

Klasifikasi Churn Dengan Algoritma XGBoost Menggunakan Feature Selection Boruta-Shap

Dwi Wahyu Kuncoro Hadi Sakaro*, Puspita Nurul Sabrina, Edvin Ramadhan

Fakultas Sains dan Informatika, Informatika, Universitas Jenderal Achmad Yani, Cimahi, Indonesia

Email: ^{1,*}dwiwahyukhs21@if.unjani.ac.id, ²puspita.sabrina@lecture.unjani.ac.id, ³edvin.ramadhan@lecture.unjani.ac.id

Email Penulis Korespondensi: dwiwahyukhs21@if.unjani.ac.id

Submitted: 10/07/2025; Accepted: 01/09/2025; Published: 04/09/2025

Abstrak—Customer churn merupakan permasalahan krusial bagi perusahaan telekomunikasi, karena berdampak langsung pada pendapatan dan keberlangsungan bisnis. Penelitian ini mengusulkan pengembangan model prediksi churn menggunakan algoritma Extreme Gradient Boosting (XGBoost) yang dikombinasikan dengan metode seleksi fitur Boruta dan interpretasi fitur berbasis SHAP (SHapley Additive exPlanations). Dataset yang digunakan adalah Telco Customer Churn dari Kaggle, terdiri dari 7043 data pelanggan dan 21 fitur. Tahapan penelitian meliputi data preprocessing, transformasi data, pembagian data latih dan uji (80:20), penyeimbangan data menggunakan SMOTE, seleksi fitur dengan Boruta, interpretasi fitur menggunakan SHAP, serta klasifikasi menggunakan XGBoost. Kinerja model dievaluasi menggunakan metrik akurasi, presisi, recall, dan F1-score. Hasil menunjukkan bahwa model XGBoost dengan Boruta-SHAP (Model B) mencapai akurasi 0,7576, sedikit lebih tinggi dibandingkan model tanpa seleksi fitur (Model A) dengan akurasi 0,7512. Model B juga menunjukkan peningkatan performa pada kelas mayoritas (tidak churn), dengan recall meningkat dari 0,76 menjadi 0,79 dan F1-score dari 0,82 menjadi 0,83. Namun, pada kelas minoritas (churn), recall menurun dari 0,72 menjadi 0,66 meskipun precision meningkat dari 0,52 menjadi 0,54. Temuan ini menunjukkan bahwa integrasi Boruta-SHAP mampu meningkatkan efisiensi dan interpretabilitas model, tetapi memerlukan strategi tambahan untuk mempertahankan kinerja pada kelas minoritas.

Kata Kunci: Prediksi Churn; XGBoost; Boruta-SHAP; SMOTE; Feature Selection; Telekomunikasi; Explainable AI

Abstract—Customer churn is a critical issue for telecommunications companies, as it directly impacts revenue and business sustainability. This study proposes the development of a churn prediction model using the Extreme Gradient Boosting (XGBoost) algorithm combined with the Boruta feature selection method and SHAP (SHapley Additive exPlanations)-based feature interpretation. The dataset used is the Telco Customer Churn dataset from Kaggle, consisting of 7,043 customer records and 21 features. The research stages include data preprocessing, data transformation, an 80:20 train-test split, data balancing using SMOTE, feature selection with Boruta, feature interpretation with SHAP, and classification using XGBoost. The model's performance was evaluated using accuracy, precision, recall, and F1-score metrics. Results show that the XGBoost model with Boruta-SHAP (Model B) achieved an accuracy of 0.7576, slightly higher than the model without feature selection (Model A), which achieved 0.7512. Model B also demonstrated improved performance for the majority class (non-churn), with recall increasing from 0.76 to 0.79 and F1-score from 0.82 to 0.83. However, for the minority class (churn), recall decreased from 0.72 to 0.66, although precision increased from 0.52 to 0.54. These findings indicate that integrating Boruta-SHAP can enhance model efficiency and interpretability, but additional strategies are required to maintain performance for the minority class.

