

Analisis Sentimen Ulasan Aplikasi M-Pajak pada Google Play Store Menggunakan XGboost

Theresia Aurelly Claudia Budiarto¹, Budi Arif Dermawan², Mohamad Jajuli³

^{1,2,3}Informatika, Fakultas Ilmu Komputer, Universitas Singaperbangsa Karawang

¹2110631170038@student.unsika.ac.id; ²budi.arif@staff.unsika.ac.id; ³mohamad.jajuli@staff.unsika.ac.id
Jl. HS.Ronggo Waluyo, Puseurjaya, Telukjambe Timur, Karawang, Jawa Barat 41361, Telp. (0267) 641177

Keywords:

*Sentiment Analysis,
M-Pajak, KDD,
XGBoost.*

Abstract

In this digital era, the M-Pajak application serves as an innovation by the Indonesian government aimed at facilitating taxpayers in fulfilling their tax obligations online. Although it has been downloaded more than one million times, the application still faces various technical issues, as reflected in its low rating on the Google Play Store. This study aims to identify the features that received the most negative feedback from users. Based on the analysis results, the features that received the most criticism are related to the login process and efin verification. This indicates that technical and administrative issues remain the main obstacles in users' experience with the application. The method used is Knowledge Discovery in Database (KDD) with the XGBoost algorithm as the classification model. Using Stratified K-Fold for data splitting, the model achieved an accuracy of 95%, and when tested on new data different from the training data, it achieved an accuracy of 93%. It is recommended that the M-Pajak application developers improve the login, verification, and efin submission systems to enhance user experience and minimize technical issues.

Kata Kunci:

*Analisis Sentimen,
M-Pajak, KDD,
XGBoost.*

Abstrak

Pada era digital ini, aplikasi M-Pajak menjadi inovasi pemerintah Indonesia yang bertujuan memudahkan wajib pajak dalam memenuhi kewajiban perpajakan secara online. Meskipun telah diunduh lebih dari satu juta kali, aplikasi ini masih menghadapi berbagai kendala teknis yang tercermin dari rendahnya rating di Google Play Store. Penelitian ini bertujuan untuk mengidentifikasi fitur-fitur yang paling banyak mendapat kritik negatif dari para pengguna. Berdasarkan hasil analisis, fitur yang paling banyak menerima kritik negatif yaitu terkait proses login dan verifikasi efin. Hal ini menunjukkan bahwa kendala teknis dan administratif masih menjadi hambatan utama dalam pengalaman pengguna aplikasi. Metode yang digunakan adalah Knowledge Discovery in Database (KDD) dengan algoritma XGBoost sebagai model klasifikasi. Dengan menggunakan Stratified K-Fold untuk membagi data, model memperoleh akurasi sebesar 95%, dan saat diuji menggunakan data baru yang berbeda dari data pelatihan, model menghasilkan akurasi sebesar 93%. Pengembang aplikasi M-Pajak disarankan melakukan perbaikan pada sistem login, verifikasi, dan pengajuan efin agar pengalaman pengguna menjadi lebih baik dan kendala teknis dapat diminimalkan.

1. Pendahuluan

Pada era digital ini, pemerintah Indonesia melalui Direktorat Jenderal Pajak (DJP) melakukan inovasi untuk mempermudah wajib pajak dalam memenuhi kewajiban perpajakannya dengan mengoptimalkan penggunaan teknologi informasi [1]. Aplikasi M-Pajak merupakan salah satu inovasi pemerintah yang diluncurkan bersamaan dengan peringatan Hari Pajak pada tahun 2021 [2]. Hingga saat ini, aplikasi ini telah diunduh lebih

dari satu juta kali yang mencerminkan tingginya antusiasme masyarakat dalam memanfaatkan teknologi digital untuk memenuhi kebutuhan perpajakan. Aplikasi M-Pajak memegang peranan penting dalam mendukung modernisasi layanan perpajakan di Indonesia dengan memberikan fleksibilitas lebih, karena dapat diakses kapan saja dan di mana saja melalui perangkat seluler yang dilengkapi dengan antarmuka yang sederhana dan responsif. Selain itu, aplikasi ini juga memiliki fitur pengingat otomatis terkait tenggat waktu pembayaran pajak atau pelaporan SPT yang tidak tersedia di situs web, sehingga dapat membantu wajib pajak menghindari keterlambatan atau sanksi administrasi [3].

Namun, tidak jarang pengguna menghadapi kendala dalam menggunakan aplikasi M-Pajak. Hal ini dibuktikan berdasarkan laporan Redaksi DDTC News yang mengungkap adanya masalah pada proses login aplikasi tersebut [4]. Kendala ini juga tercermin dalam rating aplikasi M-Pajak di Google Play Store, yang hanya memperoleh rating 2 dari 5 bintang. Rating rendah ini tidak hanya menunjukkan adanya ketidakpuasan pengguna akibat berbagai kendala teknis, tetapi juga menggambarkan persepsi kurang baik terhadap kualitas aplikasi secara keseluruhan. Ulasan negatif yang dominan dapat memengaruhi persepsi calon pengguna baru dan menurunkan minat mereka untuk mencoba aplikasi [5]. Akibatnya kepercayaan masyarakat terhadap layanan perpajakan digital pemerintah dapat menurun dan menghambat tujuan aplikasi untuk mempermudah serta meningkatkan kepatuhan wajib pajak. Oleh karena itu, penting untuk mengidentifikasi apa saja permasalahan yang dialami pengguna ketika mencoba login ke dalam aplikasi, serta melihat apakah terdapat keluhan lain yang juga sering dikeluhkan pengguna aplikasi M-Pajak yang mengakibatkan rendahnya rating aplikasi. Untuk menjawab hal tersebut, ulasan pengguna di Google Play Store dapat menjadi sumber informasi yang berharga [6]. Namun, karena jumlah ulasan sangat banyak, maka diperlukan pendekatan yang lebih efektif dan efisien untuk mengekstrak informasi dari data tersebut. Salah satu pendekatan yang dapat digunakan adalah analisis sentimen. Teknik ini dapat membantu mengelompokkan ulasan berdasarkan persepsi pengguna, sehingga kendala yang paling sering muncul seperti masalah login dan keamanan aplikasi dapat diidentifikasi dengan lebih efisien [7].

Berbagai penelitian yang berkaitan dengan analisis sentimen telah dilakukan sebelumnya. Pada penelitian yang dilakukan oleh Shenifya (2024), analisis sentimen layanan kurir online di Arab Saudi dengan menggunakan algoritma Decision Tree dan SVM menunjukkan kinerja yang baik dengan akurasi masing-masing sebesar 95.01% dan 93.60%. Namun, SVM memerlukan waktu pelatihan yang lama pada dataset besar karena kompleksitas pada penyesuaian hyperparameter [8].

