

Sentiment Analysis of President Prabowo's Performance on Twitter (X) with a Comparative Study of SVM, XGBoost, and AdaBoost

Anang Ma'ruf^{1*}, Afril Efan Pajri^{2**}, Putri Liana^{3***}

^{*}Teknik Informatika, Universitas Nahdlatul Ulama Sunan Giri

^{**}Sistem Komputer, Universitas Nahdlatul Ulama Sunan Giri

anangmru@gmail.com¹, afril@unugiri.ac.id², putriliana@unugiri.ac.id³

Article Info

Article history:

Received 2025-12-31

Revised 2026-01-28

Accepted 2026-01-30

Keyword:

*Sentiment Analysis,
Machine Learning,
Prabowo Subianto,
Twitter (X).*

ABSTRACT

This study was conducted to understand how Twitter (X) users respond to President Prabowo's performance through machine learning-based sentiment analysis. Data was collected using a dataset crawling approach, then processed through a series of pre-processing stages such as cleansing, case folding, tokenisation, stopword removal, and stemming before being converted into a numerical representation with TF-IDF. The class imbalance problem was addressed by applying SMOTE so that the model could learn more evenly. Three classification algorithms, SVM, XGBoost, and AdaBoost, were tested with the help of GridSearchCV to obtain the best parameter configuration. The research evaluation showed that the XGBoost algorithm was able to provide the best performance with an accuracy of 0.8443, followed by the SVM algorithm with an RBF kernel, which achieved an accuracy of 0.8135. The AdaBoost algorithm came in third with an accuracy of 0.7868. These findings indicate that the boosting approach, especially XGBoost, is better able to handle complex language patterns and high-dimensional text data characteristics. Overall, this study provides an overview of public opinion trends on social media and can be used as a reference for the development of sentiment analysis models in future research.



This is an open access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.

I. PENDAHULUAN

Banyaknya aktivitas saat ini tidak dapat dicegah oleh media sosial. Media sosial menjadi salah satu alat utama untuk menyebarkan tentang pengetahuan, kritik, dan opini publik di era digital saat ini. Media Sosial memungkinkan pengguna di seluruh dunia untuk mengekspresikan pendapat tentang berbagai isu politik secara real-time [1], [2]. Penelitian ini di fokuskan untuk menganalisis opini masyarakat pengguna Media Sosial Twitter (X) pada kinerja Presiden Prabowo Subianto dalam menjalankan sistem pemerintahan Republik Indonesia dalam skala waktu [3], [4] satu tahun. Banyaknya tantangan kebijakan program pemerintahan yang di jalankan oleh presiden Prabowo Subianto memberikan dampak opini sentimen positif dan negatif pada media sosial terutama pada aplikasi Twiter (X). Di antara berbagai platform, Twitter (X) merupakan ruang digital yang sering dijadikan objek riset, terutama dalam penelitian yang berfokus pada analisis

sentimen dan opini publik [5]. Opinion mining adalah suatu riset yang bertujuan memahami bagaimana seseorang menyampaikan pandangan, penilaian, kritik, maupun ekspresi emosional terhadap suatu topik melalui teks [6].

Masyarakat umum menggunakan internet untuk mencari informasi politik, sosial, dan pemerintah melalui platform daring. Secara khusus platform Twitter (X) memiliki keunggulan signifikan dalam komunikasi publik di Indonesia. Twitter (X) tidak hanya berperan sebagai medium bersosialisasi digital, tetapi juga berfungsi sebagai sekumpulan jaringan sosial yang saling terhubung untuk memperluas penyebaran informasi melalui interaksi pengguna [7]. Dari situs Datarepotal.com, jumlah pengguna Twitter (X) telah mencapai 25,2 juta di seluruh Indonesia [8].

Platform Twitter (X) kerap dimanfaatkan sebagai sumber data untuk riset melalui berbagai unggahan yang berisi opini, tanggapan, atau ekspresi pengguna. Analisis sentimen pada platform ini bertujuan guna mengklasifikasikan pendapat atau

tweet, baik bersifat positif maupun negatif. Hasil dari analisis ini dapat memberikan insight mengenai bagaimana pandangan mayoritas pengguna terhadap topik yang sedang dibahas [9].

Media sosial kini menjadi ruang komunikasi yang sangat dinamis, di mana pemerintah dan masyarakat dapat saling berinteraksi tanpa batas. Platform digital memberikan informasi, membangun keterhubungan, serta mendorong partisipasi warga dalam berbagai isu publik. Karena sifatnya yang cepat dan interaktif, media sosial sering dimanfaatkan sebagai sarana untuk berdiskusi, menyampaikan aspirasi, hingga menjembatani dua arah antara pemerintah dan masyarakat [10]. Dalam penelitian terdahulu [11] yang dilakukan oleh (Nguyen, 2021) “media sosial telah menjadi sangat diperlukan sebagai platform pengaturan agenda (*Agenda Setting*), dan adanya teknologi buatan dan kecerdasan buatan (*Artificial Intelligence*) memungkinkan untuk memperoleh wawasan kritis mengenai persepsi dan sikap individu warga negara, dan menginformasikan proses pembuatan suatu kebijakan”. Dari Penelitian yang sudah dilakukan terkait opini publik atas isu politik. Namun belum terlalu banyak yang meneliti tentang sentimen kinerja presiden atau pemerintahan pada sistem demokrasi negara Indonesia.

Penggunaan Algoritma *Support Vector Machine* (SVM) di harapkan mampu dalam mengklasifikasikan sentimen publik pada platform Twitter (X). Penggunaan Algoritma *Support Vector Machine* (SVM) sudah banyak di terapkan pada penelitian terdahulu terutama pada penelitian Sentimen dan menunjukan performa serta akurasi yang baik. penelitian yang di lakukan Alfatah dan Cahyono, [12] dengan melakukan riset “Sentimen analisis pengguna Twitter terhadap Provider XI Axiata Menggunakan metode *Support Vector Machine*” SVM-Linear memiliki nilai akurasi tertinggi sebesar 83% dan SVM-RBF memiliki nilai akurasi tertinggi kedua yaitu sebesar 80%. Selain itu, [13] riset yang dilakukan oleh Hadi, dkk. “Analisis sentimen masyarakat terkait pembangunan Ibu Kota Negara (IKN) dengan *Support Vector Machine* (SVM) dengan *Radial Basis Function* (RBF)” menghasilkan akurasi pengujian 80% dan akurasi pelatihan 99%, dibandingkan dengan *Logistic Regression* dan *Naïve Bayes* yang memiliki akurasi pengujian sebesar 79% dan akurasi pelatihan masing-masing 88% dan 86%.

Algoritma *Support Vector Machine* (SVM) menunjukan akurasi bagus dalam mengolah data di penelitian sebelumnya. Selain penggunaan metode SVM penelitian ini memungkinkan menggunakan metode XGBoost dan AdaBoost guna perbandingan akurasi yang dihasilkan. Penggunaan dari sisi klasifikasi, model XGBoost menjadi salah satu metode ensemble yang unggul karena mampu memperbaiki kesalahan prediksi secara iteratif, serta dilengkapi regularisasi untuk menghindari *overfitting* [14]. Penelitian yang dilakukan Hendrawan, Utami, dan Hartanto membandingkan kinerja algoritma *Naive Bayes* representasi dan XGBoost TF-IDF dan *Word2Vec* pada ulasan Produk lokal di *e-commerce*. Hasilnya menunjukkan bahwa kombinasi *Word2Vec* dan XGBoost

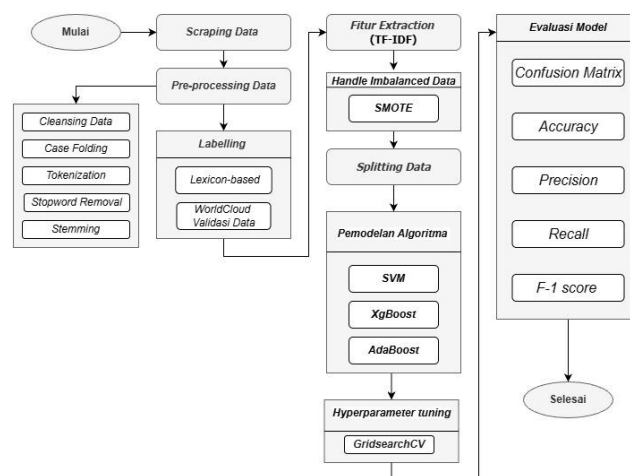
menghasilkan nilai f1-score tertinggi sebesar 0.941, serta lebih efektif dalam menangani data tidak seimbang dibandingkan *Naïve Bayes* [15].

Selain itu Penelitian yang di lakukan Choudhury, dkk. [16] dengan melakukan riset evaluasi kinerja algoritma *machine learning* dalam mendeteksi skenario *Beyond Standard Model* (BSM) di LHC, XgBoost menghasilkan akurasi 92%-97%, menjadikannya model dengan kinerja tertinggi baik dari sisi akurasi maupun nilai AUC, sedangkan AdaBoost menunjukkan akurasi 88%-94% karena komputasi yang lebih ringan serta kestabilannya pada dataset dengan sinyal lemah.

Dengan mengombinasikan ketiga model SVM, XgBoost, AdaBoost, penelitian ini diharapkan dapat meningkatkan performa secara lebih menyeluruh dan objektif. Pendekatan ini juga membantu mengetahui model mana yang paling efektif dalam menganalisa dinamika opini publik terhadap kinerja Presiden Prabowo, sebagaimana tercermin melalui sentimen masyarakat di platform Twitter (X).

II. METODE

Penelitian ini menerapkan Pendekatan *Teks Mining* dalam menganalisis sentimen untuk mengklasifikasikan opini pengguna Twitter (X) terkait kinerja Presiden Prabowo Subianto. Berikut alur proses penelitian riset ditampilkan pada Gambar 1.



