

Analisis Sentimen Pengguna Coretax Pada Media Sosial menggunakan *Decision Tree* dan *Adaptive Boosting*

Nur Safika¹, Elvi Rahmi²

^{1,2} Program Studi Rekayasa Perangkat Lunak- Politeknik Negeri Bengkalis

Jalan Bathin Alam Sei Alam – Bengkalis - Indonesia

nursafikaefendi@gmail.com, elvirahmi@polbeng.ac.id

Abstrak— Implementasi sistem administrasi perpajakan digital terbaru di Indonesia, Coretax, memicu beragam opini dan keluhan dari masyarakat di media sosial. Komentar publik dalam jumlah besar dan tidak terstruktur menyulitkan pihak otoritas dalam mengevaluasi layanan secara manual. Penelitian ini bertujuan melakukan analisis sentimen terhadap komentar pengguna Coretax di platform X (Twitter) dan Instagram untuk mengelompokkan opini ke dalam kategori positif, negatif, dan netral. Metode yang digunakan adalah algoritma Decision Tree yang dioptimasi dengan teknik ensemble learning Adaptive Boosting (AdaBoost). Tahapan penelitian meliputi web scraping, preprocessing teks (termasuk normalisasi slang dan negation merging), pembobotan kata menggunakan TF-IDF, hingga evaluasi model. Dataset yang digunakan berjumlah 1.414 data setelah melalui proses augmentasi untuk menjaga keseimbangan kelas. Hasil pengujian menunjukkan bahwa optimasi AdaBoost berhasil meningkatkan akurasi model sebesar 12,18% dari baseline Decision Tree, dengan akurasi akhir mencapai 82,23% pada data uji dan rata-rata akurasi cross validation sebesar 83,74%. Sistem diimplementasikan dalam aplikasi berbasis web menggunakan arsitektur hybrid Laravel-Flask untuk visualisasi hasil analisis secara interaktif. Penelitian ini diharapkan membantu Direktorat Jenderal Pajak (DJP) memahami keluhan pengguna secara real-time untuk perbaikan kualitas layanan perpajakan digital.

Kata Kunci— Analisis Sentimen, Coretax, Pohon Keputusan, Adaptive Boosting, Media Sosial

Abstract— The implementation of Indonesia's newest digital tax administration system, Coretax, has triggered diverse public opinions and complaints on social media. Large and unstructured public comments make it difficult for authorities to evaluate services manually. This study aims to conduct sentiment analysis on Coretax user comments on X (Twitter) and Instagram platforms to classify opinions into positive, negative, and neutral categories. The method used is the Decision Tree algorithm optimized with the Adaptive Boosting (AdaBoost) ensemble learning technique. The research stages include web scraping, text preprocessing (including slang normalization and negation merging), word weighting using TF-IDF, and model evaluation. The dataset used consists of 1,414 data after going through an augmentation process to maintain class balance. The test results show that AdaBoost optimization successfully increased the model accuracy by 12.18% from the baseline Decision Tree, reaching a final accuracy of 82.23% on the test data and an average cross-validation accuracy of 83.74%. The system is implemented in a web-based

application using a hybrid Laravel-Flask architecture for interactive visualization of analysis results. This research is expected to help the Directorate General of Taxes (DJP) understand user complaints in real-time to improve digital tax service quality.

Keywords – Sentiment Analysis, Coretax, Decision Tree, Adaptive Boosting, Social Media

I. PENDAHULUAN

Perkembangan teknologi informasi yang sangat pesat telah membawa perubahan besar dalam kehidupan manusia modern. Teknologi kini menjadi bagian yang tidak terpisahkan dari berbagai aktivitas, terutama dalam hal komunikasi dan pertukaran informasi. Salah satu platform yang memegang peran sentral dalam komunikasi digital adalah media sosial X (sebelumnya Twitter) dan Instagram. Platform ini dimanfaatkan oleh masyarakat untuk menyampaikan berbagai opini, saran, maupun keluhan terhadap berbagai isu publik secara terbuka. Indonesia merupakan salah satu negara dengan jumlah pengguna media sosial aktif terbesar di dunia. Menurut laporan Digital 2025 Global Overview Report, terdapat sekitar 143 juta identitas pengguna media sosial aktif di Indonesia pada awal tahun 2025, yang setara dengan 50,2% dari total populasi [1]. Tingginya aktivitas digital ini menjadikan media sosial sebagai instrumen yang sangat representatif untuk mengukur kepuasan dan respons masyarakat terhadap suatu kebijakan atau sistem layanan yang diluncurkan oleh pemerintah.