Kemudian, penelitian oleh Dhamayanthi & Lavanya (2024) yang melakukan analisis sentimen pada produk online menunjukkan bahwa algoritma SVM memiliki akurasi tertinggi sebesar 85%, diikuti Decision Tree (80%) dan Naïve Bayes (78%). Meskipun ketiga algoritma yang digunakan sudah menghasilkan akurasi yang cukup baik, namun Naïve Bayes memiliki kelemahan dalam menangani bahasa yang memiliki makna ganda (ambigu) karena model ini menganggap setiap kata dalam data tidak saling berkaitan [9].

Kemudian pada penelitian yang dilakukan oleh Abrari & Abdulloh (2024), analisis sentimen aplikasi Digital Identity dengan menggunakan 3 algoritma yaitu SVM, Naïve Bayes, dan KNN. Hasilnya menunjukkan algoritma SVM memiliki akurasi tertinggi sebesar 85,60%, dan KNN yang terendah sebesar 74,10%. KNN memiliki kinerja terendah karena bekerja berdasarkan jarak (k) sehingga membutuhkan waktu pemrosesan yang lama [10].

Selanjutnya, pada penelitian yang dilakukan oleh Haq et al. (2024), analisis sentimen untuk mendeteksi hoaks di media sosial menggunakan algoritma Naïve Bayes menghasilkan akurasi sebesar 88%. Namun, algoritma ini terbatas dalam menangani hubungan antar kata, karena menganggap setiap kata berdiri sendiri [11].

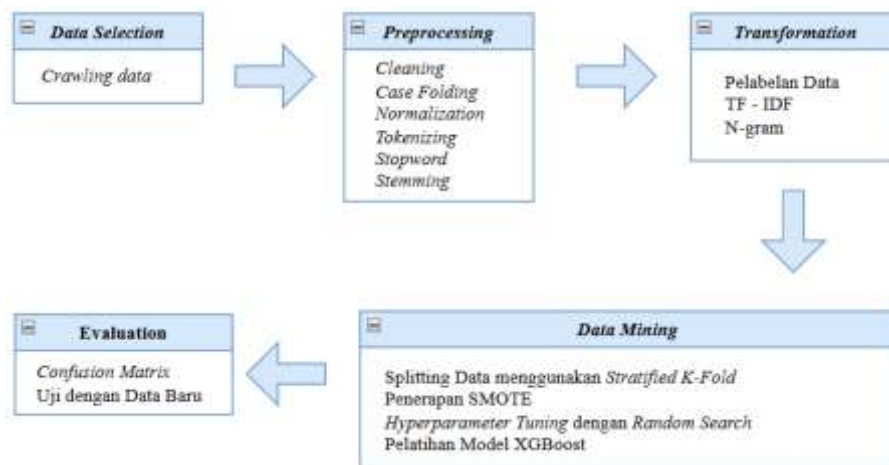
Beberapa model yang digunakan pada penelitian sebelumnya masih memiliki keterbatasan dalam aspek efisiensi, interpretasi konteks, dan skalabilitas, sehingga diperlukan model yang mampu mengatasi tantangan

tersebut secara menyeluruh dan memberikan hasil yang lebih optimal. Salah satu pendekatan yang dinilai efektif untuk mengatasi berbagai keterbatasan tersebut adalah dengan menerapkan algoritma Extreme Gradient Boosting (XGBoost), yang merupakan bagian dari metode Ensemble Learning. XGBoost dikenal memiliki keunggulan dalam menangani data dalam jumlah besar dengan efisien, karena didukung oleh teknik regularisasi serta mekanisme parallel processing dan tree pruning yang mempercepat proses pelatihan [12].

Berdasarkan permasalahan penelitian sebelumnya, akan dilakukan analisis sentimen ulasan pengguna pada aplikasi M-Pajak menggunakan XGBoost dengan penerapan Random Search untuk mengatasi kompleksitas dalam penyesuaian hyperparameter. Selain itu, N-gram akan digunakan untuk menangkap hubungan antar kata. Penelitian ini diharapkan dapat memberikan wawasan mengenai persepsi pengguna terhadap aplikasi, sehingga dapat membantu pengembang aplikasi dalam meningkatkan kualitas layanan.

2. Metodologi Penelitian

Dalam penelitian ini, metode yang digunakan yaitu Knowledge Discovery in Database (KDD) dengan tahapan yang terdiri dari *Data Selection*, *Preprocessing*, *Transformation*, *Data Mining*, dan *Evaluation*.



Gambar 1. Rancangan Penelitian

Berikut keterangan dari rancangan penelitian yang akan dilakukan:

a. *Data Selection*

Tahap pertama yang akan dilakukan adalah proses penarikan data untuk mengumpulkan ulasan aplikasi M-Pajak dari Google Play Store, lalu memilih kolom data yang diperlukan.

b. *Preprocessing*

Pada tahap preprocessing, data yang diperoleh akan dibersihkan untuk menghilangkan elemen yang tidak relevan, seperti simbol, spasi berlebih, dan kata-kata umum yang tidak memiliki makna. Proses ini bertujuan untuk meningkatkan kualitas data sehingga analisis sentimen dapat dilakukan dengan lebih akurat.

1. *Cleaning*

Proses ini bertujuan untuk menghapus elemen-elemen yang tidak diinginkan dalam teks, seperti spasi berlebih, karakter, URL, atau tanda baca yang tidak relevan dengan analisis sentimen.

2. *Case Folding*

Pada tahap ini, seluruh teks akan diubah menjadi huruf kecil (lowercase) untuk memastikan konsistensi dan menghindari perbedaan antara kata yang sama yang ditulis dengan huruf besar atau kecil.

3. *Normalization*

Proses ini berguna untuk menyeragamkan kata-kata informal menjadi bentuk yang lebih baku agar dapat dikenali sebagai entitas yang sama.

4. *Tokenizing*

Proses ini akan memecah teks menjadi unit-unit terkecil atau token, yang dapat berupa kata atau frasa.

5. *Stopwords Removal*

Proses ini menghapus kata-kata umum atau tidak penting yang tidak memiliki kontribusi dalam proses klasifikasi. Dengan melakukan stopwords removal, fokus analisis dapat diarahkan pada kata-kata yang lebih relevan.