Gambar 1. Tahapan Penelitian

Pada gambar 1, merupakan alur dari proses penelitian, dimulai dengan *crawling dataset* melalui proses *scraping tweet* pada aplikasi Twitter (X) menggunakan kata kunci relevan. Data mentah kemudian dibersihkan pada tahap pre-processing, pelabelan data sentimen menggunakan kamus *lexicon-based* dan divalidasi dengan visualisasi *word cloud*, kemudian data dirubah menjadi representasi *numerik* menggunakan pendekatan TF-IDF, untuk mengatasi kelas yang tidak seimbang, diterapkan teknik SMOTE dan *splitting dataset* pada penelitian. Pemodelan dilakukan dengan tiga algoritma, yaitu SVM, XGBoost, dan AdaBoost, dengan pengoptimalan parameter melalui *GridSearchCV*. Tahap

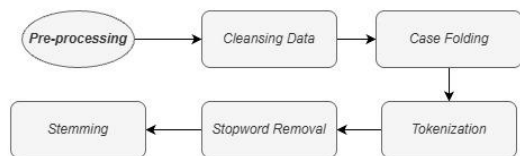
akhir adalah evaluasi dengan *confusion matrix*, akurasi, presisi, recall, dan F1-score guna menentukan algoritma dengan kinerja terbaik dalam memetakan *tweet* publik.

A. Scraping Data

Tahap pertama riset adalah pengumpulan data atau Crawling data dari Twitter (X), data dari web scraping dengan menggunakan API (*Application Programming Interface*) untuk mengunduh *tweet* yang berisi kata kunci spesifik yang terkait dengan kinerja Presiden Prabowo Subianto, seperti “kinerja presiden”, “presiden indonesia”, “Prabowo Subianto”, dan “Presiden Prabowo” dengan dimulai waktu 20-10-2024 hingga waktu 21-10-2025. Pengambilan data dari Twitter (X) terbilang sangat mendukung serta memudahkan, Platform ini juga menyediakan API yang memungkinkan para pengembang mengambil data secara langsung dari sistem Twitter (X). Melalui API tersebut, peneliti dapat mengunduh *tweet*, metadata, maupun informasi pengguna secara terstruktur sehingga proses pengumpulan data menjadi lebih cepat, efisien, dan sesuai dengan kebutuhan penelitian [17]. Data yang didapat dalam proses *scraping* senyak 2.451 data dari platform Twitter (X).

B. Pre-Processing Data

Setelah mendapatkan pengumpulan data melalui scraping, kemudian tahapan *pre-processing*. Tahapan ini bertujuan untuk mengolah dan menganalisis data supaya selama *processing* data dapat lebih optimal hasilnya. Menurut Haddi, dkk. *Pre-processing* bertujuan untuk mengonversi bentuk data yang sebelumnya masih berantakan diubah menjadi terstruktur [18]. Berikut alur tahapan *pre-processing* penelitian ini pada gambar 2.



Gambar 2. Tahapan Pre-processing

1) Cleansing

Pada tahapan Cleansing adalah tahapan penghapusan elemen tidak signifikan seperti tautan (*URL*), penandaan akun (@*user*), emoji, angka, tanda baca, dan karakter khusus yang sangatlah penting dikarenakan dapat menurunkan kualitas analisis dan dapat mempengaruhi hasil penelitian. Tanda baca tertentu umumnya dihapus pada tahap pembersihan teks, terutama yang tidak memberikan makna penting bagi analisis [19].

2) Case folding

Case folding adalah bagian dari prapemrosesan teks yang berfungsi untuk menormalisasi data dengan merubah seluruh karakter huruf pada *tweet* menjadi format huruf kecil

(*lowercase*). Proses ini bertujuan untuk menyeragamkan representasi kata sehingga perbedaan penggunaan huruf kapital tidak memengaruhi hasil seperti “Presiden”, “PRESIDEN”, menjadi kata “presiden”. *Case Folding* adalah proses pengubahan huruf atau penyeragaman kalimat menjadi huruf kecil secara keseluruhan (*lowercase*) [20].

3) Tokenization

Tokenization merupakan proses pemecahan teks menjadi unit-unit kata atau istilah sehingga setiap elemen bahasa dapat diproses secara terstruktur oleh algoritma. *Tokenization* berperan penting karena teks media sosial umumnya bersifat tidak formal, singkat, dan mengandung banyak variasi penulisan. Tahapan ini bertujuan untuk memecah teks menjadi bagian-bagian terkecil berupa kata atau simbol yang dikenal sebagai token, yang kemudian digunakan sebagai fondasi dalam proses pengolahan teks pada tahap berikutnya [21].

4) Stopword removal

Stopword removal merupakan tahap penyaringan teks dengan cara menghapus kata-kata yang kemunculannya sangat sering tetapi tidak memiliki kontribusi signifikan terhadap pemahaman isi, sehingga proses analisis dapat lebih difokuskan pada kata-kata yang benar-benar bermakna, misalnya “yang”, “dan”, “atau”, dan “dengan”. pada penelitian sentimen tahap ini penting karena dapat mengurangi noise serta menurunkan dimensi fitur sehingga model fokus pada kata bermakna yang mempengaruhi polaritas sentimen. Penelitian terdahulu menunjukkan bahwa menghapus stopwords dapat meningkatkan keefektifan model klasifikasi, terutama ketika diterapkan pada teks tidak terstruktur [22].

5) Stemming

Stemming merupakan proses pengubahan kata berimbuhan menjadi bentuk dasarnya, sehingga variasi imbuhan teks seperti “membangun”, “pembangunan”, “dibangun” dapat direduksi menjadi satu bentuk dasar yaitu “bangun”. Pada analisis sentiment, tahap ini penting untuk mengurangi duplikasi fitur dan memastikan bahwa kata dengan makna sama tidak diperlakukan sebagai token berbeda. Studi terdahulu menunjukkan bahwa adaptasi *stemming* berdampak signifikan terhadap performa analisis sentimen, meskipun pengaruhnya dapat bervariasi tergantung tahapan pemrosesan dan jenis data [23].

C. Labeling Dataset

Pelabelan (*labeling*) data dalam analisis sentimen dapat dilakukan secara otomatis menggunakan pendekatan *lexicon-based* atau secara manual melalui anotator. Pendekatan *lexicon-based* menggunakan kamus sentimen Bahasa Indonesia (*InSet*) sering dipilih untuk efisiensi karena dapat memberi label puluhan ribu *tweet* secara cepat berdasarkan skor polaritas kata. Pada penelitian ini pelabelan dataset

memungkinkan menggunakan pendekatan *lexicon-based* [24]. Dataset tersebut kemudian diberikan label menggunakan *class* “positif” dan “negatif” dengan *InSet* kamus bahasa.

1) *Lexicon-based*

Pendekatan *lexicon-based* merupakan metode pelabelan sentimen yang menetapkan polaritas teks dengan mengacu pada skor kata positif dan negatif yang terdapat dalam kamus sentimen. Kamus *lexicon* merupakan daftar kata yang telah dilengkapi dengan nilai atau bobot sentimen tertentu, Sistem *lexicon* dapat menilai kata memberikan kontribusi positif atau negatif terhadap keseluruhan kalimat [25].

2) *WorldCloud Validasi Data*

Word cloud digunakan sebagai teknik visualisasi untuk menampilkan frekuensi kemunculan kata dalam kumpulan *tweet* secara intuitif. Menurut Yusuf, dkk. “Semakin sering sebuah kata muncul, semakin besar ukuran kata tersebut dalam tampilan visual sedangkan kata yang jarang muncul memiliki ukuran huruf yang lebih kecil” [26]. Pada penelitian analisis sentimen, *word cloud* berfungsi sebagai alat bantu eksplorasi awal dalam mengidentifikasi pola umum, topik dominan, serta kata-kata yang mencerminkan kecenderungan sentimen pada data.

D. Feature Extraction (TF-IDF)

TF-IDF (*Term Frequency–Inverse Document Frequency*) adalah teknik pembobotan kata guna membantu mengidentifikasi tingkat penting sebuah istilah dalam dokumen. Metode ini berfungsi untuk merepresentasikan data teks yang awalnya tidak terstruktur ke dalam bentuk numerik atau vektor fitur, sehingga dapat diproses secara optimal oleh algoritma. Pendekatan TF-IDF menjadi pilihan efektif untuk ekstraksi fitur pada dataset sentimen, dan seringkali mengungguli atau sepadan dengan metode *embedding* sederhana dalam analisis data berukuran menengah hingga besar [27].

1) *Term Frequency (TF)*

Term Frequency (TF) merupakan metode yang digunakan untuk mengukur tingkat kemunculan suatu kata dalam sebuah korpus teks. Dalam analisis data teks, nilai TF membantu menilai pentingnya sebuah kata dalam konteks tertentu. Semakin kerap suatu kata muncul, semakin tinggi nilai TF-nya, sehingga kata tersebut dianggap lebih representatif terhadap isi dokumen.

$$TF(t, d) = \frac{f_{t,d}}{\sum_{k=1}^n f_{k,d}} \quad (1)$$

$TF(t,d)$ adalah nilai *Term Frequency* untuk term (kata), $f_{t,d}$ adalah jumlah munculnya kata t dalam d , $\sum_{k=1}^n f_{k,d}$ adalah total jumlah seluruh kata dalam d .

2) *Inverse Document Frequency (IDF)*

Inverse Document Frequency (IDF) berperan dalam menilai seberapa jarang suatu kata muncul dengan melihat distribusinya di seluruh dokumen dalam satu korpus, di mana istilah yang frekuensinya rendah akan diberikan bobot lebih tinggi dibandingkan dengan kata-kata umum yang sering dijumpai.

$$IDF(t) = \log \left(\frac{N}{df_t} \right) \quad (2)$$

Rumus tersebut menghasilkan nilai bobot yang menunjukkan tingkat kepentingan sebuah kata dalam keseluruhan korpus. Jika sebuah term hanya muncul pada sedikit dokumen sehingga nilai df_t kecil, maka rasio $\frac{N}{df_t}$ menjadi besar dan menghasilkan nilai IDF tinggi. Kondisi tersebut menunjukkan bahwa kata bersifat unik dan informatif dalam membedakan suatu dokumen dari dokumen lainnya. Sebaliknya, jika sebuah kata muncul di hampir semua dokumen sehingga nilai df_t besar, maka IDF bernilai rendah dan menandakan bahwa kata tersebut terlalu umum sehingga tidak memberikan kontribusi signifikan dalam proses klasifikasi.