Keywords: Churn Prediction; XGBoost; Boruta-SHAP; SMOTE; Feature Selection; Telecommunications; Explainable AI

1. PENDAHULUAN

Istilah customer churn umumnya digunakan untuk menggambarkan kecenderungan pelanggan yang berhenti berbisnis dengan suatu perusahaan dalam periode waktu atau kontrak tertentu. Secara tradisional, penelitian mengenai customer churn berawal dari bidang Customer Relationship Management (CRM) [1]. Dataset telekomunikasi secara alami memiliki ketidakseimbangan, di mana jumlah pelanggan yang berhenti berlangganan (churn) lebih sedikit dibandingkan dengan yang tetap berlangganan. Hanya sedikit algoritma yang secara bawaan dapat menangani masalah ini secara efektif, salah satunya adalah XGBoost [2]. Extreme Gradient Boosting (XGBoost) merupakan algoritma berbasis ensemble learning yang dirancang untuk efisiensi, skalabilitas tinggi, dan mampu mengatasi ketidakseimbangan kelas [3], [4]. Dalam berbagai studi, XGBoost menunjukkan kinerja superior dalam tugas klasifikasi termasuk prediksi churn. Dalam penelitian [5], XGBoost dikombinasikan dengan metode balancing data SMOTE untuk mengklasifikasikan churn pelanggan telekomunikasi, dan terbukti memberikan akurasi yang lebih baik dibandingkan Random Forest. Studi [3] juga menegaskan keunggulan XGBoost dalam mengklasifikasikan pelanggan churn, terutama setelah pengoptimalan parameter menggunakan algoritma genetika. Namun, beberapa penelitian menunjukkan bahwa kinerja XGBoost sangat dipengaruhi oleh dimensi fitur yang tinggi dan jumlah sampel yang terbatas. Kondisi ini dapat menyebabkan model gagal mempelajari informasi penting dari data, yang berdampak pada buruknya hasil klasifikasi [6], [7].

Untuk mengatasi masalah tersebut, proses pemilihan fitur (feature selection) menjadi langkah penting dalam tahapan pra-pemrosesan data. Feature selection bertujuan mengidentifikasi atribut yang paling relevan dan mengurangi noise serta kompleksitas model, sehingga meningkatkan akurasi dan efisiensi waktu komputasi [8], [9]. Boruta merupakan metode feature selection berbasis wrapper yang menggunakan teknik permutation importance untuk mengidentifikasi semua fitur relevan. Meskipun efektif, metode ini dikenal memiliki beban komputasi tinggi dan hasil yang kurang stabil pada dataset kompleks. Sebagai alternatif, metode SHAP (SHapley Additive exPlanations) menawarkan interpretasi global dan lokal atas kontribusi fitur terhadap output model. SHAP telah banyak digunakan pada model berbasis pohon seperti XGBoost untuk mendukung pemahaman yang lebih transparan terhadap

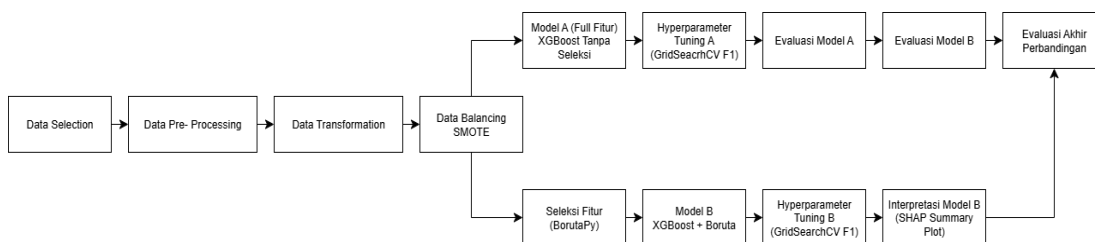
pentingnya fitur [10]. Penelitian Li Y, Yan K ditahun 2025 berhasil mengintegrasikan XGBoost dengan SHAP dalam prediksi churn pada sektor perbankan, mencapai akurasi tinggi (~97%) dan menunjukkan fitur-fitur kunci seperti total transaksi dan variasi pola transaksi mempengaruhi churn secara signifikan [11]. Sementara itu, studi [12] mengintegrasikan XGBoost, SHAP, dan teknik seleksi fitur dalam konteks pemodelan risiko kredit, yang menunjukkan peningkatan stabilitas dan akurasi prediksi.