6. *Stemming*

Proses ini akan menghilangkan imbuhan dan mengembalikan kata ke bentuk dasarnya untuk menyederhanakan kata, sehingga analisis dapat dilakukan dengan lebih konsisten dan efisien.

c. *Transformation*

Data yang telah melalui tahap *preprocessing* selanjutnya diubah ke dalam format yang sesuai untuk analisis lebih lanjut. Pada tahap awal, data diberi label sentimen menggunakan pendekatan berbasis Lexicon. Setelah pelabelan, data teks diubah menjadi representasi numerik menggunakan teknik TF-IDF (*Term Frequency-Inverse Document Frequency*), yang mengukur pentingnya setiap kata dalam. Selain itu, N-gram juga akan diterapkan untuk menangkap hubungan antar kata.

d. *Data Mining*

Pada tahap *data mining*, tujuan utamanya adalah untuk membangun model analisis sentimen dengan menggunakan data yang telah melalui proses *preprocessing* dan transformasi. Algoritma XGBoost akan diterapkan untuk mengklasifikasikan ulasan pengguna. *Splitting data* akan dilakukan menggunakan Stratified K-Fold. Untuk mengatasi masalah ketidakseimbangan kelas dalam data, digunakan metode SMOTE (*Synthetic Minority Over-sampling Technique*) untuk mengatasi ketidakseimbangan kelas dalam data, dengan cara menghasilkan sampel data sintesis untuk kelas minoritas. Kemudian, dilakukan penyesuaian hyperparameter dengan metode Random Search untuk menemukan kombinasi hyperparameter terbaik, yang bertujuan untuk meningkatkan kinerja model. Setelah menemukan kombinasi yang optimal, model XGBoost akan dilatih menggunakan dataset yang telah ditransformasi untuk mengklasifikasikan ulasan pengguna menjadi dua kategori sentimen, yaitu positif dan negatif.

e. *Evaluation*

Evaluasi model akan dilakukan dengan menggunakan confusion matrix untuk mendapatkan nilai accuracy model. Selain itu, pengujian pada data baru yang belum pernah dilihat atau digunakan selama proses pelatihan juga dilakukan, sehingga dapat memberikan gambaran yang lebih objektif terhadap kemampuan generalisasi model dalam mengklasifikasikan ulasan pengguna secara akurat di situasi nyata.

3. Hasil dan Pembahasan

Penelitian ini bertujuan untuk melakukan analisis sentimen terhadap ulasan aplikasi M-Pajak pada Google Play Store dengan menggunakan algoritma XGBoost sebagai metode klasifikasi. Model XGBoost yang telah dilatih kemudian dievaluasi menggunakan confusion matrix untuk memperoleh nilai *accuracy*.

Dalam analisis sentimen, capaian keberhasilan dari metrik *accuracy* dipengaruhi oleh banyak faktor, seperti teknik pelabelan yang digunakan, proporsi distribusi kelas dalam data dan tujuan analisis yang akan dilakukan. Secara umum, tolak ukur minimum nilai *accuracy* yang cukup baik dalam analisis sentimen adalah di atas 80% (Lexalytics, 2024). Namun, pada kasus analisis sentimen ulasan aplikasi, akurasi di atas 70% sudah mencerminkan performa model yang cukup baik. Sebagai contoh, pada penelitian yang dilakukan oleh Nurzaman et al. (2024), analisis sentimen ulasan aplikasi Threads menghasilkan akurasi sebesar 73%, yang dinilai cukup untuk menggambarkan persepsi pengguna. Kemudian penelitian yang dilakukan oleh Engkus et al. (2024), dalam menganalisis sentimen ulasan aplikasi MyPertamina menunjukkan bahwa model sudah cukup baik pada tingkat akurasi sebesar 78%. Akurasi diatas 70% dianggap sudah cukup karena analisis sentimen ulasan aplikasi bersifat opiniatif dan tidak berdampak pada keputusan yang kritis.

3.1. Data Selection

Langkah pertama dalam penelitian ini adalah mengumpulkan data ulasan aplikasi M-Pajak dari Google Play Store. Data yang digunakan mencakup ulasan pengguna dalam rentang waktu 14 Juli 2021 hingga 16 Januari 2025 dengan penerapan filter “Relevan” untuk memastikan keterkaitan hasil analisis. Data yang didapatkan dari proses scraping berjumlah 3.942 baris ulasan aplikasi M-Pajak dalam bentuk teks. Data tersebut terdiri dari beragam bentuk tulisan, dan mengandung banyak noise seperti penggunaan kata yang disingkat, istilah tidak baku, serta elemen-elemen tidak terstruktur seperti simbol, emoji, dan tanda baca. Oleh karena itu perlu dilakukan preprocessing untuk membersihkannya.

3.2. Preprocessing

Pada tahap *preprocessing*, akan dilakukan serangkaian langkah untuk mempersiapkan data agar siap digunakan dalam analisis. Proses ini mencakup pengolahan teks dan penyesuaian format data sesuai dengan kebutuhan penelitian.

3.2.1. Cleaning

Pada dataset yang akan digunakan, masih terdapat elemen-elemen yang tidak diperlukan seperti emoji, tanda baca berlebih, karakter selain huruf dan angka, serta spasi berlebih. Oleh karena itu akan perlu dilakukan proses cleaning untuk menghilangkannya agar teks menjadi lebih terstruktur. Berikut hasil dari proses cleaning (Tabel 1):

Tabel 1. Sampel Hasil *Data Cleaning*

Sebelum	Sesudah
Aplikasi anjayyy bgttt Ngebug parah😭. Ini gimana mau laporan SPT kalau nomor efin aja susahnya mintak ampun.. Tolonglah, ini aplikasi kan dibuat permudah untuk masyarakat kenapa lama banget sih. sudah klik meminta permohonan malah balik ke tampilan awal.	Aplikasi anjayyy bgttt Ngebug parah Ini gimana mau laporan SPT kalau nomor efin aja susahnya mintak ampun Tolonglah ini aplikasi kan dibuat permudah untuk masyarakat kenapa lama banget sihsudah klik meminta permohonan malah balik ke tampilan awal

3.2.2. Case Folding

Pada tahap *case folding*, seluruh teks diubah menjadi huruf kecil untuk mengatasi perbedaan kapitalisasi yang dapat menyebabkan ketidakkonsistenan (Tabel 2):

Tabel 2. Sampel Hasil *Case Folding*

Sebelum	Sesudah
Aplikasi anjayyy bgttt Ngebug parah Ini gimana mau laporan SPT kalau nomor efin aja susahnya mintak ampun Tolonglahini aplikasi kan dibuat permudah untuk masyarakat kenapa lama banget sihsudah klik meminta permohonan malah balik ke tampilan awal	aplikasi anjayyy bgttt ngebug parah ini gimana mau laporan spt kalau nomor efin aja susahnya mintak ampun tolonglahini aplikasi kan dibuat permudah untuk masyarakat kenapa lama banget sihsudah klik meminta permohonan malah balik ke tampilan awal