Salah satu isu publik yang memicu diskusi intensif di media sosial adalah implementasi Coretax System. Coretax merupakan sistem administrasi perpajakan digital terbaru yang diluncurkan oleh Direktorat Jenderal Pajak (DJP) Kementerian Keuangan Republik Indonesia pada awal tahun 2025. Sistem ini dirancang untuk mengintegrasikan seluruh proses bisnis inti administrasi perpajakan, mulai dari pendaftaran wajib pajak, pelaporan Surat Pemberitahuan (SPT), hingga pembayaran pajak secara online dengan tujuan meningkatkan efisiensi dan kepatuhan sukarela wajib pajak [2].

Namun, dalam masa awal implementasinya, sistem Coretax menghadapi berbagai kendala teknis dan operasional yang dikeluhkan oleh pengguna di media sosial. Beberapa kendala yang sering dilaporkan meliputi lambatnya respons server, kegagalan saat mengunggah SPT PPN maupun *e-faktur*, *server down* pada awal bulan saat aktivitas pelaporan tinggi, hingga ketidaksesuaian data wajib pajak dengan sistem lama [3], [4]. Keluhan-keluhan ini tersebar luas dalam bentuk komentar teks

tidak terstruktur dengan volume yang sangat besar, sehingga menyulitkan DJP dalam mengevaluasi efektivitas layanan secara manual [5].

Untuk mengatasi tantangan tersebut, penelitian ini menerapkan teknik Natural Language Processing (NLP) berupa analisis sentimen berbasis *machine learning*. Algoritma klasifikasi yang digunakan adalah Decision Tree, yang dikenal karena kemampuannya memodelkan aturan keputusan dalam bentuk pohon terstruktur yang mudah diinterpretasikan. Guna memitigasi kelemahan bawaan Decision Tree seperti kerentanan terhadap *overfitting* dan ketidakstabilan pada data kompleks, metode dioptimasi menggunakan teknik ensemble learning Adaptive Boosting (AdaBoost). Metode AdaBoost bekerja dengan menggabungkan beberapa pohon keputusan sederhana (*weak learners*) secara iteratif untuk membentuk satu model klasifikasi yang kuat (*strong learner*) [6].

Penelitian ini memberikan kontribusi kebaruan melalui dua aspek utama. Pertama, penerapan teknik negation merging pada tahap preprocessing teks bahasa Indonesia untuk mempertahankan konteks kata negasi (misalnya mengubah 'tidak error' menjadi 'tidak_error') agar tidak hilang pada pembobotan kata TF-IDF. Kedua, implementasi model ke dalam aplikasi berbasis web menggunakan arsitektur hybrid Laravel-Flask. Backend Laravel bertindak sebagai portal interaktif, sedangkan Flask API menangani proses web scraping menggunakan Selenium dan eksekusi model *machine learning* secara *real-time*. Melalui pendekatan ini, hasil analisis sentimen dapat divisualisasikan secara interaktif bagi pemangku kebijakan untuk perbaikan layanan.

Beberapa penelitian terdahulu yang mendasari optimasi ini antara lain dilakukan oleh Qadrini et al. (2021) yang mengombinasikan Decision Tree dan AdaBoost untuk klasifikasi dengan akurasi mencapai 95% [6]. Septiawan & Chairani (2023) menggunakan Decision Tree dan SVM yang dioptimasi dengan AdaBoost untuk deteksi ujaran kebencian di Twitter dengan hasil performa meningkat signifikan dari base model [7]. Faizi & Nugroho (2023) menerapkan AdaBoost tunggal untuk sentimen kenaikan harga BBM dengan akurasi 86,8% [8]. Sementara itu, Maharani & Fathoni (2024) menganalisis sentimen pengguna PayPal dengan Decision Tree tunggal dan mencapai akurasi 97,33% [9].