Berdasarkan studi literatur tersebut, dapat disimpulkan bahwa meskipun XGBoost telah banyak digunakan dalam prediksi churn, integrasi metode seleksi fitur seperti Boruta dan interpretasi berbasis SHAP belum banyak dieksplorasi secara komprehensif pada domain churn pelanggan telekomunikasi. Oleh karena itu, penelitian ini bertujuan untuk mengembangkan model klasifikasi churn menggunakan algoritma XGBoost yang dikombinasikan dengan feature selection Boruta dan interpretasi SHAP untuk meningkatkan performa, efisiensi, dan interpretabilitas model.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Metode penelitian ini dimulai dengan data selection dan berakhir pada proses evaluasi. Berdasarkan jurnal[6], berikut alur metode penelitian yang akan dilakukan dapat dilihat pada Gambar 1 berikut.



Gambar 1. Metodologi Penelitian

2.2 Data Selection

Tahap pertama adalah pemilihan dataset. Penelitian ini menggunakan dataset bersumber dari Kaggle yang bernama Telco Customer Churn. Dataset ini terdiri dari 7043 data dengan 21 fitur. Targetnya terdapat pada kolom Churn yang menunjukkan apakah seorang pelanggan berhenti berlangganan (Yes) atau tidak (No) dapat dilihat pada Tabel 1 berikut.

Tabel 1. Dataset Penelitian

Atribut	Penjelasan
customerID	ID unik untuk setiap pelanggan
Gender	Jenis kelamin pelanggan (Male/Female)
SeniorCitizen	Apakah pelanggan adalah warga senior (0: Tidak, 1: Ya)
Partner	Apakah pelanggan memiliki pasangan (Yes/No)
Dependents	Apakah pelanggan memiliki tanggungan (Yes/No)
tenure	Lama pelanggan berlangganan (dalam bulan)
PhoneService	Apakah pelanggan memiliki layanan telepon (Yes/No)
MultipleLines	Apakah pelanggan memiliki lebih dari satu saluran telepon (Yes/No/No phone service)
InternetService	Jenis layanan internet (DSL/Fiber optic/No)
OnlineSecurity	Apakah pelanggan memiliki perlindungan keamanan online (Yes/No/No internet service)
OnlineBackup	Apakah pelanggan memiliki layanan backup online (Yes/No/No internet service)
DeviceProtection	Apakah pelanggan memiliki perlindungan perangkat (Yes/No/No internet service)
TechSupport	Apakah pelanggan memiliki dukungan teknis (Yes/No/No internet service)
StreamingTV	Apakah pelanggan memiliki layanan streaming TV (Yes/No/No internet service)
StreamingMovies	Apakah pelanggan memiliki layanan streaming film (Yes/No/No internet service)
Contract	Jenis kontrak pelanggan (Month-to-month/One year/Two year)
PaperlessBilling	Apakah pelanggan menggunakan tagihan tanpa kertas (Yes/No)
PaymentMethod	Metode pembayaran pelanggan
MonthlyCharges	Biaya bulanan yang dibebankan ke pelanggan
TotalCharges	Total biaya yang telah dibebankan ke pelanggan
Churn	Status apakah pelanggan berhenti berlangganan (Yes/No)

Tabel 1 menjelaskan berbagai atribut yang terdapat dalam dataset pelanggan sebuah perusahaan layanan telekomunikasi. Setiap atribut merepresentasikan informasi spesifik tentang pelanggan, seperti identitas, layanan yang digunakan, hingga status berhenti berlangganan (churn). Beberapa atribut bersifat demografis, seperti: customerID: ID unik pelanggan, Gender: jenis kelamin pelanggan, SeniorCitizen, Partner, dan Dependents yang memberikan

informasi tentang status sosial dan usia. Atribut lainnya menggambarkan layanan yang digunakan, seperti: PhoneService, MultipleLines, InternetService, hingga layanan tambahan seperti StreamingTV, TechSupport, dan DeviceProtection. Atribut Contract, PaymentMethod, MonthlyCharges, dan TotalCharges memberikan informasi mengenai jenis kontrak dan pembayaran, sedangkan Churn menunjukkan apakah pelanggan telah berhenti berlangganan layanan. Informasi ini dapat digunakan untuk menganalisis perilaku pelanggan, melakukan segmentasi, serta membangun model prediktif untuk mendeteksi risiko churn.

