56–60, 2021, doi: 10.30591/jpit.v6i1.3245.
- [13] N. Amalia Putri, A. Srirahayu, and N. Arif Sudibyo, “Analisis Sentimen Terhadap Aplikasi KitaLulus Menggunakan Metode Naive Bayes dari Ulasan Google Play Store,” *Smart Comp: Jurnalnya Orang Pintar Komputer*, vol. 14, no. 2, pp. 269–279, 2025, doi: 10.30591/smartcomp.v14i2.7230.
- [14] R. Ferdiana, F. Jatmiko, D. D. Purwanti, A. S. T. Ayu, and W. F. Dicka, “Dataset Indonesia untuk Analisis Sentimen,” *Jurnal Nasional Teknik Elektro dan Teknologi Informasi (JNTETI)*, vol. 8, no. 4, p. 334, 2019, doi: 10.22146/jnteti.v8i4.533.
- [15] V. A. Sulistiani and M. Hamka, “Analisis Sentimen Pengguna Media Sosial Terhadap Identitas Kependudukan Digital Menggunakan Metode Support Vector Machine (SVM),” *Journal of Information System Research (JOSH)*, vol. 5, no. 4, pp. 1312–1322, 2024, doi: 10.47065/josh.v5i4.5211.
- [16] Alisya Mutia Mantika, Agung Triayudi, and Rima Tamara Aldisa, “Sentiment Analysis on Twitter Using Naïve Bayes and Logistic Regression for the 2024 Presidential Election,” *SaNa: Journal of Blockchain, NFTs and Metaverse Technology*, vol. 2, no. 1, pp. 44–55, 2024, doi: 10.58905/sana.v2i1.267.
- [17] S. N. Nugraha, R. Pebrianto, A. Latif, and M. R. Firdaus, “Analisis Sentimen Twitter Terhadap Menteri Indonesia Dengan Algoritma Support Vector Machine Dan Naive Bayes,” *E-Link: Jurnal Teknik Elektro dan Informatika*, vol. 17, no. 1, p. 1, 2022, doi: 10.30587/e-link.v17i1.3965.
- [18] N. V. R. Jhosephin and C. Dewi, “Analisis Sentimen Crawling Data dari Sosial Media X tentang Gaza Menggunakan Metode SVM dan Decision Tree,” *Jurnal Indonesia : Manajemen Informatika dan Komunikasi*, vol. 6, no. 1, pp. 427–437, 2025, doi: 10.35870/jimik.v6i1.1225.
- [19] S. A. Azzahra and A. Wibowo, “Analisis Sentimen Multi-Aspek Berbasis Konversi Ikon Emosi Dengan Algoritme Naïve Bayes Untuk Ulasan Wisata Kuliner Pada Web Tripadvisor,” *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 7, no. 4, pp. 737–744, 2020, doi: 10.25126/jtiik.202071907.
- [20] A. Nadira, N. Y. Setiawan, and W. Purnomo, “Analisis Sentimen Pada Ulasan Aplikasi Mobile Banking Menggunakan Metode Naïve Bayes Dengan Kamus Inset,” *Indexia*, vol. 5, no. 01, p. 35, 2023, doi: 10.30587/indexia.v5i01.5138.
- [21] D. Septiani and I. Isabela, “Analisis Term Frequency Inverse Document Frequency (TF-IDF) Dalam Temu Kembali Informasi Pada Dokumen Teks,” *SINTESIA: Jurnal Sistem dan Teknologi Informasi Indonesia*, vol. 1, no. 2, pp. 81–88, 2022, doi: doi.org/10.23960/jitet.v13i1.5559
- [22] Z. Hidayat and R. Kurniawan. R, “Penerapan Naive Bayes Classifier untuk Prediksi Kelayakan Penerima Bantuan Program Indonesia Pintar (PIP),” *EXPERT: Jurnal Manajemen Sistem Informasi dan Teknologi*, vol. 15, no. 1, p. 22, 2025, doi: 10.36448/expert.v15i1.4326.
- [23] M. R. Raihandika and R. R. Suryono, “Perbandingan Algoritma Random Forest , KNN , SVM Untuk Analisis,” *Jurnal Instek Informatika Sains Dan Teknologi*, vol. 6, no. 4, pp. 2403–2412, 2025, doi: 10.47065/bits.v6i4.6797.
- [24] S. F. Amrillah, D. Krisbiantoro, and A. Prasetyo, “Penerapan Metode K-Nearest Neighbors dan Naïve Bayes pada Analisis Sentimen Pengguna Aplikasi Bstation melalui Platform Playstore,” *Building of Informatics, Technology and Science (BITS)*, vol. 6, no. 3, pp. 1281–1292, 2024, doi: 10.47065/bits.v6i3.5863.
- [25] B. Setiawan, K. Ahmad Baihaqi, E. Nurlaelasari, and H. Hikmayanti Handayani, “Analisis Sentimen Ulasan Aplikasi Identitas Kependudukan Digital Menggunakan Algoritma Logistic Regression dan K-Nearest Neighbor,” *Technology and Science (BITS)*, vol. 6, no. 1, pp. 533–540, 2024, doi: 10.47065/bits.v6i1.5389.