3.2.3. Normalization

Pada tahap *normalization*, dilakukan penyamaan variasi penulisan kata, seperti singkatan atau ejaan yang tidak baku, guna meningkatkan konsistensi data teks. Langkah ini penting agar proses analisis lebih akurat. Berikut hasil dari proses *normalization* (Tabel 3):

Tabel 3. Sampel Hasil *Normalization*

Sebelum	Sesudah
aplikasi anjayyy bgttt ngebug parah ini gimana mau laporan spt kalau nomor efin aja susahnya mintak ampun tolonglahini aplikasi kan dibuat permudah untuk masyarakat kenapa lama banget sihsudah klik meminta permohonan malah balik ke tampilan awal	aplikasi anjayyy sangat lambat parah ini bagaimana mau laporan spt kalau nomor efin saja sulit minta ampun tolong lah ini aplikasi kan dibuat mudah untuk masyarakat kenapa lama sekali sih sudah klik minta permohonan malah kembali ke tampilan awal

3.2.4. Tokenizing

Pada tahap *tokenizing*, teks yang telah melalui proses normalisasi kemudian dipecah menjadi unit-unit terkecil, yaitu kata (token). Berikut hasil dari proses *tokenizing* (Tabel 4):

Tabel 4. Sampel Hasil *Tokenizing*

Sebelum	Sesudah
aplikasi anjayyy sangat lambat parah ini bagaimana mau laporan spt kalau nomor efin saja sulit minta ampun tolong lah ini aplikasi kan dibuat mudah untuk masyarakat kenapa lama sekali sih sudah klik minta permohonan malah kembali ke tampilan awal	['aplikasi', 'anjayyy', 'sangat', 'lambat', 'parah', 'ini', 'bagaimana', 'mau', 'laporan', 'spt', 'kalau', 'nomor', 'efin', 'saja', 'sulit', 'minta', 'ampun', 'tolong', 'lah', 'ini', 'aplikasi', 'kan', 'dibuat', 'mudah', 'untuk', 'masyarakat', 'kenapa', 'lama', 'sekali', 'sih', 'sudah', 'klik', 'minta', 'permohonan', 'malah', 'kembali', 'ke', 'tampilan', 'awal']

3.2.5. Stopword Removal

Pada tahap *stopwords removal*, kata-kata umum yang tidak memberikan informasi penting akan

dihapus. Penghapusan ini bertujuan untuk meningkatkan kualitas data dan memfokuskan analisis pada kata-kata yang lebih relevan. Berikut hasil dari proses *stopwords removal* (Tabel 5):

Tabel 5. Sampel Hasil *Stopword Removal*

Sebelum	Sesudah
['aplikasi', 'anjayyy', 'sangat', 'lambat', 'parah', 'ini', 'bagaimana', 'mau', 'laporan', 'spt', 'kalau', 'nomor', 'efin', 'saja', 'sulit', 'minta', 'ampun', 'tolong', 'lah', 'ini', 'aplikasi', 'kan', 'dibuat', 'mudah', 'untuk', 'masyarakat', 'kenapa', 'lama', 'sekali', 'sih', 'sudah', 'klik', 'minta', 'permohonan', 'malah', 'kembali', 'ke', 'tampilan', 'awal']	['lambat', 'parah', 'laporan', 'spt', 'nomor', 'efin', 'sulit', 'ampun', 'mudah', 'masyarakat', 'klik', 'permohonan', 'tampilan', 'pakai', 'data', 'pakai', 'wifi']

3.2.6. Stemming

Pada tahap *stemming*, kata-kata yang telah melalui proses penghapusan *stopwords* diubah menjadi bentuk dasarnya. Proses ini membantu mengurangi variasi kata dan memastikan bahwa kata-kata yang memiliki arti serupa bisa dikenali sebagai satu entitas yang sama. Berikut hasil dari proses *Stemming* (Tabel 6):

Tabel 6. Sampel Hasil *Stemming*

Sebelum	Sesudah
['lambat', 'parah', 'laporan', 'spt', 'nomor', 'efin', 'sulit', 'ampun', 'mudah', 'masyarakat', 'klik', 'permohonan', 'tampilan', 'pakai', 'data', 'pakai', 'wifi']	['lambat', 'parah', 'lapor', 'spt', 'nomor', 'efin', 'sulit', 'ampun', 'mudah', 'masyarakat', 'klik', 'mohon', 'tampilan', 'pakai', 'data', 'pakai', 'wifi']

3.3. Transformation

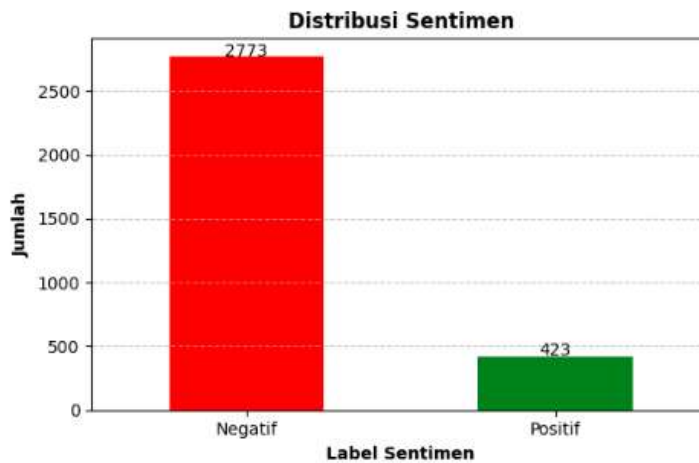
Pada tahap ini akan dilakukan pelabelan menggunakan pendekatan berbasis Lexicon. Proses ini dimulai dengan menghitung skor sentimen untuk setiap data ulasan. Skor sentimen dihitung berdasarkan jumlah nilai dari kata-kata yang muncul di dalam ulasan. Setiap kata dibandingkan dengan daftar kata positif dan negatif dari kamus Lexicon. Kata positif akan diberi skor 1 dan kata negatif akan diberi skor -1. Hasil perhitungan skor sentimen dapat dilihat pada Gambar 2.