E. Handling Imbalance Data (SMOTE)

Teknik SMOTE (*Synthetic Minority Over-sampling Technique*) digunakan dalam penelitian guna menyeimbangkan kelas. SMOTE berfungsi untuk mengatasi ketidakseimbangan data dengan cara menghasilkan data sintesis pada kelas minoritas melalui teknik interpolasi antara data-data yang serupa, sehingga distribusi antar kelas menjadi lebih proporsional dan model mampu melakukan pembelajaran dengan lebih efektif. Pendekatan ini telah banyak digunakan dalam kajian analisis sentimen studi klasifikasi sentimen ulasan aplikasi Mobile JKN, penggunaan SMOTE bersama LSTM mampu meningkatkan *precision*, *recall*, dan *F1-score* pada kelas minoritas [28].

F. Splitting Dataset

Tahap *data splitting* dilakukan untuk memastikan validitas serta kemampuan generalisasi dari model yang dikembangkan. Dataset yang telah terkumpul selanjutnya dipisahkan ke dalam dua kelompok, yakni data latih yang dimanfaatkan untuk proses pembentukan model serta data uji yang digunakan sebagai sarana penilaian terhadap performa model. Data ini berperan dalam menilai kinerja akhir model secara objektif, sehingga dapat diketahui sejauh mana model mampu melakukan prediksi dengan baik pada data baru yang belum pernah ditemui [29].

G. Modeling Algoritma Support Vector Machine

Support Vector Machine (SVM) merupakan algoritma klasifikasi yang diperkenalkan oleh Vapnik pada tahun 1992 [30] sebagai pendekatan matematis yang bertujuan mencari *hyperplane* paling optimal sebagai pemisah antar kelas dalam ruang fitur, sehingga data dari kelas yang berbeda dapat dipisahkan secara maksimal. Pendekatan ini membuat SVM

sangat efektif terutama pada data berdimensi tinggi seperti teks, karena model berusaha menemukan pemisah yang paling optimal berdasarkan prinsip margin maksimum.

$$K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j) \quad (3)$$

Pada rumus tersebut menjelaskan konsep kernel trick dalam Support Vector Machine, yaitu teknik yang memungkinkan SVM memetakan data dari ruang asli yang mungkin tidak terpisahkan secara linear ke ruang fitur berdimensi lebih tinggi melalui fungsi transformasi $\phi(x)$. Namun, alih-alih menghitung transformasi tersebut secara eksplisit *which computationally expensive* SVM menggunakan fungsi kernel $K(\cdot)$ untuk langsung menghitung hasil dot product di ruang fitur baru tanpa harus memproyeksikan data ke dimensi tinggi secara nyata. Hal ini membuat SVM mampu mempelajari pola kompleks dan non-linear dengan efisiensi yang jauh lebih tinggi. Jika nilai kernel antara dua sampel besar, berarti kedua data tersebut memiliki kemiripan tinggi di ruang fitur hasil transformasi sebaliknya, nilai kecil menandakan perbedaan yang lebih jauh. Dengan menggunakan kernel seperti RBF (*Radial Basis Function*), *Polynomial*, atau *Sigmoid*.

H. Modeling Algoritma XgBoost

Extreme Gradient Boosting (XGBoost) merupakan model *ensemble learning* berbasis *gradient boosting* yang dikembangkan oleh Chen & Guestrin sebagai teknik yang dirancang untuk memberikan kinerja tinggi, efisiensi komputasi, serta kemampuan generalisasi yang kuat pada permasalahan klasifikasi dan regresi [31]. XGBoost bekerja dengan membangun model secara bertahap (*sequential boosting*), setiap pohon keputusan (*decision tree*) baru dibentuk untuk memperbaiki kesalahan prediksi dari keputusan sebelumnya melalui pendekatan fungsi kerugian yang diminimalkan menggunakan *gradient optimization*.

$$\mathcal{L} = \sum_{i=1}^n l(y_i, \hat{y}_i^{(t)}) + \sum_{k=1}^t \Omega(f_k) \quad (4)$$

Rumus objektif XGBoost menggabungkan fungsi kerugian yang mengukur kesalahan prediksi dengan fungsi regularisasi yang mengontrol kompleksitas struktur pohon. Kombinasi ini memungkinkan XGBoost menghasilkan model yang akurat tetapi tetap stabil dan tidak *overfitting*.

I. Modeling Algoritma AdaBoost

Adaptive Boosting (AdaBoost) [32] merupakan model *ensemble learning* yang dikembangkan oleh Freund & Schapire (1997) dan menjadi salah satu metode boosting paling awal yang terbukti efektif dalam peningkatan akurasi model klasifikasi. AdaBoost bekerja membangun serangkaian *weak learners* dengan pohon keputusan kecil (*decision stump*) yang dilatih secara berurutan. Pada setiap iterasi, AdaBoost memberi nilai bobot lebih besar kepada data latih yang salah diprediksi pada perulangan data sebelumnya, sehingga model berikutnya akan lebih fokus pada sampel-sampel yang sulit.

$$H(x) = \text{sign}(\sum_{t=1}^T \alpha_t h_t(x)) \quad (5)$$

Rumus tersebut merupakan representasi dari model akhir AdaBoost yang menggabungkan seluruh *weak learners* menjadi satu *strong classifier*. Dengan mekanisme ini, AdaBoost secara efektif melakukan *weighted majority voting*, di mana keputusan akhir bukan ditentukan oleh satu model saja, tetapi oleh kombinasi puluhan hingga ratusan *weak learner* yang bekerja secara sinergis. Pendekatan agregatif ini membuat AdaBoost mampu menghasilkan model yang kuat dan stabil meskipun setiap *weak learner* memiliki kapasitas prediktif yang terbatas.

J. Hyperparameter Tuning

Hyperparameter tuning [33] adalah proses sistematis untuk mengeksplorasi dan menemukan kombinasi nilai hyperparameter yang optimal dalam algoritma machine learning, dengan tujuan meningkatkan performa model secara signifikan. Teknik ini memungkinkan menilai stabilitas dan menyamaratakan model dengan teliti pada berbagai set hyperparameter. Penelitian oleh Putri, dkk. mengaplikasikan tuning hyperparameter pada beberapa algoritma klasifikasi untuk prediksi penyakit jantung dengan hasil akurasi tertinggi setelah optimasi menggunakan *Grid Search* dan *Random Search* [34].

K. Evaluasi model

1) Confusion Matrix

TABEL 1. CONFUSION MATRIX

Kelas	Klasifikasi	
Positif	True Positive (TP)	False Negative (FP)
Negatif	False Negative (FN)	True Negative (TN)

Pada Tabel 1, menjelaskan *True Positive* (TP) merepresentasikan jumlah data dalam kelas positif dan berhasil dikenali sebagai positif oleh model, sedangkan *True Negative* (TN) menunjukkan banyaknya data yang berada pada kelas negatif dan diprediksi secara tepat sebagai negatif oleh sistem. Sementara itu, *False Positive* (FP) merujuk pada kondisi ketika data yang seharusnya termasuk kelas negatif keliru diklasifikasikan positif oleh model, dan *False Negative* (FN) menggambarkan situasi saat data yang sebenarnya bersifat positif malah salah diprediksi sebagai negatif oleh model.

Hasil evaluasi menggunakan *confussion matrix* menunjukkan kinerja prediksi dengan hasil nilai berikut:

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (6)$$

- Akurasi menghitung proporsi prediksi model yang benar dengan seluruh data uji.

$$\text{Precision} = \frac{TP}{TP+FP} \quad (7)$$

- *Precision* menghitung nilai akurat model guna memberikan prediksi positif.

$$Recall = \frac{TP}{TP+FN} \quad (8)$$

- *Recall* menggambarkan kemampuan model menilai ketepatan dalam mengenali data positif.

$$F1-Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (9)$$

- *F1-Score* adalah ukuran evaluasi yang digunakan untuk menilai performa suatu model dengan cara mengombinasikan nilai precision dan recall ke dalam satu skor yang bersifat seimbang.

III. HASIL DAN PEMBAHASAN

Tahap ini menjadi bagian yang merepresentasikan hasil akhir dari keseluruhan rangkaian proses penelitian yang telah dilaksanakan, mulai dari pengumpulan data melalui *crawling*, tahap pra-pemrosesan (*pre-processing*), ekstraksi fitur menggunakan metode TF-IDF, hingga penanganan ketidakseimbangan data dengan teknik SMOTE, *Hyperparameter tuning* (*GridSearchCV*) hingga evaluasi model klasifikasi menggunakan model SVM, XGBoost, dan AdaBoost. Penyajian hasil pada tahap ini bertujuan memberikan hasil menyeluruh mengenai performa setiap model mengklasifikasikan sentimen pengguna Twitter (X) terkait Kinerja Presiden Prabowo Subianto dalam jangka waktu satu tahun.

A. Pengumpulan Data

Pengumpulan data riset dilakukan melalui proses *crawling* menggunakan API Twitter (X) pada web *scraping* untuk memperoleh teks unggahan (*tweet*) yang relevan dengan topik kinerja Presiden Prabowo. Proses *crawling* dilakukan dengan memanfaatkan kata kunci, tagar hashtag (#), serta frasa yang sering digunakan oleh pengguna dalam membahas terkait topik tersebut. Data mentah yang di peroleh peneliti beragam data, namun peneliti hanya mengambil bagian data penting seperti *id*, *created_at*, *created_at_datetime*, *text*, *user_id*, *user_name* yang dapat di proses guna analisis sentimen. Data dari hasil *crawling* terkumpul sebanyak 2.451 data, dengan rentang waktu 20-10-2024 hingga 21-10-2025 dan hanya di ambil kolom bagian *text* dari beberapa kolom data tersebut untuk di *processing*.

B. Pre-processing

Pada tahap ini dilakukan guna membersihkan dan menormalkan data sehingga siap digunakan untuk proses analisis sentimen. Beberapa tahapan yang dilakukan mencakup pembersihan teks dari tanda baca, *URL*, angka, serta karakter yang tidak diperlukan. Case folding dilakukan untuk menstandarkan teks dengan mengubah seluruh huruf menjadi huruf kecil (*lowercase*), kemudian teks dipecah menjadi kata-kata melalui *tokenization*, *stopword removal* untuk menghilangkan kata umum yang tidak berpengaruh

terhadap analisis. Terakhir, Proses *stemming* bertujuan mengonversi berbagai bentuk kata menjadi kata dasar, sehingga data teks menjadi lebih seragam dan lebih mudah diolah dalam tahap pemodelan. Serangkaian proses tersebut bertujuan untuk meningkatkan kualitas data *tweet*, sehingga model klasifikasi dapat bekerja lebih optimal dan menghasilkan perkiraan data yang lebih akurat. Berikut tahapan pre-processing:

1) *Cleansing Data*

Pada tahapan pertama pre-processing data. Pada tahap bertujuan menghilangkan elemen tidak relevan seperti tautan (*URL*), mention akun (@*user*), emoji, angka, tanda baca, dan karakter khusus, seperti pada tampilan tabel 2.