2.3 Pre-Processing Data

Setelah data dilakukan, data kemudian melalui proses pembersihan dan persiapan data agar siap digunakan dalam model. Langkah pertama adalah data cleaning, yaitu menangani nilai yang hilang (missing values) dengan teknik imputasi atau penghapusan jika diperlukan. Tahap preprocessing ini bertujuan untuk mempersiapkan data agar siap digunakan dalam proses pelatihan model. Tahapan ini mencakup: Menghapus atribut customerID yang tidak relevan, Mengonversi kolom TotalCharges menjadi numerik, Menghapus nilai-nilai yang hilang (missing value), Encoding label target Churn menjadi numerik: Yes = 1, No = 0, Encoding variabel kategorikal menggunakan Label Encoding.

2.4 Data Transformation

Data dalam bentuk kategorikal dikonversi menjadi bentuk numerik menggunakan teknik Label Encoder. Metode ini mengubah setiap nilai dalam kolom menjadi kolom baru dan mengisinya dengan nilai biner (0 dan 1). Tahapan transformasi data dilakukan untuk mempersiapkan data agar berada dalam format yang sesuai sebelum digunakan dalam pelatihan model. Pada dataset churn, terdapat banyak fitur dengan tipe data kategorikal seperti Contract, PaymentMethod, dan InternetService. Oleh karena itu, diperlukan proses encoding untuk mengubah data tersebut menjadi bentuk numerik. Untuk fitur kategorikal yang hanya memiliki dua nilai seperti gender atau Partner, digunakan Label Encoding.

2.5 Data Splitting

Setelah proses data processing, tahap selanjutnya dalam penelitian ini adalah membagi data menjadi dua bagian, yaitu data latih (training data) dan data uji (testing data). Dataset dibagi menjadi data latih dan data uji dengan rasio 80:20 menggunakan stratifikasi agar distribusi kelas tetap seimbang.

2.6 Data Balancing

Sebelum dilakukan proses balancing, data churn menunjukkan distribusi kelas yang tidak seimbang, dengan jumlah sampel pada kelas tidak churn (label 0) sebanyak 4125, sedangkan kelas churn (label 1) hanya sebanyak 1500 dari total 5625 data. Ketidakeimbangan ini dapat menyebabkan model cenderung bias terhadap kelas mayoritas, sehingga performa dalam mengenali pelanggan yang churn menjadi kurang optimal. Untuk mengatasi permasalahan ini, dilakukan proses balancing menggunakan metode Synthetic Minority Over-sampling Technique (SMOTE) guna meningkatkan representasi kelas minoritas dan memastikan model dapat belajar secara adil dari kedua kelas. SMOTE adalah pembangkitan data minoritas sebanyak data mayoritas[13]. dengan menggunakan rumus:

$$x_{new} = x_i + \delta \times (x_{zi} - x_i) \quad (1)$$

Dengan δ adalah nilai acak antara 0 dan 1.

2.7 Feature Selection

2.7.1 Boruta

Metode Boruta adalah algoritma seleksi fitur (Feature Selection/FS) yang dikembangkan berdasarkan prinsip kerja teknik Random Forest (RF) [9]. metode wrapper yang membandingkan fitur asli dengan fitur acak (shadow features) dan, dalam variasi terbaru seperti yang diusulkan pada sistem kontrol PEMFC, menggunakan algoritma XGBoost sebagai learner utama untuk meningkatkan akurasi dan ketahanan model[14].

2.7.2 SHAP (Shapley Additive exPlanations)

Additive exPlanation (SHAP), yang dikembangkan oleh Lundberg dan Lee, adalah metode untuk memahami prediksi yang dihasilkan oleh model pembelajaran mesin dengan memanfaatkan konsep nilai Shapley [15]. Secara konseptual, nilai SHAP dapat dianggap sebagai kontribusi marginal rata-rata suatu fitur berdasarkan semua kemungkinan kombinasi fitur lainnya. Kontribusi ini bisa bersifat positif atau negatif. Nilai positif memperkuat proses prediksi, sedangkan nilai negatif melemahkannya[9]. Dengan kata lain, SHAP memberikan pemahaman mengenai pentingnya setiap fitur pada identifikasi sample [16]. SHAP juga dilakukan pada tahap evaluasi model. Shapley mengukur nilai kontribusi yang diberikan setiap pemain dalam permainan. Dengan menerapkan komponen ini, nilai SHAP mengukur seberapa besar fitur berkontribusi terhadap output model [17]. SHAP menghitung kontribusi masing-masing fitur terhadap hasil prediksi dengan menilai rata-rata kontribusi marjinal fitur tersebut dalam berbagai kombinasi subset fitur yang mungkin.