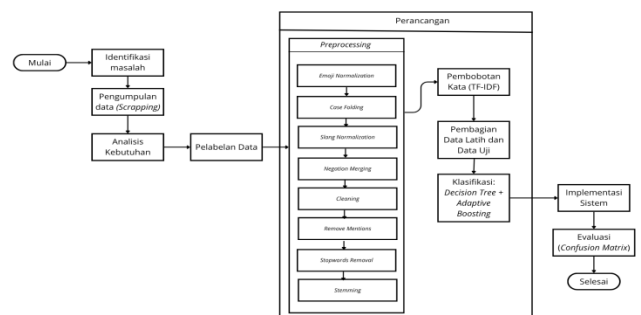
II. METODE PENELITIAN

Metodologi penelitian ini mengadopsi pendekatan terstruktur untuk memastikan validitas dan reproduktibilitas hasil analisis sentimen. Prosedur penelitian terdiri atas beberapa tahap utama, yaitu:

- Identifikasi Masalah: Merumuskan masalah terkait kendala operasional sistem Coretax di media sosial.
- Akuisisi Data (Scraping): Mengumpulkan opini dari media sosial X dan Instagram menggunakan Selenium Python berdasarkan kata kunci 'Coretax'.

- Analisis Kebutuhan Sistem: Merancang fungsionalitas sistem web analisis sentimen menggunakan *Use Case Diagram*.
- Pelabelan Data: Mengelompokkan data ulasan ke dalam kategori sentimen positif, negatif, dan netral melalui proses semi-otomatis serta menerapkan targeted data augmentation.
- Tahapan Preprocessing Data: Melakukan penyaringan teks secara mendalam melalui delapan urutan langkah: *Emoji Normalization*, *Case Folding*, *Slang Normalization*, *Negation Merging*, *Cleaning*, *Remove Mentions*, *Stopwords Removal*, dan *Stemming* (Detail dijelaskan pada Poin B).
- Pembobotan Kata (TF-IDF): Menghitung nilai pembobotan kata menggunakan frekuensi kemunculan kata dalam dokumen.
- Pembagian Data Latih dan Data Uji: Membagi dataset menggunakan teknik pembagian data terstratifikasi (*stratified split*).
- Pelatihan Model (Decision Tree + Adaptive Boosting): Membangun klasifikasi ensemble secara iteratif dengan menggabungkan pohon keputusan dasar (*weak learners*).
- Evaluasi Model: Menguji performa hasil klasifikasi akhir model menggunakan hold-out set (Akurasi, Presisi, Recall, F1-Score) serta pengujian stabilitas model dengan Stratified K-Fold Cross-Validation (k=5).
- Implementasi Sistem: Menerapkan model ke dalam aplikasi berbasis web menggunakan arsitektur hybrid Laravel-Flask.

Tahapan penelitian secara keseluruhan ditunjukkan pada Gambar 1.



Gbr. 1 Prosedur Penelitian

A. Akuisisi Data dan Perangkat Alat

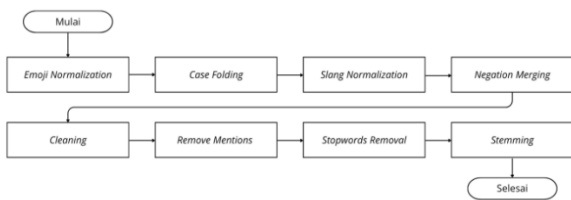
Penelitian ini mengumpulkan data opini dari media sosial X dan Instagram melalui teknik web scraping menggunakan Selenium Python berdasarkan kata kunci 'Coretax' [10]. Tahap pembersihan dilakukan untuk mengeliminasi duplikasi, spam, dan ulasan kosong guna menghasilkan dataset yang valid. Spesifikasi perangkat yang digunakan mencakup laptop Lenovo Thinkpad T470 (Intel Core i5, RAM 16GB), bahasa pemrograman Python 3.14, VS Code, basis data MySQL, serta framework Laravel 10 sebagai antarmuka. Penggunaan Selenium dipilih karena efektivitasnya dalam ekstraksi otomatis pada situs web dinamis sesuai tren penelitian data teks daring saat ini [11].