TABEL 2.
HASIL CLEANSING

No	text	clean_text
1	@KompasTV Lah, siapa dia evakusia kinerja presiden? Yg berhak evaluasi presiden itu ya rakyat.	Lah siapa dia evakusia kinerja presiden Yg berhak evaluasi presiden itu ya rakyat
2	Satu Tahun Pemerintahan Presiden Prabowo, Pertamina Perkokoh Ketahanan Energi kinerja yang bagus nih 🇮🇩 https://t.co/5yfdHZmwE6	Satu Tahun Pemerintahan Presiden Prabowo Pertamina Perkokoh Ketahanan Energi kinerja yang bagus nih

Pada tahap ini juga di lakukan penghapusan duplikat data sebanyak 12 data duplikat. data awal sebanyak 2.451 data menjadi 2.439 data.

2) *Case Folding*

Pada tahap ini menyeragamkan semua huruf menjadi huruf kecil (*lowercase*). Fase *case folding*, bertujuan supaya data menjadi lebih konsisten dan memudahkan model guna mengenali pola kata secara lebih akurat. Seperti hasil pada tabel 3.

TABEL 3.
HASIL CASE FOLDING

No	clean_text	lowercase_text
1	Lah siapa dia evakusia kinerja presiden Yg berhak evaluasi presiden itu ya rakyat	lah siapa dia evakusia kinerja presiden yg berhak evaluasi presiden itu ya rakyat
2	Satu Tahun Pemerintahan Presiden Prabowo Pertamina Perkokoh Ketahanan Energi kinerja yang bagus nih	satu tahun pemerintahan presiden prabowo pertamina perkokoh ketahanan energi kinerja yang bagus nih

3) *Tokenization*

Tahap tokenisasi memiliki peran untuk menguraikan teks menjadi satuan-satuan kata yang dikenal sebagai token, seperti "pemerintahan presiden prabowo" menjadi token

“pemerintahan”, “presiden”, “prabowo”. Berikut hasil pemecah kata menjadi token pada tabel 4.

TABEL 4.
HASIL TOKENIZATION

No	lowercase_text	tokens
1	lah siapa dia evakusia kinerja presiden yg berhak evaluasi presiden itu ya rakyat	['lah', 'siapa', 'dia', 'evakusia', 'kinerja', 'presiden', 'yg', 'berhak', 'evaluasi', 'presiden', 'itu', 'ya', 'rakyat']
2	satu tahun pemerintahan presiden prabowo pertamina perkokoh ketahanan energi kinerja yang bagus nih	['satu', 'tahun', 'pemerintahan', 'presiden', 'prabowo', 'pertamina', 'perkokoh', 'ketahanan', 'energi', 'kinerja', 'yang', 'bagus', 'nih']

4) Stopword removal

Stopword removal adalah tahap pra-pemrosesan yang dilakukan dengan cara menghapus kata-kata umum seperti “yang”, “dan”, atau “ketika”. Langkah ini bertujuan mengurangi kata-kata yang tidak relevan sehingga model dapat lebih fokus pada istilah penting untuk analisis teks sentimen. Berikut hasil dari penelitian pada tahap bagian ini pada tabel 5.

TABEL 5.
HASIL STOPWORD REMOVAL

No	tokens	tokens_nostop
1	['lah', 'siapa', 'dia', 'evakusia', 'kinerja', 'presiden', 'yg', 'berhak', 'evaluasi', 'presiden', 'itu', 'ya', 'rakyat']	['lah', 'siapa', 'evakusia', 'kinerja', 'presiden', 'yg', 'berhak', 'evaluasi', 'presiden', 'rakyat']
2	['satu', 'tahun', 'pemerintahan', 'presiden', 'prabowo', 'pertamina', 'perkokoh', 'ketahanan', 'energi', 'kinerja', 'yang', 'bagus', 'nih']	['satu', 'tahun', 'pemerintahan', 'presiden', 'prabowo', 'pertamina', 'perkokoh', 'ketahanan', 'energi', 'kinerja', 'bagus', 'nih']

5) Stemming

Tahapan *stemming* merupakan pengubahan setiap kata dalam bentuk dasar melalui penghapusan imbuhan seperti awalan, sisipan, dan akhiran dengan memanfaatkan pustaka *sastrawi*. Tahap *stemming* ini dilakukan untuk memastikan bahwa variasi kata seperti “membantu”, “dibantu”, dan “bantuan” dikenali sebagai satu kata dasar yang sama yaitu “bantu” seperti hasil pada tabel 6.

TABEL 6.
HASIL STEMMING

No	tokens_nostop	Tokens_stemmed
1	['lah', 'siapa', 'evakusia', 'kinerja', 'presiden', 'yg', 'berhak', 'evaluasi', 'presiden', 'rakyat']	['lah', 'siapa', 'evakusia', 'kerja', 'presiden', 'yg', 'hak', 'evaluasi', 'presiden', 'rakyat']

2	['satu', 'tahun', 'pemerintahan', 'presiden', 'prabowo', 'pertamina', 'perkokoh', 'ketahanan', 'energi', 'kinerja', 'bagus', 'nih']	['satu', 'tahun', 'perintah', 'presiden', 'prabowo', 'pertamina', 'kokoh', 'tahan', 'energi', 'kerja', 'bagus', 'nih']
---	---	--

Setelah tahap *stemming*, data tersebut diubah menjadi kalimat seperti hasil pada tabel 7.

TABEL 7.
HASIL PRE-PROCESSING

No	tokens_stemmed	text_preprocessed
1	['lah', 'siapa', 'evakusia', 'kerja', 'presiden', 'yg', 'hak', 'evaluasi', 'presiden', 'rakyat']	lah siapa evakusia kerja presiden yg hak evaluasi presiden rakyat
2	['satu', 'tahun', 'perintah', 'presiden', 'prabowo', 'pertamina', 'kokoh', 'tahan', 'energi', 'kerja', 'bagus', 'nih']	satu tahun perintah presiden prabowo pertamina kokoh tahan energi kerja bagus nih

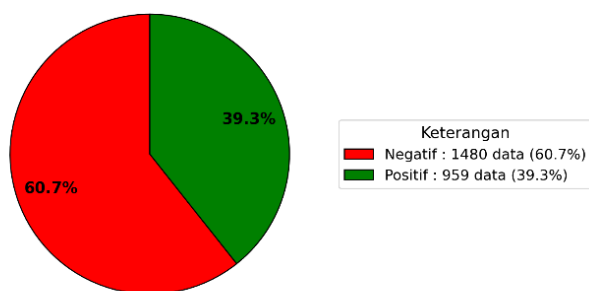
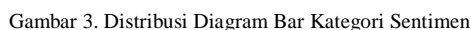
C. Labeling Dataset

Pelabelan sentimen dalam penelitian ini menggunakan pendekatan *lexicon-based* dengan memanfaatkan korpus sentimen *InSet* yang telah banyak digunakan dalam analisis teks berbahasa Indonesia.

TABEL 8.
HASIL LABEL SCORE

No	text_preprocessed	sentiment score	sentiment label
1	walaupun presiden klau kerja buruk mesti nggk dapet sebut sungguh adil bgi rakyat	-5.0	Negatif
2	satu tahun perintah presiden prabowo pertamina kokoh tahan energi kerja bagus nih	1.0	Positif
3	baper itu serah presiden mo pecat menteri apa engga lagi liat noh kerja lo udh benrr ga jd menteri aja data bocor kena hack judul gmn judul kagak becus isi boomer gaptek	-41.0	Negatif

Pada Tabel 8, menjelaskan setiap kata dalam *tweet* diberikan skor sentimen berdasarkan frase kata positif dan negatif. Akumulasi nilai seluruh kata kemudian menentukan kelas sentimen akhir, skor lebih besar dari 0 bernilai (1) dikategorikan sebagai positif, dan jika skor nilai kurang dari 0 bernilai (-1) negatif dikategorikan sebagai negatif. Berikut hasil dari pelabelan *tweet* dengan pendekatan *lexicon-based* dan kamus sentimen *InSet*. Dari tahap pelabelan *tweet* yang relevan dimuat data sebanyak 2.439 dengan jumlah data label negatif 1.480 dan label positif 959 dengan visualisasi hasil pada gambar 3, gambar 4, dan tabel 9.



Gambar 4. Distribusi Diagram Bar Kategori Sentimen

TABEL 9.
HASIL PELABELAN DATA

Sentimen	Jumlah <i>Tweet</i>	Persentase (%) Hasil
Negatif	1.480	60.7%
Positif	959	39.3%
Total	2.439	100%

Gambar 5. Wordcloud Positif

Pada gambar 5, *WordCloud* positif menampilkan representasi visual dari kata yang sering muncul pada *tweet* dengan *tweet* positif terkait kinerja Presiden Prabowo. Kata dengan ukuran terbesar seperti “*presiden*”, “*prabowo*”, “*kerja presiden*”, dan “*perintah presiden*” menunjukkan frekuensi kemunculan yang tinggi, memperlihatkan fokus utama publik terhadap kepemimpinan dan kebijakan yang dianggap memberi dampak baik. Selain itu, kata-kata seperti “*masyarakat puas*”, “*baik*”, “*nilai positif*”, “*program*”, “*hasil survei*”, dan “*percaya*” memperlihatkan adanya apresiasi publik terhadap capaian dan performa pemerintahan.