	normalized_review	word_scores	sentiment_score	Label
0	aplikasi tidak berguna sudah 3 jam mengakses l...	aplikasi(-1), tidak(-1), berguna(1), sudah(0),...	-12	Negatif
1	mau aplikasi situs website atau instansi lain...	mau(0), aplikasi(-1), situs(0), website(0), at...	-11	Negatif
2	aplikasi anjayyy sangat lambat parah ini bagai...	aplikasi(-1), anjayyy(0), sangat(0), lambat(1)...	-11	Negatif
3	pendaftaran lewat coretax rumit tidak ada pand...	pendaftaran(1), lewat(0), coretax(0), rumit(-1)...	-2	Negatif
4	aplikasi tidak jelas telpon nomor cs sibuk ter...	aplikasi(-1), tidak(-1), jelas(0), telpon(0), ...	-7	Negatif
...
3796	telkomsel roli	telkomsel(0), roli(0)	0	Netral
3797	oke nyaman	oke(1), nyaman(1)	2	Positif
3798	jos sekali	jos(0), sekali(-1)	-1	Negatif
3799	oke	oke(1)	1	Positif
3800	oke	oke(1)	1	Positif

3801 rows × 4 columns

Gambar 2. Hasil Perhitungan Skor Sentimen

Hasil pelabelan sentimen dapat dilihat pada Gambar 3. Pada penelitian ini, sentimen netral tidak digunakan, sehingga total datanya berkurang menjadi 2.773 ulasan.



Gambar 3. Distribusi Sentimen

Selanjutnya, data akan diubah menjadi representasi numerik karena algoritma XGBoost hanya dapat memproses data dalam bentuk numerik. Oleh karena itu, data teks ulasan akan dikonversi menggunakan metode TF-IDF (*Term Frequency-Inverse Document Frequency*). Hasil dari pembobotan TF-IDF dapat dilihat pada Gambar 4.

	TERM	TF	IDF	TF-IDF
19088	oke	0.011521	3.106811	0.035793
15660	mantap	0.008365	3.360125	0.028108
3949	buruk	0.006285	2.722548	0.017112
23753	sampah	0.003963	3.369175	0.013352
8457	ganggu	0.004929	2.548194	0.012560
...
10750	jabar mudik bertahun-tahun	0.000008	8.069655	0.000067
19263	orang kpp contoh	0.000008	8.069655	0.000067
13360	kulonprogo pindah domisili	0.000008	8.069655	0.000067
4926	contoh warga kulonprogo	0.000008	8.069655	0.000067
4925	contoh warga	0.000008	8.069655	0.000067

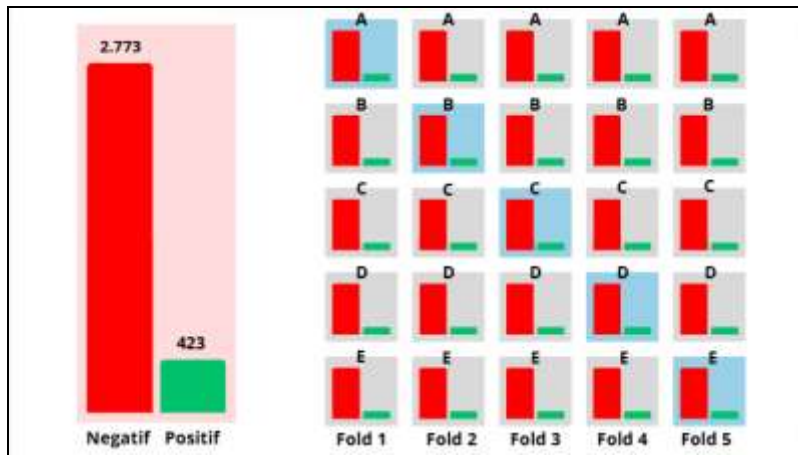
29274 rows × 4 columns

Gambar 4. Hasil Pembobotan TF-IDF

TF-IDF bekerja dengan cara menghitung bobot setiap kata dalam ulasan berdasarkan frekuensi kemunculannya di suatu dokumen (Term Frequency) dan distribusinya di seluruh dokumen (Inverse Document Frequency). Dalam penelitian ini, TF-IDF diterapkan dengan menggunakan teknik N-gram (unigram, bigram, dan trigram) untuk menangkap pola kata yang lebih kompleks dalam teks ulasan.

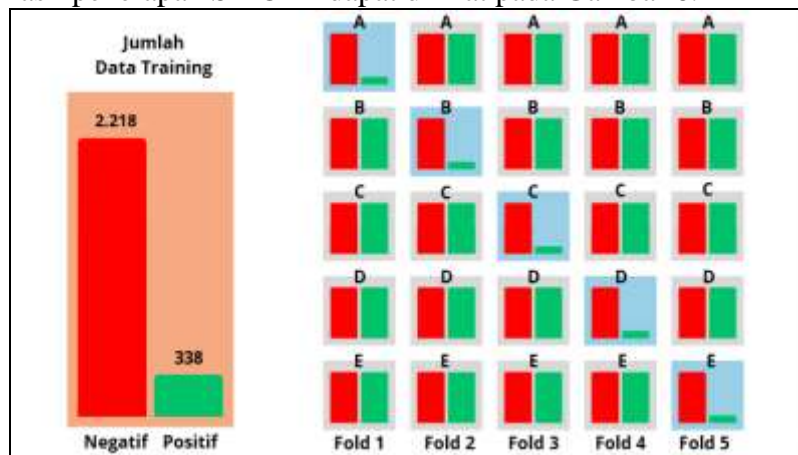
3.4. Data Mining

Pada tahap data mining, dilakukan pelatihan model klasifikasi menggunakan algoritma XGBoost. Data dibagi menjadi data training dan testing menggunakan teknik Stratified K-Fold Cross Validation dengan nilai K=5, karena dataset bersifat tidak seimbang (2.773 data negatif dan 423 data positif). Teknik ini menjaga distribusi label sentimen yang seimbang di setiap fold, sehingga evaluasi model menjadi lebih adil dan konsisten [21]. Proses pembagian data dapat dilihat pada Gambar 5 berikut ini:



Gambar 5. Proses Splitting Data Stratified K-Fold

Setiap *fold* mempertahankan proporsi kelas secara konsisten. Fold berwarna biru merupakan data testing, sementara abu-abu menunjukkan data training. Satu fold digunakan sebagai testing secara bergantian, sehingga seluruh data mendapatkan giliran. Namun, karena dataset yang digunakan tidak seimbang, maka akan diterapkan teknik SMOTE (*Synthetic Minority Over-sampling Technique*) pada data training di setiap fold. SMOTE menyeimbangkan jumlah data antar kelas dengan mensintesis sampel baru pada kelas minoritas, tanpa diterapkan pada data testing untuk menghindari *data leakage* [22]. Hasil penerapan SMOTE dapat dilihat pada Gambar 6.