B. Tahapan Preprocessing Data

Tahap preprocessing teks terdiri atas delapan langkah terstruktur:

- a. *Emoji Normalization* mengubah emoji menjadi teks deskriptif,
- b. *Case Folding* menyamakan huruf menjadi kecil,
- c. *Slang Normalization* mengubah kata tidak baku menggunakan kamus kamus gaul,
- d. *Negation Merging* menggabungkan kata negasi (tidak, kurang, gak) dengan kata setelahnya,
- e. *Cleaning* menghilangkan tanda baca, URL, dan angka,
- f. *Remove Mentions* menghapus username X,
- g. *Stopwords Removal* menghapus kata umum yang tidak penting kecuali kata negasi, dan
- h. *Stemming* mengubah kata berimbuhan menjadi kata dasar menggunakan algoritma Sastrawi.

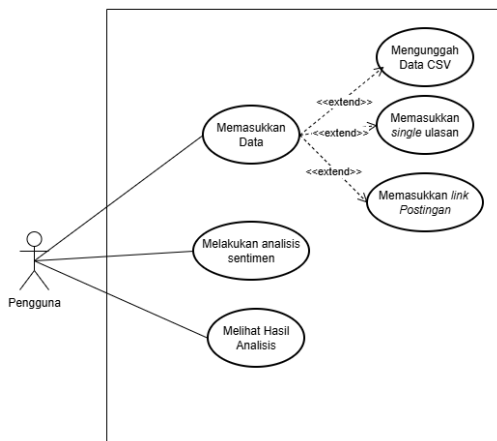
Alur preprocessing ditunjukkan pada Gambar 2.



Gbr. 2 Alur Preprocessing Teks

C. Perancangan Sistem Hybrid

Perancangan sistem web analisis sentimen menggunakan Use Case Diagram untuk menggambarkan interaksi aktor dengan sistem. Aktor utama (Pengguna/DJP) dapat mengunggah file CSV, menginputkan teks ulasan tunggal, menginputkan URL postingan X untuk di-scraping langsung, mengeksekusi analisis sentimen, serta melihat visualisasi grafik persentase sentimen positif, negatif, dan netral. Diagram use case sistem disajikan pada Gambar 3.



Gbr. 3 Use Case Diagram Sistem

D. Formulasi Matematis Pembobotan dan Model

Metode Term Frequency-Inverse Document Frequency (TF-IDF) digunakan untuk pembobotan kata dengan

menghitung frekuensi kemunculan kata dalam dokumen (TF) dikalikan dengan bobot kebalikan frekuensi dokumen (IDF) [12], seperti ditunjukkan pada Persamaan (1) dan (2):

$$IDF(w) = \log\left(\frac{N}{DF(w)}\right) \quad (1)$$

$$TF-IDF(w, d) = TF(w, d) \times IDF(w) \quad (2)$$

Di mana N merupakan total dokumen dan $DF(w)$ adalah jumlah dokumen yang mengandung kata w .

Algoritma Decision Tree membangun pohon klasifikasi dengan mencari atribut pemisah terbaik berdasarkan nilai Entropy terendah atau Information Gain tertinggi [10], yang dirumuskan pada Persamaan (3) dan (4):

$$Entropy(S) = \sum_{i=1}^c P_i \cdot \log_2(P_i) \quad (3)$$

$$Gain(S, A) = entropy(S) - \sum_{v \in Values(A)} \left(\frac{|S_v|}{|S|} \cdot Entropy(S_v) \right) \quad (4)$$

Di mana S adalah himpunan kasus, A adalah atribut, dan P_i adalah proporsi kelas ke- i pada dataset.

Metode Adaptive Boosting (AdaBoost) meningkatkan akurasi klasifikasi dengan membangun klasifikasi ensemble secara iteratif. Bobot awal setiap data diinisialisasi sama rata, seperti pada Persamaan (5):

$$W_1(i) = \frac{1}{N} \quad (5)$$

Setiap pohon ke- m (*weak learner*) dilatih pada data terbobot dan tingkat kesalahan epsilon dihitung dengan rumus pada Persamaan (6):

$$\epsilon_m = \frac{\sum_{i=1}^N W_m(i) \cdot I(y_i \neq H_m(x_i))}{\sum_{i=1}^N W_m(i)} \quad (6)$$

Bobot kontribusi pohon α_m ditentukan berdasarkan tingkat kesalahan pohon tersebut, seperti pada Persamaan (7):

$$\alpha_m = 0.5 \cdot \ln\left(\frac{1 - \epsilon_m}{\epsilon_m}\right) \quad (7)$$