$$\phi_i = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(|N|-|S|-1)!}{|N|!} [f(S \cup \{i\}) - f(S)] \quad (2)$$

Nilai ϕ_i merepresentasikan nilai SHAP untuk fitur ke- i , yaitu besaran kontribusi fitur tersebut terhadap hasil prediksi model. Simbol $S \subseteq N \setminus \{i\}$ menunjukkan himpunan subset dari semua fitur N kecuali fitur i , sedangkan $|S|$ adalah jumlah fitur yang terdapat dalam subset S , dan $|N|$ adalah jumlah total fitur dalam model. Fungsi $f(S)$ merepresentasikan prediksi model ketika hanya menggunakan subset fitur S , sementara $f(S \cup \{i\})$ menunjukkan prediksi model ketika fitur i ditambahkan ke subset S . Perbedaan antara $f(S \cup \{i\})$ dan $f(S)$ digunakan untuk mengukur kontribusi spesifik fitur i terhadap perubahan output model, yang menjadi dasar perhitungan nilai SHAP.

2.8 Klasifikasi XGBoost

Setelah fitur yang paling relevan dipilih, tahap selanjutnya adalah membangun model klasifikasi menggunakan algoritma XGBoost (Extreme Gradient Boosting). Model dilatih menggunakan dataset hasil seleksi fitur dan dilakukan proses optimasi parameter untuk meningkatkan performa model. Proses pelatihan dilakukan pada data latih (training set), sementara data uji (test set) digunakan untuk mengukur sejauh mana model mampu menggeneralisasi pola churn pelanggan. XGBoost (Extreme Gradient Boosting) adalah pengembangan lebih lanjut dari algoritma Gradient Tree Boosting berbasis ensemble, yang dapat secara efektif menangani masalah machine learning dengan skala yang lebih besar [18]. Algoritma ini bekerja dengan menggabungkan beberapa pohon keputusan sebelumnya (CART). Keunggulan utama XGBoost terletak pada kemampuannya beradaptasi dalam berbagai kondisi, yang diperoleh melalui penyempurnaan dari algoritma sebelumnya [4]. Algoritma ini banyak dipilih oleh data scientist karena kecepatan eksekusinya yang tinggi berkat perhitungan out-of-core [19]. Model dikembangkan menggunakan algoritma XGBoost karena algoritme ini mampu mengatasi data yang tidak seimbang serta memiliki kinerja prediksi yang unggul. Pelatihan model dilakukan menggunakan data yang telah diseimbangkan serta fitur-fitur yang telah diseleksi. XGBoost mengoptimalkan kinerjanya dengan meminimalkan fungsi objektif berikut:

$$L^{(t)} = \sum_{i=1}^n l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t) \quad (3)$$

$L^{(t)}$ merupakan fungsi objektif total pada iterasi ke- t yang digunakan untuk mengukur kinerja model pada tahap tersebut. Variabel n menunjukkan jumlah data latih yang digunakan, sedangkan y_i merepresentasikan nilai aktual atau target dari sampel ke- i . Notasi $\hat{y}_i^{(t-1)}$ adalah nilai prediksi yang dihasilkan pada iterasi sebelumnya ($t-1$), sementara $f_t(x_i)$ adalah fungsi prediktor—dalam konteks XGBoost biasanya berupa pohon keputusan yang ditambahkan pada iterasi ke- t . Fungsi loss $l(y, \hat{y})$ digunakan untuk mengukur tingkat kesalahan prediksi, misalnya menggunakan log-loss untuk klasifikasi atau squared error untuk regresi. Selain itu, $\Omega(f_t)$ merupakan fungsi regulasi (regularization) yang berperan mengontrol kompleksitas model agar tidak overfitting dengan memberikan penalti terhadap model yang terlalu rumit.