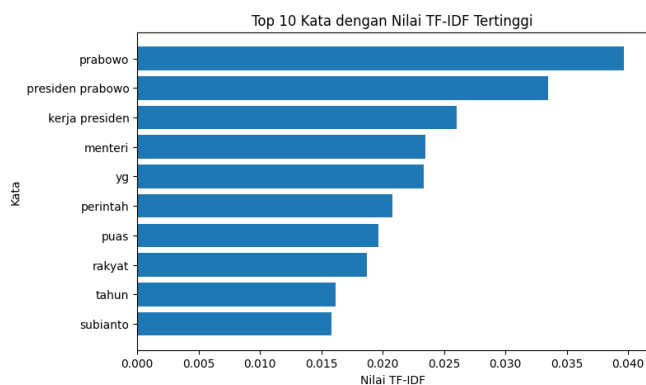


Gambar 6. WordCloud Negatif

Pada gambar 6, *WordCloud* visualisasi kata yang sering muncul dalam *tweet* negatif mengenai kinerja Presiden Prabowo. Kata-kata yang berukuran besar seperti “*presiden*”, “*prabowo*”, “*kerja presiden*”, dan “*rakyat*” menunjukkan bahwa kritik publik terutama berfokus pada kinerja dan kebijakan pemerintahan. Selain itu, istilah seperti “*ga*”, “*tdk*”, “*bukan*”, “*masalah*”, dan “*kritik*” menggambarkan munculnya ketidakpuasan dan keraguan publik terhadap efektivitas program pemerintah.

D. Feature Extraction (TF-IDF)

Dalam penelitian ini, tahap ekstraksi fitur diterapkan dengan memanfaatkan TF-IDF *Vectorizer*, yaitu metode yang menetapkan bobot pada setiap kata berdasarkan frekuensi kemunculannya di dalam dokumen serta distribusinya di seluruh korpus. Melalui pendekatan ini, kata-kata yang memiliki tingkat relevansi dan informasi lebih tinggi terhadap topik pembahasan akan memberikan kontribusi yang lebih besar dalam proses pembentukan model. Penerapan TF-IDF pada penelitian ini menggunakan pengaturan parameter *ngram_range* (1,2) untuk menangkap pola kata dalam bentuk unigram dan bigram, serta *max_features* sebesar 5000 guna membatasi jumlah fitur sehingga proses pemodelan tetap berjalan secara efisien. Hasil dari proses ekstraksi fitur pada penelitian ini ditampilkan pada Gambar 7 dan Tabel 10.



Gambar 7. Top 10 Kata Nilai TF-IDF Tertinggi

TABEL 10.
KATA NILAI TF-IDF TERTINGGI

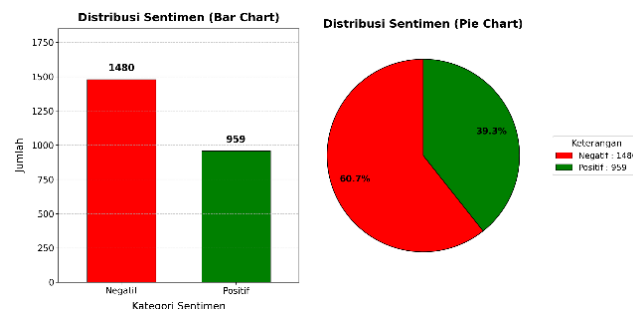
Word	TF-IDF
prabowo	0.039640
presiden prabowo	0.033500
kerja presiden	0.026023
menteri	0.023508
yg	0.023347
perintah	0.020818
puas	0.019685
rakyat	0.018708
tahun	0.016146
subianto	0.015809

Berdasarkan gambar 7 dan tabel 10, visualisasi pada grafik Top 10 Nilai TF-IDF Tertinggi dan tabel Kata dengan Nilai TF-IDF Tertinggi, terlihat bahwa kata “prabowo” memiliki nilai TF-IDF paling tinggi, diikuti oleh frasa “presiden prabowo” dan “kerja presiden”. Hal ini menunjukkan bahwa kata-kata tersebut merupakan istilah paling dominan dan paling tipikal dalam keseluruhan korpus *tweet* yang dianalisis. Selain itu, kata seperti “menteri”, “rakyat”, “tahun”, dan “puas” juga memiliki bobot yang besar, menandakan bahwa topik terkait kinerja pemerintahan, kepuasan publik, serta dinamika politik menjadi fokus utama dalam percakapan warganet. Temuan ini memberikan gambaran awal mengenai isu-isu yang paling sering dibahas dan menjadi konteks penting sebelum dilakukan pemodelan klasifikasi sentimen.

E. Handling Imbalance Data (SMOTE)

Pada penelitian ini menerapkan metode SMOTE (*Synthetic Minority Over-sampling Technique*) sebagai pendekatan menangani permasalahan ketidakseimbangan kelas dalam pembelajaran mesin dengan cara menambah data buatan pada kelas minoritas. Teknik ini bekerja dengan membentuk sampel sintesis melalui proses interpolasi antara data-data minoritas yang memiliki kemiripan, sehingga proporsi antar kelas menjadi lebih seimbang dan mendukung proses pelatihan model. Pembentukan sampel baru tidak dilakukan secara acak, tetapi didasarkan pada kedekatan

posisi data dalam ruang fitur agar karakteristik data tetap terjaga.

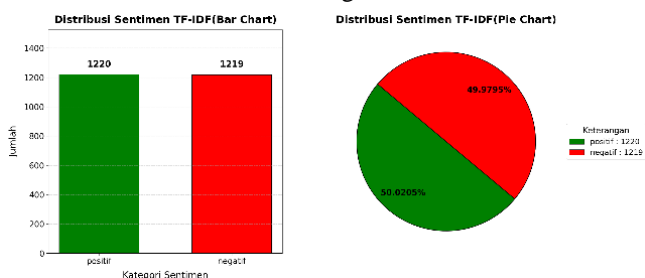


Gambar 8. Distribusi Label Sebelum TF-IDF dan SMOTE

TABEL 11.
LABEL SENTIMEN AWAL

Sentimen (Awal)	Jumlah Tweet	Persentase (%) Hasil
Negatif	1.480	60.7%
Positif	959	39.3%
Total	2.439	100%

Pada gambar 8 dan Tabel 11 merupakan visualisasi pelabelan data sentimen awal memperoleh hasil *tweet* negatif berjumlah 1.480 data dengan persentase 60.7%, sedangkan *tweet* positif berjumlah 959 data dengan persentase 39.3%. Setelah melakukan labeling pada *tweet*, selanjutnya data diproses melalui tahap TF-IDF (ekstraksi Fitur). Pelabelan setelah tahap TF-IDF menghasilkan data sebanyak 2.439 data dengan distribusi perlabel *tweet* positif berjumlah 1.220 data dengan persentase 50.02% dan *tweet* negatif berjumlah 1.219 data dengan persentase 49.97%. Berikut hasil visualisasi Pelabelan setelah TF-IDF Pada gambar 9 dan tabel 12.

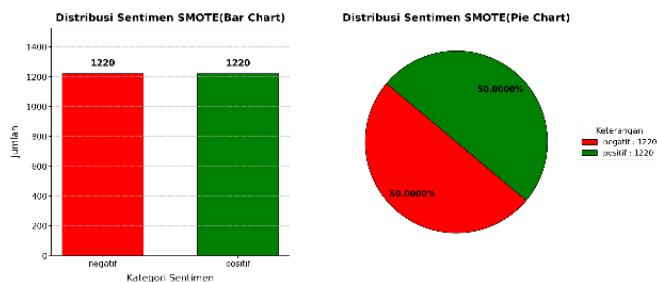


Gambar 9. Distribusi Label Pasca TF-IDF

TABEL 12.
LABEL SENTIMEN SETELAH TF-IDF

Sentimen (TF-IDF)	Jumlah Tweet	Persentase (%) Hasil
Negatif	1.219	49.97%
Positif	1.220	50.02%
Total	2.439	100%

Penerapan metode SMOTE juga di terapkan guna mengantasi ketidakseimbangan data.



Gambar 10. Distribusi Label SMOTE

TABEL 13.
LABEL SENTIMEN SMOTE

Sentimen (SMOTE)	Jumlah Tweet	Persentase (%) Hasil
Negatif	1.220	50.0%
Positif	1.220	50.0%
Total	2.440	100%

Pada gambar 10 dan tabel 13 Pendekatan SMOTE menghasilkan data seimbang. Distribusi label *tweet* negatif 1.220 data dengan persentase 50.0% dan distribusi label *tweet* positif 1.220 data dengan persentase 50.0%.

Hasil eksplorasi distribusi label menunjukkan bahwa pada tahap TF-IDF, jumlah data positif dan negatif sudah berada pada kondisi hampir seimbang (perbedaannya sangat kecil). Setelah SMOTE diterapkan, distribusi tersebut menjadi benar-benar seimbang. Temuan ini mengindikasikan bahwa kontribusi SMOTE pada penelitian ini lebih bersifat “penyempurnaan” distribusi daripada perubahan besar pada struktur data, karena ketidakseimbangan kelas sebelum SMOTE sudah rendah. SMOTE menambah sampel sintetis pada kelas yang lebih sedikit, dampak yang paling umum terlihat adalah meningkatnya kemampuan model dalam mengenali kelas minoritas (misalnya *recall* minoritas dan *macro-F1*), meskipun terkadang terdapat *trade-off* berupa penurunan *precision* akibat bertambahnya prediksi positif atau negatif yang keliru.

F. Splitting Dataset

Tahapan pembagian dataset dilakukan dengan cara memisahkan data ke dalam training set dan testing set, dengan tujuan agar model machine learning dapat mempelajari pola dari data latih sekaligus dievaluasi kemampuannya dalam melakukan prediksi pada data yang belum pernah digunakan sebelumnya.

TABEL 14.
SPLITTING DATASET

Dataset	Jumlah Data	Persentase (%)
Data Latih	1.952	80%
Data Uji	488	20%
Total	2.440	100%

Pada tabel 14 dataset dibagi menggunakan rasio 80% sebagai data latih dan 20% sebagai data uji. Pembagian ini merupakan konfigurasi umum yang digunakan dalam

berbagai studi *machine learning* terutama ada analisis sentimen, karena dianggap mampu memberikan keseimbangan secara optimal dalam jumlah data guna pelatihan model dan jumlah data yang kompeten untuk mengevaluasi performanya.

G. Pemodelan Dan Hyperparameter Tunning

Pada riset ini menggunakan tiga Pemodelan algoritma *Machine Learning* yaitu *Support Vector Machine* (SVM), *Extreme Gradient Boosting* (XGBoost), dan *Adaptive Boosting* (AdaBoost). Sebelum model digunakan untuk prediksi akhir, dilakukan proses hyperparameter tuning guna memperoleh kombinasi parameter terbaik. Pada penelitian ini menggunakan *GridSearchCV* sebagai metode pencarian hyperparameter berbasis *exhaustive search* yang menguji seluruh kombinasi parameter dalam suatu grid pencarian, kemudian memilih konfigurasi terbaik berdasarkan nilai evaluasi *cross-validation*.