Bobot data diperbarui pada iterasi berikutnya untuk memberikan bobot lebih tinggi pada data yang salah diklasifikasi, dirumuskan pada Persamaan (8):

$$W_{m+1}(i) = \frac{W_m(i) \cdot \exp(-\alpha_m \cdot y_i \cdot H_m(x_i))}{Z_m} \quad (8)$$

Di mana Z_m adalah faktor normalisasi. Klasifikasi akhir ensemble merupakan hasil voting terbobot dari seluruh pohon keputusan penyusun, sebagaimana ditunjukkan pada Persamaan (9):

$$H(x) = \text{sign}\left(\sum_{m=1}^M \alpha_m \cdot H_m(x)\right) \quad (9)$$

III. HASIL DAN PEMBAHASAN

A. Analisis Distribusi Eksperimen Data

Hasil pengumpulan data eksternal awal dari platform X dan Instagram menghasilkan 694 data ulasan valid. Distribusi kelas sentimen menunjukkan ketidakseimbangan (Class Imbalance) yang signifikan di mana kelas positif mendominasi (46,1%), diikuti kelas netral (37,5%), dan kelas negatif (16,4%). Untuk mengatasinya, diterapkan targeted data augmentation berupa penambahan 30 data manual khusus pola negasi dan oversampling positif sebanyak 2 kali lipat. Setelah augmentasi, total dataset mencapai 984 data, dengan data latih teraugmentasi sebesar 984 data yang digunakan untuk pemodelan. Distribusi lengkap data disajikan pada Tabel 1.

TABEL I

DISTRIBUSI DATA SEBELUM DAN SESUDAH AUGMENTASI

Kelas Sentimen	Data Awal	Persentase Awal	Data Teraugmentasi (Latih)	Persentase Akhir
Positif	320	46.1%	696	70.7%
Netral	260	37.5%	264	26.8%
Negatif	114	16.4%	126	12.8%
Total	694	100.0%	984	100.0%

B. Detail Hasil Transformasi Tahapan Preprocessing Teks

Hasil preprocessing teks memperlihatkan efektivitas pembersihan data yang sangat baik. Untuk melihat bagaimana satu data kalimat berubah secara detail pada setiap langkah preprocessing (dari data awal hingga stemming), berikut disajikan simulasi runtutannya pada Tabel 2.

TABEL II

URUTAN TRANSFORMASI DATA PADA TAHAPAN PREPROCESSING

No	Tahapan Preprocessing	Kondisi Kalimat Setelah tahapan	Keterangan Perubahan Data
1	Teks Asli (Raw Data)	RT @DJP_Pajak: Coretax error terus 🤔 gak bisa akses e-faktur!	Teks awal hasil penarikan (<i>scraping</i>).
2	Emoji Normalization	RT @DJP_Pajak: Coretax error terus [sedih] gak bisa akses e-faktur!	Karakter emoji 🤔 diubah menjadi kata
3	Case Folding	rt @djp_pajak: coretax error terus [sedih] gak bisa akses e-faktur!	Mengubah semua huruf besar menjadi huruf kecil.
4	Slang Normalization	rt @djp_pajak: coretax gangguan terus [sedih] tidak bisa akses e-faktur!	Mengubah kata slang (<i>error</i> → <i>gangguan</i> , <i>gak</i> → <i>tidak</i>).
5	Negation Merging	rt @djp_pajak: coretax gangguan terus [sedih] tidak bisa akses e-faktur!	Menggabungkan kata negasi <i>tidak</i> dan <i>bisa</i> .

No	Tahapan Preprocessing	Kondisi Kalimat Setelah tahapan	Keterangan Perubahan Data
6	Cleaning	rt djp_pajak coretax gangguan terus sedih tidak bisa akses efaktur	Menghapus tanda baca, tanda kurung, simbol, dan strip.
7	Remove Mentions	rt coretax gangguan terus sedih tidak bisa akses efaktur	Menghapus tagar/nama akun username @djp_pajak.
8	Stopwords Removal	ganggu terus sedih tidak bisa akses	Menghapus stopwords non-esensial (rt, coretax, efaktur).
9	Stemming (Akhir)	ganggu terus sedih tidak bisa akses	Mengubah kata berimbuhan menjadi kata dasar.