Nilai gain bisa ditentukan untuk penentuan splitting node. Berikut ini adalah rumus untuk mencari nilai gain pada XGboost:

$$L_{split} = \frac{1}{2} \left[\frac{(\sum_{i \in L} g_i)^2}{\sum_{i \in L} h_i + \lambda} + \frac{(\sum_{i \in R} g_i)^2}{\sum_{i \in R} h_i + \lambda} - \frac{(\sum_{i \in I} g_i)^2}{\sum_{i \in I} h_i + \lambda} \right] - \gamma \quad (4)$$

g_i merupakan turunan pertama dari fungsi loss untuk instance ke- i , yang digunakan untuk mengukur arah dan besarnya perubahan prediksi yang diperlukan agar mendekati nilai aktual. Sementara itu, h_i merepresentasikan turunan kedua dari fungsi loss, yang memberikan informasi tentang kelengkungan fungsi loss dan membantu dalam menentukan besar langkah pembaruan model secara lebih akurat. Parameter λ digunakan sebagai regularisasi L2 untuk menghindari overfitting dengan memberikan penalti terhadap bobot model yang terlalu besar. Sedangkan γ adalah parameter regularisasi tambahan yang berfungsi menghukum kompleksitas model, khususnya dalam proses pembentukan pohon keputusan, sehingga hanya pembelahan (split) yang memberikan peningkatan signifikan pada performa model yang akan dipilih.

2.9 Evaluasi

Confusion matrix adalah metode yang digunakan untuk mengevaluasi kinerja model klasifikasi dengan membandingkan hasil prediksi model terhadap label asli. Matriks ini menunjukkan jumlah prediksi yang benar maupun salah untuk setiap kelas. Confusion matrix terdiri dari empat elemen utama, yaitu True Positive (TP), True Negative (TN), False Positive (FP), dan False Negative (FN). Dalam penelitian ini, kinerja model diukur menggunakan metrik seperti akurasi, presisi, recall, dan F1-score [20]. Pengujian dilakukan dengan menggunakan beberapa metrik evaluasi yang umum dalam masalah klasifikasi biner, untuk menilai seberapa baik model dalam memprediksi churn pelanggan. Evaluasi dilakukan terhadap dua model: Model A (XGBoost tanpa seleksi fitur) dan Model B (XGBoost dengan seleksi fitur menggunakan Boruta + SHAP). Adapun metrik evaluasi yang digunakan dalam penelitian ini dijelaskan sebagai berikut.

Tabel 2. Confusion Matrix

	Prediksi Positif	Prediksi Negatif
Aktual Positif	True Positive (TP)	False Negative (FN)
Aktual Negatif	False Positive (FP)	True Negative (TN)



True Positive (TP) adalah jumlah memprediksi hasil positif dan itu benar, False Positive (FP) adalah jumlah memprediksi hasil negatif dan itu benar, False Negative (FN) adalah jumlah memprediksi hasil negative dan itu salah, True Negative (TN) adalah jumlah memprediksi hasil negative dan itu benar [21].

2.9.1 Accuracy

Accuracy merupakan gambaran seberapa akurat model klasifikasi yang benar [22]. Metrik ini menunjukkan proporsi prediksi yang benar dari keseluruhan data yang diuji, baik untuk kelas positif maupun negatif.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{5}$$

2.9.2 Precision

Presi digunakan untuk mengukur ketepatan prediksi positif, yaitu jumlah prediksi positif yang benar terhadap jumlah total prediksi positif. Rumus berikut ini dapat digunakan untuk menghitung nilai presisi [23]. Rumus perhitungannya adalah:

$$Precision = \frac{TP}{TP+FP} \tag{6}$$

Presi yang tinggi menunjukkan bahwa model lebih andal dalam memprediksi kelas positif tanpa banyak kesalahan.

2.9.3 Recall

Recall mengukur persentase instance positif yang berhasil ditemukan [24]. Mengukur seberapa baik model dalam mendeteksi semua kasus positif yang sebenarnya. Hal ini penting dalam konteks di mana melewatkan kasus positif (False Negative) harus dihindari. Rumusnya adalah:

$$Recall = \frac{TP}{TP+FN} \tag{7}$$

Nilai recall yang tinggi menandakan bahwa model mampu menangkap sebagian besar dari data positif yang ada.

2.9.4 F1-Score

F1-Score merupakan rata-rata harmonis dari precision dan recall [25]. Metode ini sangat berguna ketika terdapat ketidakseimbangan antar kelas. F1-Score dihitung menggunakan rumus:

$$F_1 \text{ score} = \frac{2 \cdot TP}{2 \cdot TP + FP + FN} = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \tag{8}$$

Nilai F1-Score yang tinggi menunjukkan keseimbangan yang baik antara kemampuan model dalam mendeteksi kasus positif dan ketepatannya dalam melakukan prediksi positif.