Gambar 6. Hasil SMOTE

Distribusi kelas menjadi lebih seimbang setelah SMOTE diterapkan, yang membantu mengurangi bias dan meningkatkan kemampuan model dalam mengklasifikasikan data secara lebih akurat. Selain itu, pada algoritma XGBoost pengaturan hyperparameter juga penting karena berpengaruh terhadap performa model. Parameter seperti jumlah pohon, kedalaman, dan learning rate perlu disesuaikan agar model tidak overfitting [23]. Penyesuaian dilakukan dengan metode Random Search, yang dipilih karena efisien dalam menemukan kombinasi parameter optimal [24]. Hasil penyesuaian Hyperparameter dapat dilihat pada Tabel 7.

Tabel 7. Hasil Penyesuaian *Hyperparameter*

Hyperparameter	Random Search Values	Nilai Hyperparameter Terbaik				
		Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
<i>n_estimators</i>	50, 100, 150, 200	200	150	200	200	200
<i>max_depth</i>	5, 7	7	5	7	7	7
<i>learning_rate</i>	0.05, 0.1, 0.2	0.2	0.2	0.2	0.2	0.2
<i>subsample</i>	0.8, 0.9, 1	1	0.8	1	1	1
<i>cosample_bytree</i>	0.4, 0.6	0.4	0.4	0.4	0.4	0.4
<i>gamma</i>	0.1, 1, 2	1	0.1	1	1	1
<i>reg_alpha</i>	0.01, 0.1, 0.5	0.5	0.1	0.5	0.5	0.5
<i>reg_lambda</i>	0.1, 0.5	0.1	0.1	0.1	0.1	0.1

Selanjutnya, pelatihan model XGBoost dilakukan menggunakan data yang telah diseimbangkan dengan SMOTE serta menggunakan kombinasi parameter terbaik yang diperoleh melalui proses penyesuaian hyperparameter dengan Random Search. Gambaran proses pelatihan model XGBoost dapat dilihat pada Gambar 7.

Iterasi	Loss Function	Gradien	Gamma	T (Leaf per Tree)	L1 Regularisasi	L2 Regularisasi	Regularisasi	Objektif
0	1	0.621886	0.000000	1	7	3.5	2.45	12.95 13.571886
1	2	0.575909	-0.045977	1	7	3.5	2.45	12.95 13.525909
2	3	0.549736	-0.026173	1	7	3.5	2.45	12.95 13.499736
3	4	0.526350	-0.023387	1	7	3.5	2.45	12.95 13.476350
4	5	0.488568	-0.037781	1	7	3.5	2.45	12.95 13.438568
...
195	196	0.054141	-0.000000	1	7	3.5	2.45	12.95 13.004141
196	197	0.054141	-0.000000	1	7	3.5	2.45	12.95 13.004141
197	198	0.054141	-0.000000	1	7	3.5	2.45	12.95 13.004141
198	199	0.054141	-0.000000	1	7	3.5	2.45	12.95 13.004141
199	200	0.054141	0.000000	1	7	3.5	2.45	12.95 13.004141

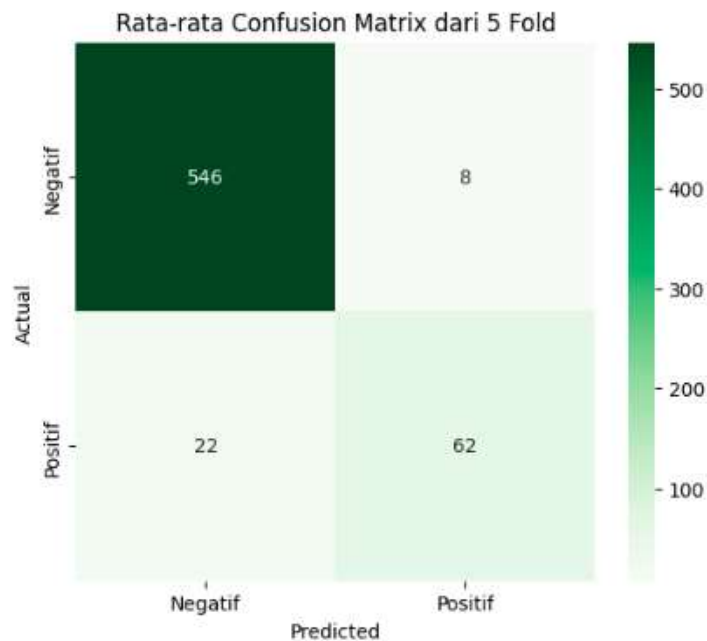
200 rows x 9 columns

Gambar 7. Pelatihan Model XGBoost

Model XGBoost akan mengoptimalkan model dengan cara meminimalkan fungsi objektif yang terdiri dari dua bagian, yaitu loss function dan regularisasi. Pada setiap iterasi, nilai *loss function* akan menurun secara bertahap yang menunjukkan peningkatan performa model. Sementara itu, regularisasi akan mengurangi bobot fitur yang tidak relevan.

3.5. Evaluation

Setelah melalui proses pelatihan, model akan dievaluasi untuk mengukur sejauh mana model dapat menggeneralisasi data testing. Evaluasi pertama akan dilakukan dengan menggunakan confusion matrix untuk melihat jumlah prediksi yang benar dan salah dari masing-masing kelas. Rata-rata nilai confusion matrix dari kelima *fold* dapat dilihat pada Gambar 8.

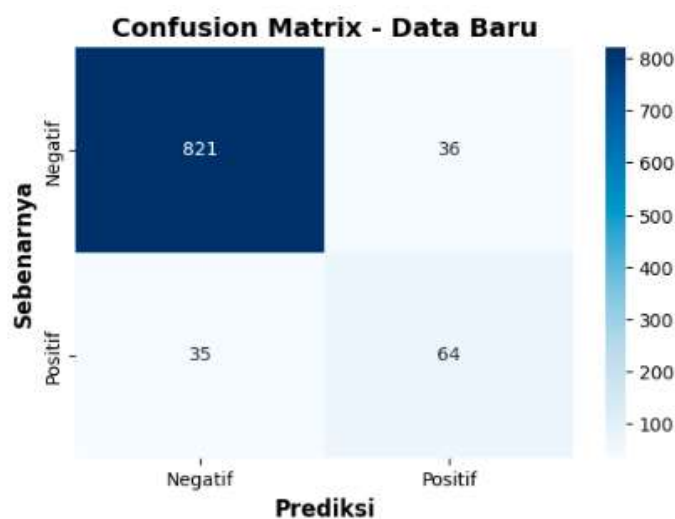


Gambar 8. *Confusioon Matrix*

Berdasarkan confusion matrix pada gambar 8, didapatkan nilai akurasi model XGBoost dari kelima *fold* sebagai berikut:

$$Accuracy = \frac{546 + 62}{546 + 62 + 8 + 22} = \frac{608}{638} = 0,952 \approx 95\% \quad (1)$$