1) Pemodelan Support Vector Machine (SVM)

Penelitian ini menggunakan pemodelan SVM Berbagai kernel seperti *Linier*, *Polynomial*, *RBF*, dan *Sigmoid*. Peneliti melakukan pelatihan model dengan data latih dan data uji dari splitting dataset. Hasil penelitian menggunakan algoritma *Support Vector Machine* (SVM) kernel *Linear*, *RBF*, *Polynomial*, dan *Sigmoid* menunjukkan adanya variasi performa yang cukup signifikan pada proses klasifikasi sentimen. Penyajian hasil dari penelitian pemodelan SVM di tampilkan pada tabel 15.

TABEL 15.
PEMODELAN SVM

No	Model	Best_Params	Accuracy
1	SVM_Linear	{'C': 1, 'gamma': 'scale', 'kernel': 'linear'}	0.8074
2	SVM_RBF	{'C': 10, 'gamma': 'scale', 'kernel': 'rbf'}	0.8135
3	SVM_Poly	{'C': 1, 'coef0': 1.0, 'gamma': 'scale', 'kernel': 'poly'}	0.8115
4	SVM_Sigmoid	{'C': 1, 'coef0': 0.5, 'gamma': 'scale', 'kernel': 'sigmoid'}	0.7889

Pada tabel 15, setiap kernel diuji menggunakan *GridSearchCV* untuk memperoleh kombinasi hyperparameter terbaik, seperti nilai *C*, *gamma*, dan *coef0*, sehingga setiap model dapat bekerja secara optimal. Kernel RBF memperoleh akurasi tertinggi dengan hasil 0.813. Kinerja terbaik ini mengidentifikasikan bahwa kernel RBF mampu menangkap pola non-linear secara lebih efektif, terutama pada data teks TF-IDF yang memiliki dimensi tinggi dan hubungan antarfitur yang kompleks.

2) Pemodelan *Extreme Gradient Boosting* (XGBoost)

Penelitian ini menggunakan algoritma *Extreme Gradient Boosting* (XGBoost) karena metode ini dikenal efektif dalam menangani data berukuran besar serta berdimensi tinggi seperti representasi TF-IDF, serta performanya yang unggul pada klasifikasi. Untuk memperoleh konfigurasi model terbaik, dilakukan proses *Hyperparameter Tuning* menggunakan *GridSearchCV*, yang menguji berbagai kombinasi parameter penting XGBoost. Pada tahap pemodelan ini untuk mempermudah interpretasi model, dilakukan mapping label dengan nilai (0) sebagai negatif dan nilai (1) sebagai positif, hal ini digunakan pada tahap pelatihan untuk memastikan konsistensi format label yang diproses oleh XGBoost.

TABEL 16.
PEMODELAN ALGORITMA XGBOOST

No	Model	Best_Params	Accuracy
1	XGBosst	{'colsample_bytree': 0.7, 'learning_rate': 0.2, 'max_depth': 6, 'n_estimators': 300, 'subsample': 0.7}	0.8443

Model XGBoost terbaik yang diperoleh dari kombinasi parameter tersebut mampu mencapai nilai akurasi sebesar 0.8443, yang menandakan model memiliki kinerja klasifikasi yang sangat baik pada data sentimen yang telah diproses menggunakan TF-IDF dan SMOTE.

3) Pemodelan *Adaptive Boosting* (AdaBoost).

Pemodelan algoritma *Adaptive Boosting* (AdaBoost) diterapkan sebagai salah satu pendekatan boosting yang bertujuan memperkuat kinerja classifier sederhana (*weak learner*) melalui proses pembelajaran berulang. AdaBoost bekerja dengan meningkatkan fokus pada data yang sebelumnya salah diklasifikasikan. Pada setiap iterasi, sampel yang keliru mendapat bobot lebih besar sehingga model berikutnya lebih menekankan perbaikan terhadap kesalahan tersebut. Sama halnya dengan Pemodelan XGBoost, pada tahap pemodelan AdaBoost untuk mempermudah interpretasi model, dilakukan *mapping* label dengan nilai (0) sebagai negatif dan nilai (1) sebagai positif, hal ini digunakan pada tahap pelatihan untuk memastikan konsistensi format label yang diproses oleh XGBoost.

TABEL 17.
PEMODELAN ALGORITMA ADABOOST

No	Model	Best_Params	Accuracy
1	AdaBosst	{'learning_rate': 1.0, 'n_estimators': 300}	0.7818

Untuk memperoleh performa terbaik, dilakukan proses *Hyperparameter Tuning* menggunakan *GridSearchCV*. Hasil menunjukkan bahwa kombinasi parameter optimal bagi model AdaBoost dan menghasilkan akurasi sebesar 0.7818, Nilai ini menunjukkan bahwa performa model cukup stabil dan tidak mengalami *overfitting* yang signifikan.

H. Evaluasi Pemodelan

Tahap evaluasi model merupakan bagian paling penting dalam penelitian ini karena menjadi dasar untuk menentukan algoritma mana yang memiliki performa terbaik dalam melakukan klasifikasi sentimen terhadap opini publik mengenai kinerja Presiden Prabowo di platform Twitter (X). Penelitian ini melibatkan penerapan dan perbandingan beberapa algoritma *machine learning* dengan karakteristik yang berbeda guna mengevaluasi kinerja masing-masing algoritma secara sistematis, yaitu *Support Vector Machine* (SVM) sebagai baseline model, XGBoost sebagai algoritma *gradient boosting* yang kuat, serta AdaBoost sebagai pendekatan *adaptive boosting* yang adaptif dalam memperbaiki kesalahan iteratif.

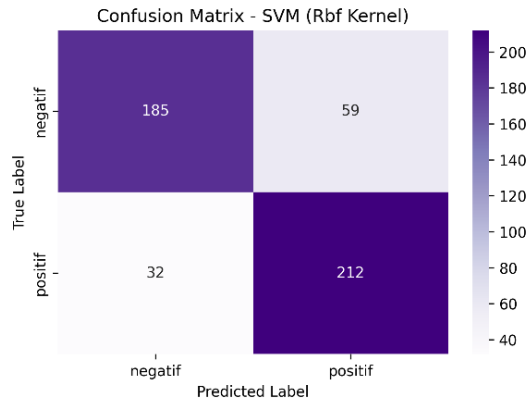
TABEL 18.
HASIL EVALUASI PEMODELAN

No	Model	Best_Params	Accuracy
1	SVM_RBF	{'C': 10, 'gamma': 'scale', 'kernel': 'rbf'}	0.8135
2	XGBoost	{'colsample_bytree': 0.7, 'learning_rate': 0.2, 'max_depth': 6, 'n_estimators': 300, 'subsample': 0.7}	0.8443
3	AdaBoost	{'learning_rate': 1.0, 'n_estimators': 300}	0.7818

Berdasarkan tabel 18, menyajikan ringkasan hasil evaluasi tiga model klasifikasi, pada algoritma *Support Vector Machine* (SVM) peneliti melakukan uji modeling dengan SVM kernel, peneliti menemukan SVM kernel RBF menghasilkan akurasi yang cukup tinggi dibandingkan SVM kernel lainnya, dengan parameter terbaik menghasilkan akurasi 0.8135. Selain itu, pemodelan algoritma XGBoost juga menghasilkan akurasi sebesar 0.8443, sedangkan algoritma AdaBoost menghasilkan akurasi 0.7818. Masing-masing model telah melalui proses *Hyperparameter Tuning* menggunakan teknik *GridSearchCV* untuk menemukan kombinasi parameter paling optimal sebelum dilakukan pengujian terhadap data uji. ketiga model algoritma *machine learning* menghaikan akurasi yang cukup tinggi.

Hasil evaluasi model *Support Vector Machine* dengan kernel RBF menunjukkan bahwa konfigurasi terbaik yang diperoleh melalui *GridSearchCV* adalah C = 10, gamma = 'scale', dan kernel = 'rbf', dengan akurasi uji sebesar 0.8135. Model XGBoost yang telah melalui proses *hyperparameter tuning* dengan parameter optimal {'colsample_bytree': 0.7, 'learning_rate': 0.2, 'max_depth': 6, 'n_estimators': 300, 'subsample': 0.7} dan menghasilkan akurasi sebesar 0.8443. Model AdaBoost yang telah dioptimalkan menggunakan parameter {'learning_rate': 1.0, 'n_estimators': 300} menunjukkan performa yang cukup baik dalam mengklasifikasikan sentimen dua kelas (negatif dan positif). Berdasarkan *classification report*, model memperoleh akurasi keseluruhan sebesar 0.7868.

Berikut hasil evaluasi *confusion matrix* dan *classification report* pada pemodelan.



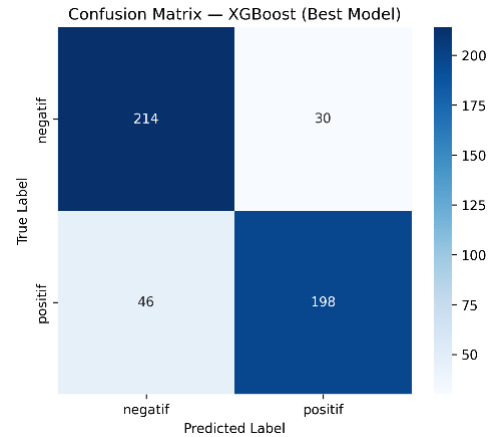
Gambar 11. Confusion Matrix SVM (RBF Kernel)

Confusion matrix yang ditampilkan pada Gambar 11, memberikan penjelasan mengenai sebaran hasil prediksi model. Dari total 244 data berlabel negatif, model SVM mampu mengklasifikasikan 185 data secara tepat, meskipun masih terdapat 59 data negatif yang salah diprediksi sebagai positif. Sementara itu, dari 244 data positif, model berhasil mengidentifikasi dengan benar sebanyak 212 data, dan hanya 32 data yang salah diklasifikasikan ke dalam kelas negatif.