C. Pengaruh Preprocessing Negation Merging (Ablation Study)

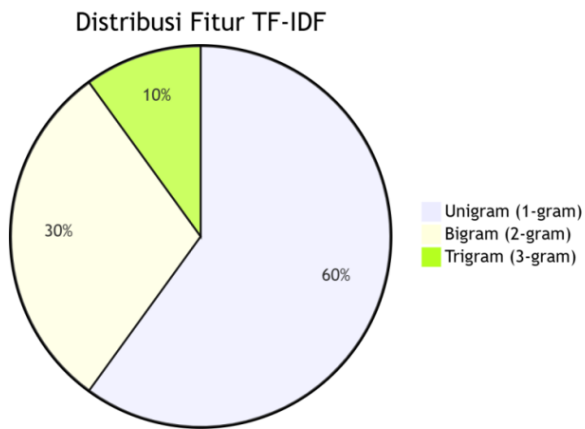
Penerapan negation merging berdampak signifikan terhadap kemampuan model menangkap konteks kalimat negasi. Eksperimen ablation study membandingkan performa model dengan dan tanpa negation merging pada hold-out set. Hasil perbandingan menunjukkan bahwa negation merging mampu menaikkan akurasi sebesar 2,38% (dari 79,85% menjadi 82,23%) dan secara dramatis menaikkan recall kelas minoritas negatif sebesar 7,9%. Hasil lengkap ablation disajikan pada Tabel 3.

TABEL III

DAMPAK PREPROCESSING NEGATION MERGING

Metrik Evaluasi	Tanpa Negation Merging	Dengan Negation Merging	Peningkatan (Delta)
Akurasi (<i>Accuracy</i>)	79.85%	82.23%	+2.38%
Presisi (<i>Precision - Negatif</i>)	72.4%	79.7%	+7.30%
Recall (<i>Recall - Negatif</i>)	68.2%	76.1%	+7.90%
F1-Score (F1 - Negatif)	70.2%	77.8%	+7.60%

Distribusi fitur TF-IDF setelah SelectKBest Chi-Square menyisakan 500 fitur paling berpengaruh. Kata negasi hasil merger seperti 'tidak_error', 'tidak_lambat', dan 'gak_rumit' memiliki bobot penting tinggi dalam vocabulary model. Visualisasi distribusi fitur TF-IDF disajikan pada Gambar 4.



Gbr. 4 Distribusi Fitur TF-IDF

D. Tuning Hyperparameter dan Evaluasi Performa

Proses tuning hyperparameter menggunakan teknik *GridSearchCV* dengan 5-fold cross validation. Hasil parameter terbaik yang dipilih untuk model final AdaBoost disajikan pada Tabel 4.

TABEL IV
HYPERPARAMETER TERBAIK HASIL TUNING

Hyperparameter	Nilai Terbaik	Range Pencarian
Base Estimator Max Depth	15	[5, 10, 15]
Number of Estimators (n_estimators)	50	[50, 100]
Learning Rate	0.5	[0.1, 0.5, 0.8]
Boosting Algorithm	SAMME	[SAMME, SAMME.R]

Evaluasi akhir dilakukan dengan membandingkan baseline model Decision Tree tunggal terhadap model Decision Tree yang dioptimasi dengan AdaBoost. Hasil pengujian hold-out set menunjukkan bahwa AdaBoost berhasil meningkatkan akurasi model secara signifikan sebesar 12,18% (dari 70,05% menjadi 82,23%) dan meningkatkan F1-Score sebesar 11,45%. Perbandingan performa komprehensif disajikan pada Tabel 5.

TABEL V
PERBANDINGAN PERFORMA KLASIFIKASI MODEL

Metrik Evaluasi	Decision Tree Baseline	AdaBoost Ensemble	Peningkatan (Delta)
Akurasi (Accuracy)	70.05%	82.23%	+12.18%
Presisi (Macro Precision)	70.91%	79.69%	+8.78%
Recall (Macro Recall)	70.34%	79.07%	+8.73%
F1-Score (Macro F1)	67.83%	79.28%	+11.45%

Pengujian stabilitas model dilakukan menggunakan Stratified K-Fold Cross-Validation (k=5) untuk menjamin