Untuk memastikan kemampuan generalisasi model terhadap data yang belum pernah dilihat sebelumnya, akan dilakukan pengujian dengan data baru. Sama seperti pada data pelatihan, data baru ini juga telah melalui proses preprocessing untuk memastikan kesesuaian format dan struktur data. Proses preprocessing yang dilakukan meliputi *cleaning*, *case folding*, *normalization*, *tokenizing*, *stopwords removal* dan *stemming*. Dengan demikian, data baru memiliki karakteristik dan kriteria yang sama dengan data yang digunakan untuk pelatihan model. Hasil confusion matrix pada data baru dapat dilihat pada gambar 9.



Gambar 9. *Confusion Matrix Data Baru*

Sehingga, nilai akurasi pada data baru sebagai berikut:

$$\text{Accuracy data baru} = \frac{821+64}{821+64+36+35} = 93\% \quad (2)$$

Model menghasilkan akurasi sebesar 93% pada data baru, sedikit menurun 2% dari data latih. Penurunan ini kemungkinan disebabkan oleh kemunculan kosakata baru yang belum dikenali model. Meskipun demikian, akurasi tersebut menunjukkan bahwa model tetap stabil dan mampu melakukan generalisasi dengan baik.

3.6. Knowledge Representation

Pada tahap ini, akan ditampilkan representasi *knowledge* dalam bentuk *wordcloud* untuk menggambarkan fitur atau layanan pada aplikasi M-Pajak yang paling banyak menerima kritik negatif dari pengguna. *Wordcloud* akan disajikan dalam tiga jenis, yaitu *unigram*, *bigram*, dan *trigram*, untuk memberikan gambaran yang lebih jelas mengenai kata yang paling sering muncul dalam ulasan bersentimen negatif. Hasil *wordcloud unigram* (satu kata) ditampilkan pada Gambar 10.



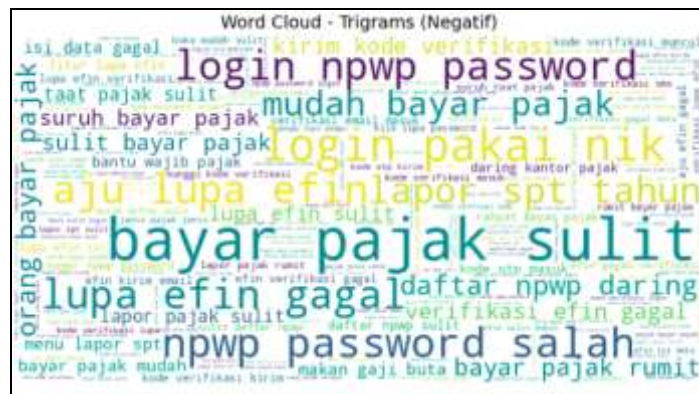
Gambar 10. Wordcloud Unigram

Selanjutnya, untuk memberikan konteks yang lebih jelas, hasil *wordcloud bigram* (dua kata) ditampilkan pada Gambar 11.



Gambar 11. Wordcloud Bigram

Selanjutnya, hasil *wordcloud trigram* (tiga kata) pada Gambar 12 akan ditampilkan untuk memberikan konteks yang lebih mendetail mengenai permasalahan.



Berdasarkan hasil *wordcloud trigram* pada Gambar 12, kombinasi kata yang sering muncul dalam ulasan negatif, yaitu bayar pajak sulit, NPWP password salah, lupa efin gagal, login NPWP password, dan aju lupa efin, semakin menegaskan bahwa pengguna menghadapi tantangan dalam pembayaran pajak dan akses akun. Untuk memperkuat temuan *wordcloud*, dilakukan analisis frekuensi kata pada ulasan negatif yang ditampilkan dalam bentuk *unigram*, *bigram*, dan *trigram* pada Tabel 8.

No.	Unigram	Frekuensi	Bigram	Frekuensi	Trigram	Frekuensi
1.	pajak	666	bayar pajak	190	bayar pajak sulit	34
2.	sulit	612	lupa efin	108	npwp password salah	16
3.	login	452	lapor pajak	70	login pakai nik	15
4.	efin	427	kode verifikasi	70	daftar npwp daring	13
5.	daftar	387	lapor spt	64	orang bayar pajak	12
6.	npwp	348	daftar npwp	61	aju lupa efin	11

Berdasarkan Tabel 8, terlihat bahwa keluhan pengguna paling banyak terdapat pada aspek pembayaran pajak, kesulitan login, serta permasalahan terkait efin. Untuk memperoleh pemahaman yang lebih mendalam mengenai bentuk keluhan yang dialami pengguna, akan ditampilkan beberapa kutipan ulasan yang merepresentasikan permasalahan utama tersebut pada Tabel 9.

Permasalahan	Kutipan Ulasan
Login sulit	Mau masuk <i>login</i> aja susah banget. Minta kode verifikasi lewat email gak dikirim. Gantian selang minta kode, yang pertama baru muncul pas sudah kadaluarsa. Sampai minta 10 kali.
NPWP password salah	Mau <i>login</i> aja susah amat dibidang npwp sama password salah mulu. Mau minta Efin ehh malah dikirimnya ke yopmail padahal semua juga pakainya gmail . . Lagian mana ada yg pakai email yopmail
Lupa efin gagal	Tidak Jelas, melakukan pengajuan lupa EFIN selalu gagal verifikasi padahal semua data sudah sesuai. Dicoba berkali-kali tetap <i>error</i> terus.
Aju lupa efin	Pengiriman kode otp ketika melakukan pengajuan lupa e-fin sangat lama baik melalui sms maupun e-mail, padahal pengiriman otp melalui sms dikenakan tarif

Berdasarkan kutipan ulasan yang ditampilkan dalam Tabel 9, dapat disimpulkan bahwa permasalahan utama yang dihadapi pengguna aplikasi M-Pajak berkaitan erat dengan kendala teknis. Permasalahan tersebut meliputi kesulitan dalam proses login, ketidaksesuaian data seperti NPWP dan password yang dianggap salah oleh sistem meskipun telah benar, serta kegagalan dalam pengajuan dan verifikasi efin. Permasalahan-permasalahan ini menunjukkan bahwa aplikasi belum sepenuhnya memenuhi ekspektasi pengguna dalam hal kemudahan akses dan pelayanan digital yang seharusnya mempermudah, bukan memperumit proses perpajakan.