TABEL 19.
CLASSIFICATION REPORT SVM (RBF KERNEL)

Klasifikasi	<i>precision</i>	<i>recall</i>	<i>f1-score</i>	<i>support</i>
negatif	0.85	0.76	0.80	244
positif	0.78	0.87	0.82	244
accuracy			0.81	488
Macro avg	0.82	0.81	0.81	488
weighted avg	0.82	0.81	0.81	488

Berdasarkan *classification report* pada tabel 19, model mencapai *precision* 0.85 untuk kelas negatif dan 0.78 untuk kelas positif. Nilai *recall* menunjukkan pola yang sebaliknya, di mana kelas positif memiliki *recall* lebih tinggi (0.87) dibandingkan kelas negatif (0.76), yang berarti model lebih sensitif dalam mendeteksi sentimen positif. *F1-score* kedua kelas berada di kisaran 0.80–0.82, sehingga performa model dapat dianggap stabil dan seimbang. Pola ini mengindikasikan bahwa SVM dengan kernel RBF mampu melakukan generalisasi dengan baik, ditunjukkan oleh tingkat kesalahan prediksi yang rendah dan relatif seimbang pada kedua kelas.



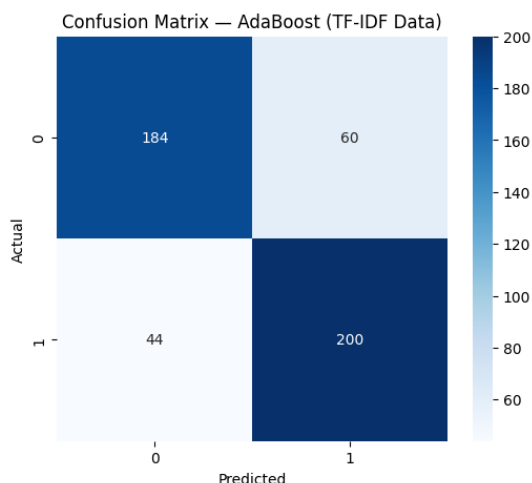
Gambar 12. Confusion Matrix model XGBoost

Berdasarkan *confusion matrix* yang ditampilkan pada Gambar 12, kinerja model XGBoost terlihat cukup baik dalam melakukan klasifikasi. Pada kelas sentimen negatif, model mampu memberikan 214 prediksi yang benar dan hanya 30 data yang keliru diklasifikasikan sebagai positif. Sementara itu, pada kelas sentimen positif, model juga menunjukkan performa yang kuat dengan 198 prediksi yang tepat, meskipun masih terdapat 46 data yang salah diklasifikasikan sebagai negatif.

TABEL 20.
CLASSIFICATION REPORT XGBOOST

Klasifikasi	<i>precision</i>	<i>recall</i>	<i>f1-score</i>	<i>support</i>
negatif	0.823077	0.877049	0.849206	244
positif	0.868421	0.811475	0.838983	244
accuracy	0.844262	0.844262	0.844262	0.844262
Macro avg	0.845749	0.844262	0.844095	488
weighted avg	0.845749	0.844262	0.844095	488

Berdasarkan hasil evaluasi yang disajikan pada Tabel 20, dapat disimpulkan bahwa model klasifikasi menunjukkan performa yang sangat baik dalam mengenali sentimen negatif maupun positif. Pada kelas negatif, nilai *precision* sebesar 0,822 menunjukkan bahwa sebagian besar prediksi negatif yang dihasilkan oleh model berada pada kategori yang benar. Selain itu, nilai *recall* sebesar 0,873 mencerminkan kemampuan model yang tinggi dalam mengidentifikasi hampir seluruh data negatif yang tersedia. Nilai *F1-score* sebesar 0,847 menegaskan adanya keseimbangan yang baik antara tingkat ketepatan dan kelengkapan hasil prediksi. Untuk kelas positif, model juga memperlihatkan kinerja yang solid dengan nilai *precision* sebesar 0,865 dan *recall* sebesar 0,811, yang menandakan bahwa model cukup andal dalam memprediksi sekaligus mengenali sentimen positif. Nilai *F1-score* sebesar 0,837 menunjukkan bahwa model mampu menjaga konsistensi performa pada kelas positif, meskipun karakteristik dan variasi pola teks pada sentimen ini cenderung lebih kompleks. Secara keseluruhan, model mencapai akurasi 0.842, menandakan bahwa 84,2% keseluruhan data uji berhasil diklasifikasikan dengan benar.



Gambar 13. Confusion Matrix Model AdaBoost

Mengacu pada *confusion matrix* yang ditampilkan pada Gambar 13, model AdaBoost menunjukkan kemampuan klasifikasi yang cukup baik pada kelas negatif (0) maupun positif (1). Model ini mampu memprediksi dengan benar sebanyak 184 data negatif dan 200 data positif. Namun demikian, masih ditemukan kesalahan prediksi, yaitu 60 data negatif yang keliru diklasifikasikan sebagai positif serta 44 data positif yang salah masuk ke dalam kelas negatif. Hasil tersebut mengindikasikan bahwa meskipun kinerja AdaBoost relatif stabil, model masih menghadapi tantangan dalam membedakan data negatif dan positif yang memiliki karakteristik yang mirip.

TABEL 21.
CLASSIFICATION REPORT ADABOOST

Klasifikasi	<i>precision</i>	<i>recall</i>	<i>f1-score</i>	<i>support</i>
negatif	0.807018	0.754098	0.779661	244
positif	0.769231	0.819672	0.793651	244
accuracy	0.786885	0.786885	0.786885	0.786885
Macro avg	0.788124	0.786885	0.786656	488
weighted avg	0.788124	0.786885	0.786656	488

Pada tabel 21, evaluasi yang mengindikasikan bahwa hampir 80% data uji berhasil diprediksi dengan benar. Untuk kelas negatif, model mencapai *precision* sebesar 0.8070, yang berarti sebagian besar prediksi negatif benar adanya. Namun, *recall*-nya sebesar 0.7540 menunjukkan masih terdapat contoh negatif yang salah terklasifikasi sebagai positif. Nilai *F1-score* sebesar 0.7796 menegaskan performa yang seimbang antara *precision* dan *recall* pada kelas ini. Sementara itu, kelas positif menunjukkan pola yang berlawanan, dengan *precision* sedikit lebih rendah yaitu 0.7692, tetapi *recall* lebih tinggi (0.8197). Hal ini mengindikasikan bahwa model lebih sensitif dalam mendeteksi sentimen positif, meskipun tingkat ketepatannya sedikit lebih rendah dibandingkan kelas negatif. Nilai *F1-score* kelas positif sebesar 0.7936 menunjukkan performa yang stabil. Baik nilai *macro average* maupun *weighted average* berada pada kisaran 0.786, menandakan bahwa model bekerja secara konsisten pada kedua kelas, tanpa bias

yang terlalu besar terhadap salah satu kelas. Secara keseluruhan, hasil ini menunjukkan bahwa AdaBoost mampu menangani data sentimen dengan baik meskipun performanya masih sedikit lebih rendah dibandingkan model XGBoost dan SVM-RBF dalam penelitian ini.

IV. KESIMPULAN

Penelitian ini mengembangkan bagaimana pengguna Twitter (X) mengekspresikan sentimen terhadap kinerja Presiden Prabowo dengan menggunakan tiga model klasifikasi *machine learning*, yaitu SVM, XGBoost, dan AdaBoost. Seluruh proses dimulai dari *pre-processing* data, ekstraksi fitur menggunakan TF-IDF, penyeimbangan data dengan SMOTE, hingga pencarian parameter terbaik melalui *GridSearchCV*. Dari hasil pengujian, Algoritma XGBoost menghasilkan model dengan performa paling baik, mencapai akurasi sebesar 0.8443. Model ini mampu membaca pola pada data teks berdimensi tinggi dengan lebih efektif dibanding dua algoritma lainnya. SVM dengan kernel RBF berada di posisi kedua dengan akurasi 0.8135, menunjukkan bahwa pendekatan non-linear tetap relevan dan kompetitif untuk analisis sentimen. Sementara itu, AdaBoost memperoleh akurasi 0.7868, yang meskipun lebih rendah, tetap memberikan gambaran yang konsisten tentang kecenderungan sentimen di media sosial. Secara keseluruhan, hasil ini menunjukkan bahwa metode boosting, terutama XGBoost, memiliki kemampuan lebih efektif dalam menangani karakteristik data teks yang kompleks. Temuan ini juga memberikan dasar yang kuat untuk penelitian lanjutan, baik dalam pengembangan model sentimen maupun pemanfaatan data media sosial sebagai sumber informasi publik yang dinamis.

DAFTAR PUSTAKA

- [1] R. Rahmatulloh, M. Iqbal Ibrahim, M. R. Handayani, K. Umam, and N. C. H. Wibowo, "Model Klasifikasi Naive Bayes untuk Pemetaan Persepsi Publik Secara Real-Time pada Media Sosial: Studi Kasus RUU TNI 2025," *Decod. J. Pendidik. Teknol. Inf.*, vol. 5, no. 2 SE-Articles, pp. 365–379, Jul. 2025, doi: 10.51454/decode.v5i2.1139.
- [2] D. P. Astuti and L. Wulandari, "Satire Media Sosial: Medium Baru Pembentukan Persepsi Politik Generasi Z," *JISPO J. Ilmu Sos. dan Ilmu Polit.*, vol. 15, no. 1 SE-, pp. 1–30, Jun. 2025, doi: 10.15575/jispo.v15i1.43980.
- [3] S. K. RI, "Satu Tahun Kabinet Merah Putih, Presiden Prabowo: "Kita Telah Bekerja Keras dan Hasilnya Dirasakan Rakyat Read more: <https://setkab.go.id/satu-tahun-kabinet-merah-putih-presiden-prabowo-kita-telah-bekerja-keras-dan-hasilnya-dirasakan-rakyat/>." [Online]. Available: <https://setkab.go.id/satu-tahun-kabinet-merah-putih-presiden-prabowo-kita-telah-bekerja-keras-dan-hasilnya-dirasakan-rakyat/>
- [4] A. I., "Satu Tahun Pemerintahan Prabowo-Gibran: Fondasi Kuat Menuju Indonesia yang Mandiri dan Sejahtera," Nov. 14, 2025, *Indonesia.go.id*. [Online]. Available: <https://indonesia.go.id/kategori/feature/10204/satu-tahun-pemerintahan-prabowo-gibran-fondasi-kuat-menuju-indonesia-yang-mandiri-dan-sejahtera?lang=1>
- [5] M. AminiMotlagh, H. Shahhoseini, and N. Fatehi, "A reliable sentiment analysis for classification of tweets in social networks," *Soc. Netw. Anal. Min.*, vol. 13, no. 1, p. 7, 2022, doi: 10.1007/s13278-022-00998-2.