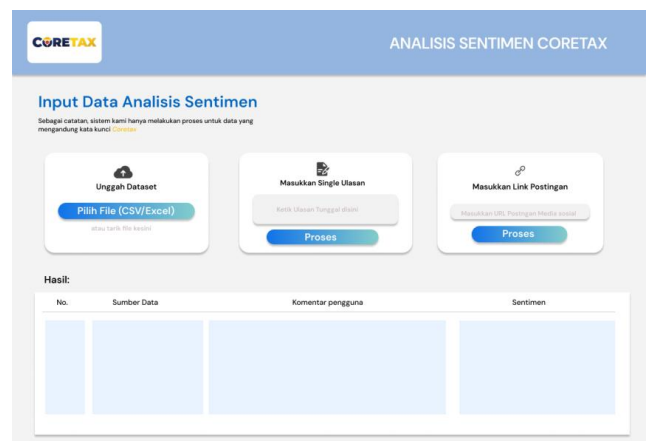
performa model stabil dan tidak overfitting. Rata-rata akurasi cross-validation yang diperoleh mencapai 83,74% dengan deviasi standar $\pm 3,49\%$, yang menunjukkan konsistensi kinerja yang tinggi. Hasil per fold disajikan pada Tabel 6.

TABEL VI
HASIL PENGUJIAN STRATIFIED K-FOLD CROSS-VALIDATION

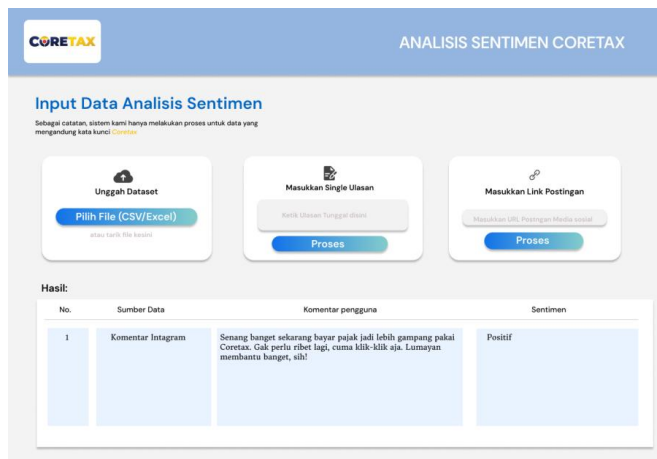
Fold Ke-	Data Latih	Data Uji	Akurasi (Accuracy)
Fold 1	787 sampel	197 sampel	84.26%
Fold 2	787 sampel	197 sampel	85.79%
Fold 3	787 sampel	197 sampel	82.23%
Fold 4	787 sampel	197 sampel	81.22%
Fold 5	787 sampel	197 sampel	85.20%
Rata-rata (Mean)	-	-	83.74%

E. Implementasi Aplikasi Web Interaktif

Sistem diimplementasikan ke dalam aplikasi berbasis web dengan arsitektur hybrid Laravel-Flask. Antarmuka web Laravel menyediakan form input teks tunggal dan upload file CSV, sedangkan Flask API bertindak sebagai backend processing engine. Gambar 5 memperlihatkan antarmuka input ulasan oleh pengguna, dan Gambar 6 memperlihatkan hasil analisis sentimen dalam bentuk representasi visual interaktif.



Gbr. 5 Tampilan Halaman Input Ulasan



Gbr. 6 Tampilan Visualisasi Hasil Analisis Sentimen

IV. KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan hasil penelitian dan pembahasan yang telah dipaparkan, dapat disimpulkan bahwa penelitian ini telah berhasil mengimplementasikan sistem analisis sentimen terhadap komentar pengguna Coretax dengan menggunakan algoritma *Decision Tree* yang dioptimasi melalui pendekatan *AdaBoost Ensemble*. Penerapan optimasi model ensemble tersebut terbukti mampu meningkatkan performa akurasi secara signifikan sebesar 12,18% dibandingkan dengan model tunggal *baseline*, hingga mencapai nilai akhir sebesar 82,23% pada *hold-out data* dan rata-rata *cross-validation* sebesar 83,74%. Selain itu, integrasi rekayasa fitur berupa teknik *preprocessing negation merging* juga sukses menaikkan akurasi sistem sebesar 2,38% serta meningkatkan sensitivitas (*recall*) pada kelas minoritas negatif secara dramatis sebesar 7,9%. Sebagai implementasi praktis, aplikasi berbasis web dengan arsitektur *hybrid* Laravel-Flask yang dikembangkan dalam penelitian ini terbukti dapat berjalan dengan lancar dalam menyajikan berbagai fitur utama, meliputi pemrosesan teks tunggal, *web scraping*, serta visualisasi data secara interaktif.