4. Kesimpulan dan Saran

Berdasarkan hasil penelitian, dapat disimpulkan bahwa algoritma XGBoost yang dioptimalkan dengan metode Random Search mampu mengklasifikasikan sentimen ulasan pengguna secara efektif, dengan akurasi 95% pada data pelatihan dan 93% pada data baru melalui pembagian data Stratified K-Fold. Analisis juga menunjukkan bahwa fitur aplikasi M-Pajak yang paling banyak mendapat kritik negatif berkaitan dengan proses login dan verifikasi efin, seperti kegagalan sistem membaca NPWP atau password, serta masalah dalam pengajuan efin meskipun data sudah benar. Oleh karena itu, disarankan bagi pengembang aplikasi untuk memperbaiki sistem login dan verifikasi agar lebih stabil dan user-friendly, serta menyederhanakan fitur lupa efin dan *reset password*. Untuk penelitian selanjutnya, disarankan menggunakan metode pelabelan otomatis berbasis *pretrained language model* seperti *IndoBERT* yang lebih memahami konteks bahasa Indonesia dan tidak bergantung pada kamus seperti metode *lexicon*.

Referensi

- [1] Yusuf Katili, A., Patadjenu, W., Alhadar, S., & Pariono, A. (2024). Implementasi Teknologi E-Filing dalam Peningkatan Kepatuhan bagi Wajib Pajak. *Jurnal Administrasi, Manajemen Dan Ilmu Sosial (JAEIS)*, 3(2), 96–107.
- [2] Isma. (2021). *DJP Luncurkan Aplikasi M-Pajak dan Buku Reformasi Perpajakan*. InfoPublic (Portal Berita Info Public). <https://www.infopublik.id/kategori/nasional-ekonomi-bisnis/548990/djp-luncurkan-aplikasi-m-pajak-dan-buku-reformasi-perpajakan>.
- [3] Ekananda, A. (2024). *M-Pajak: Layanan Pajak Mudah dalam Genggaman*. DjP. <https://www.pajak.go.id/en/node/106796>.
- [4] DDTC News. (2024). *Ada Kendala Proses Login M-Pajak via Aplikasi*. <https://news.ddtc.co.id/berita/nasional/1804109/ada-kendala-proses-login-m-pajak-via-aplikasi-djp-imbau-pakai-browser>.
- [5] Febisatria, A., & Adwishanty, P. R. (2025). Analisis Persepsi Konsumen terhadap Penggunaan Aplikasi Livin by Mandiri. *Journal Ilmiah Manajemen, Bisnis Dan Kewirausahaan*, 5, 243–255. <https://doi.org/10.55606/jurimbik.v5i2.1078>.
- [6] Septiani, A., & Budi, I. (2022). Klasifikasi Ulasan Pengguna Aplikasi: Studi Kasus Aplikasi Ipusnas Perpustakaan Nasional Republik Indonesia (PNRI). *JUPI (Jurnal Ilmiah Penelitian Dan Pembelajaran Informatika)*, 7(4), 1110–1120. <https://doi.org/10.29100/jupi.v7i4.3216>.
- [7] Tanggraeni, A. I., & Sitokdana, M. N. N. (2022). Analisis Sentimen Aplikasi E-Government pada Google Play Menggunakan Algoritma Naïve Bayes. *JATISI (Jurnal Teknik Informatika Dan Sistem Informasi)*, 9(2), 785–795. <https://doi.org/10.35957/jatisi.v9i2.1835>.
- [8] Shenifya, M. (2024). Sentiment Analysis of Social Media Discourse on Public Perception of Online Courier Services in Saudi Arabia using Machine Learning. *International Journal of Data and Network Science*, 1–10. <https://doi.org/10.5267/j.ijdns.2024.8.002>.

- [9] Dhamayanthi, N., & Lavanya, B. (2024). The Role Of Naïve Bayes, SVM, And Decision Trees In Sentiment Analysis. *Educational Administration: Theory and Practice*, 30(4), 6377–6381. <https://kuey.net/>.
- [10] Abrari, M. naufal M., & Abdulloh, F. F. (2024). Comparison of Machine Learning Algorithm for Sentiment Analysis of Digital Identity Application Users. *Journal of Computing and Information System*, 20(2), 146–154. <https://doi.org/10.33480/pilar.v20i2.5736>.
- [11] Haq, M. Z., Octiva, C. S., Ayuliana, A., Nuryanto, U. W., & Suryadi, D. (2024). Algoritma Naïve Bayes untuk Mengidentifikasi Hoaks di Media Sosial. *Jurnal Minfo Polgan*, 13(1), 1079–1084. <https://doi.org/10.33395/jmp.v13i1.13937>.
- [12] Muhammady, D. N., Nugraha, H. A. E., Nastiti, V. R. S., & Aditya, C. S. K. (2024). Students Final Academic Score Prediction Using Boosting Regression Algorithms. *Jurnal Ilmiah Teknik Elektro Komputer Dan Informatika*, 10(1), 154. <https://doi.org/10.26555/jiteki.v10i1.28352>.
- [13] Olamedy, J. (2024). *Panduan Lengkap untuk Stratified K-Fold Cross-validation untuk Data yang Tidak Seimbang*. <https://medium.com/@juanc.olamendy/a-comprehensive-guide-to-stratified-k-fold-cross-validation-for-unbalanced-data-014691060f17>.
- [14] Imani, M., Beikmohammadi, A., & Arabnia, H. R. (2025). Comprehensive Analysis of Random Forest and XGBoost Performance with SMOTE, ADASYN, and GNUS Under Varying Imbalance Levels. *Technologies*, 13(3), 1–40. <https://doi.org/10.3390/technologies13030088>.
- [15] Rithp. (2023). *XGBoost Tuning: Common Hyperparameter Pitfalls and How to Avoid Them*. Medium. <https://medium.com/%40rithpansanga/xgboost-tuning-common-hyperparameter-pitfalls-and-how-to-avoid-them-55636fc0a358>.
- [16] Rom, A. R. M., Jamil, N., & Ibrahim, S. (2024). Multi objective hyperparameter tuning via random search on deep learning models. *Telkomnika (Telecommunication Computing Electronics and Control)*, 22(4), 956–968. <https://doi.org/10.12928/TELKOMNIKA.v22i4.25847>.