- [6] L. Zhang, Q. Jiang, W. Xiong, and W. Zhao, "Evaluating the quality of digital education resources based on learners' online reviews through topic modeling and opinion mining," *Educ. Inf. Technol.*, vol. 30, no. 11, pp. 15207–15230, 2025, doi: 10.1007/s10639-025-13407-w.
- [7] M. F. Azhar and G. N. Bakry, "Pengaruh Penggunaan Twitter (X) terhadap Orientasi Pemilih Pemula pada Pemilihan Presiden 2024," *JSSH (Jurnal Sains Sos. dan Humaniora)*, vol. 9, no. 1 SE-Artikel, pp. 101–112, May 2025, doi: 10.30595/jssh.v9i1.20385.
- [8] Data Reportal, "X Pengguna, Statistik, Data, Tren, dan Lainnya — DataReportal — Wawasan Digital Global." Accessed: Oct. 13, 2025. [Online]. Available: https://datareportal.com/essential-x-stats?utm_source=DataReportal&utm_medium=Country_Article_Hyperlink&utm_campaign=Digital_2025&utm_term=Indonesia&utm_term=Indonesia&utm_content=X_Stats_Link
- [9] T. Safitri, Y. Umaidah, and I. Maulana, "Analisis Sentimen Pengguna Twitter Terhadap Grup Musik BTS Menggunakan Algoritma Support Vector Machine," *J. Appl. Informatics Comput.*, vol. 7, no. 1 SE-Articles, pp. 34–41, Jul. 2023, doi: 10.30871/jaic.v7i1.5039.
- [10] A. N. Rahmanto, "Media Sosial dan Persepsi Publik tentang Good Governance pada Pemerintah Daerah di Solo Raya," *J. Ilmu Komun.*, vol. 20, no. 1, p. 88, May 2022, doi: 10.31315/jiv20i1.6433.
- [11] M. D. Rizqi, W. A. Saputra, and M. Murahman, "Menggali Hubungan antara Media Sosial dan Kebijakan Pemerintah: Pendekatan Bibliometrik dan Tinjauan Pustaka," *J. Kawistara*, vol. 13, no. 3, p. 389, Dec. 2023, doi: 10.22146/kawistara.78591.
- [12] S. Alfataah and N. Cahyono, "Sentimen Analisis Pengguna Twitter Terhadap Provider XL Axiata Menggunakan Metode Support Vector Machine," *JATI (Jurnal Mhs. Tek. Inform.)*, 2025, doi: 10.36040/jati.v9i3.13647
- [13] N. Hadi and D. Sugarto, "Analisis Sentimen Pembangunan IKN pada Media Sosial X Menggunakan Algoritma SVM, Logistic Regression dan Naïve Bayes," *J. Inform. J. Pengemb. IT*, vol. 10, no. 1, pp. 37–49, Jan. 2025, doi: 10.30591/jpit.v10i1.7106.
- [14] I. K. Nti and A. R. Somanathan, "A Scalable RF-XGBoost Framework for Financial Fraud Mitigation," *IEEE Trans. Comput. Soc. Syst.*, vol. 11, no. 2, pp. 1556–1563, 2024, doi: 10.1109/TCSS.2022.3209827.
- [15] I. R. Hendrawan, E. Utami, and A. D. Hartanto, "Comparison of Naïve Bayes Algorithm and XGBoost on Local Product Review Text Classification," *Edumatic J. Pendidik. Inform.*, vol. 6, no. 1, pp. 143–149, 2022, doi: 10.29408/edumatic.v6i1.5613.
- [16] A. Choudhury, A. Mondal, and S. Sarkar, "Searches for the BSM scenarios at the LHC using decision tree-based machine learning algorithms: a comparative study and review of random forest, AdaBoost, XGBoost and LightGBM frameworks," *Eur. Phys. J. Spec. Top.*, vol. 233, no. 15, pp. 2425–2463, 2024, doi: 10.1140/epjs/s11734-024-01308-x.
- [17] M. R. Fauzan, S. Anraeni, and L. Budiman Ilmuwan, "Analisis Sentimen Tweet Netizen Terhadap Timnas Sepak Bola Indonesia di Era Shin Tae-Yong Menggunakan Algoritma Support Vector Machine (SVM)," *LINIER Lit. Inform. dan Komputer; Vol 2, No 3 (2025)DO* - 10.33096/linier.v2i3.3144, Oct. 2025, doi: 10.33096/linier.v2i3.3144
- [18] X. Liu, J. Sun, A. Lei, and J. Zhu, "Research and Applications of Large Language Models for Converting Unstructured Data into Structured Data," in *2024 3rd International Conference on Cloud Computing, Big Data Application and Software Engineering (CBASE)*, 2024, pp. 305–308. doi: 10.1109/CBASE64041.2024.10824634.
- [19] M. Siino, I. Tinnirello, and M. La Cascia, "Is text preprocessing still worth the time? A comparative survey on the influence of popular preprocessing methods on Transformers and traditional classifiers," *Inf. Syst.*, vol. 121, p. 102342, 2024, doi: <https://doi.org/10.1016/j.is.2023.102342>.
- [20] Supriyono, A. P. Wibawa, Suyono, and F. Kurniawan, "Advancements in natural language processing: Implications, challenges, and future directions," *Telemat. Informatics Reports*, vol. 16, p. 100173, 2024, doi: <https://doi.org/10.1016/j.teler.2024.100173>.
- [21] L. Rohmatun and A. Baita, "Machine Learning-Based Sentiment Analysis on Twitter (X): A Case Study of the 'Kabur Aja Dulu' Issue Using SVM," *J. Appl. Informatics Comput.*, vol. 9, no. 4, pp. 1972–1983, 2025, doi: 10.30871/jaic.v9i4.9991.
- [22] M. Rahardi, A. Aminuddin, F. Abdulloh, and R. Nugroho, "Sentiment Analysis of Covid-19 Vaccination using Support Vector Machine in Indonesia," *Int. J. Adv. Comput. Sci. Appl.*, vol. 13, Jan. 2022, doi: 10.14569/IJACSA.2022.0130665.
- [23] N. W. S. Saraswati, C. P. Yanti, I. D. M. K. Muku, and D. A. P. R. Dewi, "Evaluation Analysis of the Necessity of Stemming and Lemmatization in Text Classification," *MATRIK J. Manajemen, Tek. Inform. dan Rekayasa Komput.*, vol. 24, no. 2 SE-Articles, pp. 321–332, doi: 10.30812/matrik.v24i2.4833.
- [24] S. Biswas, K. Young, and J. Griffith, "A Comparison of Automatic Labelling Approaches for Sentiment Analysis," in *International Conference on Data Technologies and Applications*, 2022. doi: 10.48550/arXiv.2211.02976
- [25] G. D'Aniello, M. Gaeta, and I. La Rocca, "KnowMIS-ABSA: an overview and a reference model for applications of sentiment analysis and aspect-based sentiment analysis," *Artif. Intell. Rev.*, vol. 55, no. 7, pp. 5543–5574, 2022, doi: 10.1007/s10462-021-10134-9.
- [26] M. I. Arif Chandra and R. Yusuf, "Visualisasi Kata Kunci Pemberitaan Pemilu 2024 Menggunakan Scapy Dan Wordcloud," *Tek. Teknol. Inf. dan Multimed.*, vol. 5, no. 1 SE-Articles, pp. 41–46, Jun. 2024, doi: 10.46764/teknimedia.v5i1.187.
- [27] M. R. Ningsih and J. Unjung, "Sentiment Analysis on SocialMedia Using TF-IDF Vectorization and H2O Gradient Boosting for Student Anxiety Detection," *Sci. J. Informatics*, vol. 11, no. 4, pp. 1137–1144, 2025, doi: 10.15294/sji.v12i1.20582.
- [28] G. Tamami, W. A. Triyanto, and S. Muzid, "Sentiment Analysis Mobile JKN Reviews Using SMOTE Based LSTM," *IJCCS (Indonesian J. Comput. Cybern. Syst.)*, vol. 19, no. 1, p. 13, 2025, doi: 10.22146/ijccs.101910.
- [29] H. Bichri, A. Chergui, and M. Hain, "Investigating the Impact of Train / Test Split Ratio on the Performance of Pre-Trained Models with Custom Datasets," *Int. J. Adv. Comput. Sci. Appl.*, 2024, doi: 10.14569/IJACSA.2024.0150235
- [30] J. T. Kumalasari and I. Puspitorini, "Perbandingan Metode Klasifikasi dan SMOTE Terhadap Analisa Sentimen Mobil Listrik Indonesia," *J. Minfo Polgan*, vol. 13, no. 2 SE-, pp. 2257–2268, Jan. 2025, doi: 10.33395/jmp.v13i2.14428.
- [31] F. Izhari, "Pemodelan Perilaku Penggunaan Media Sosial Mahasiswa dengan Algoritma XGBoost," *J. Minfo Polgan*, vol. 14, no. 1, pp. 1303–1308, 2025, doi: 10.33395/jmp.v14i1.15030.
- [32] M. J. Paput, K. Suryowati, and M. T. Jatipaningrum, "Perbandingan Metode Random Forest Dan Adaptive Boosting Pada Klasifikasi Indeks Pembangunan Manusia Di Indonesia," *J. Stat. Ind. dan Komputasi*, vol. 8, no. 2, pp. 73–83, 2023, doi: 10.34151/statistika.v8i2.4458.
- [33] S. Mulyani and T. Arifin, "Prediksi Kelangsungan Hidup Pasien Gagal Jantung Menggunakan Pendekatan Machine Learning dengan Optimasi GridSearchCV," *J. Inform. Polinema*, vol. 11, no. 4 SE-Articles, pp. 577–586, Aug. 2025, doi: 10.33795/jip.v11i4.7938.
- [34] W. Nugraha and A. Sasongko, "Hyperparameter Tuning on Classification Algorithm with Grid Search," *Sistemasi*, vol. 11, no. 2, p. 391, 2022, doi: 10.32520/stmsi.v11i2.1750.