B. Saran

Penelitian selanjutnya dapat mengembangkan model menggunakan algoritma berbasis *Deep Learning* seperti LSTM, Bi-LSTM, maupun Transformer. Selain itu, cakupan data dapat diperluas ke berbagai platform media sosial lainnya agar hasil analisis lebih representatif.

DAFTAR PUSTAKA

- [1] A. T. Haryanto, "Daftar Jumlah Pengguna Media Sosial Indonesia Terbaru, Siapa yang Paling Banyak?," detikinet (detikcom). [Online]. Available: <https://inet.detik.com/cyberlife/d-7854501/daftar-jumlah-pengguna-media-sosial-indonesia-terbaru-siapa-yang-paling-banyak>
- [2] N. A. A. Wati, "Coretax: Sistem Canggih Tingkatkan Kepatuhan Sukarela." [Online]. Available: <https://www.pajak.go.id/id/artikel/coretax-sistem-canggih-tingkatkan-kepatuhan-sukarela>
- [3] Hukumonline, "Problematika Coretax, Sistem Baru Pelaporan Pajak yang Tambah Rumit," Hukumonline.com. [Online]. Available: <https://www.hukumonline.com/berita/a/problematika-coretax--sistem-baru-pelaporan-pajak-yang-tambah-rumit-lt6789cc8378d57/>
- [4] A. Al Hasan, "Bermasalah Sejak Awal Tahun, Dirjen Pajak Janji Perbaikan Coretax Selesai Juli 2025," Tempo.co. [Online]. Available: <https://www.tempo.co/ekonomi/bermasalah-sejak-awal-tahun-dirjen-pajak-janji-perbaikan-coretax-selesai-juli-2025-1364500>
- [5] A. Hamzah and R. Y. Ariyana, "Sentiment Analysis on MyPertamina Application Reviews Using NBC and SVM with Negation Handling," vol. 09, no. 08, pp. 4935–4942, 2024, doi: 10.47191/etj/v9i08.44.
- [6] L. Qadrini, A. Sepperwali, and A. Aina, "Decision Tree Dan Adaboost Pada Klasifikasi Penerima Program Bantuan Sosial," *J. Inov. Penelit.*, vol. 2, no. 7, pp. 1959–1966, 2021.
- [7] Y. Septiawan and Chairani, "Perbandingan Akurasi Metode Deteksi Ujaran Kebencian dalam Postingan Twitter Menggunakan Metode SVM dan Decision Trees yang Dioptimalkan dengan Adaboost," *J. Tek.*, vol. 17, no. 2, pp. 297–299, 2023.
- [8] A. Wahyu *et al.*, "Penerapan Metode Adaptive Boosting Pada Analisis Sentimen Kenaikan BBM Pertamina," *J. Tek. Inform. Unika ST. Thomas*, vol. 8, no. 2, pp. 2657–1501, 2023.
- [9] Maharani and Fathoni, "User Sentiment Analysis on Factors Affecting PayPal Usage Using the Decision Tree Method," *J. Ilm. Teknol. Inf. Asia*, vol. 18, no. 1, pp. 71–83, 2024.
- [10] A. Ulfah and I. Najiah, "Implementasi Web Scraping Pada Situs Jurnal Sinta Menggunakan Framework Selenium Webdriver Python," *JIKA (Jurnal Inform.)*, vol. 7, no. 1, p. 29, 2023, doi: 10.31000/jika.v7i1.7037.
- [11] F. Akbar, "Comparison of Naïve Bayes , Support Vector Machine , and Decision Tree Algorithms in Analyzing Public Opinion Sentiments on COVID-19 Vaccination in Indonesia," vol. 6, no. 1, pp. 8–17, 2023.
- [12] C. Cahyaningtyas, Y. Nataliani, and I. R. Widiyarsi, "Analisis Sentimen Pada Rating Aplikasi Shopee Menggunakan Metode Decision Tree Berbasis SMOTE," *Aiti*, vol. 18, no. 2, pp. 173–184, 2021, doi: 10.24246/aiti.v18i2.173-184.