3. HASIL DAN PEMBAHASAN

3.1 Pengecekan dan Penanganan Missing Value

Pengecekan dilakukan untuk mengidentifikasi apakah terdapat data yang tidak lengkap dalam dataset. Pada dataset yang digunakan disini missing value ditemukan pada kolom TotalCharges yang awalnya bertipe data string (object) karena terdapat nilai kosong. Cara menanganinya dengan cara mengonversi kolom TotalCharges menjadi numerik (float), nilai yang tidak bisa dikonversi seperti string kosong akan diubah menjadi NaN(Not a Number) yang berarti Missing Value, setelah menjadi NaN baris yang mengandung missing value kemudian dihapus dan dibersihkan.

3.2 Encoding

Setelah membersihkan data dari missing value, selanjutnya melakukan encoding pada kolom Churn menggunakan Label Encoding yang mengubah nilai Yes menjadi 1 dan No menjadi 0. Proses transformasi nilai kategorikal menjadi numerik dilakukan sebagaimana ditunjukkan pada Tabel 3.

Tabel 3. Sebelum Encoding

Fitur	Kategori	Nilai Encode
Gender	Female, Male	0, 1
Partner	No, Yes	0, 1
Internet Service	DSL, Fiber Optic, No	0, 1, 2
Churn	No, Yes	0, 1

Setelah dilakukan proses encoding, seluruh fitur kategorikal pada dataset telah diubah menjadi bentuk numerik agar dapat diproses oleh algoritma machine learning. Seperti yang ditunjukkan pada Table 3, fitur seperti Gender, Partner, dan Churn hanya memiliki dua kategori, sehingga masing-masing dikonversi menjadi nilai 0 dan 1. Sementara

itu, fitur Internet Service yang memiliki lebih dari dua kategori (DSL, Fiber Optic, No) dikodekan menjadi 0, 1, dan 2. Transformasi data menjadi bentuk numerik yang telah diencoding ditunjukkan pada Table 4

Tabel 4. Sesudah Encoding

Customer ID	Gender	Partner	InternetService	Churn
001	0	1	0	0
002	1	0	1	1
003	0	0	2	0

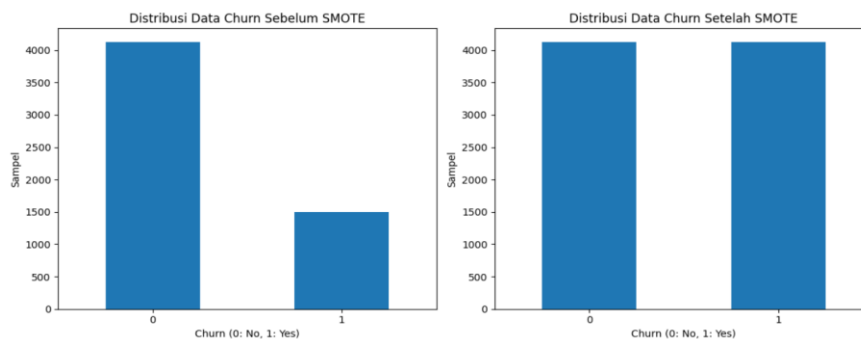
Tabel 4 menunjukkan hasil dari proses encoding pada data kategorikal. Setiap nilai kategorikal telah berhasil dikonversi menjadi format numerik. Sebagai contoh, nilai Gender yang sebelumnya berupa *Female* dan *Male* kini direpresentasikan dengan angka 0 dan 1. Fitur Partner juga mengalami konversi serupa, dengan *No* menjadi 0 dan *Yes* menjadi 1. Sementara itu, fitur InternetService yang memiliki lebih dari dua kategori — *DSL*, *Fiber Optic*, dan *No* — dikodekan menjadi 0, 1, dan 2 secara berurutan. Kolom Churn sebagai target label juga telah dikonversi dari *No* dan *Yes* menjadi 0 dan 1 untuk selanjutnya digunakan dalam proses klasifikasi menggunakan algoritma XGBoost.

3.3 Data Splitting

Setelah dilakukan proses preprocessing, data dibagi menjadi dua bagian utama, yaitu data latih (training set) dan data uji (testing set). Dari 7032 data dibagi $0.8 \times 7032 = 5625,6 \approx 5625$ data untuk data latih dan $0.2 \times 7032 = 1406,4 \approx 1407$ data untuk data uji.

3.4 Data Balancing

Pada tahap ini, metode SMOTE diterapkan pada data latih untuk mengatasi ketidakseimbangan kelas. tujuannya adalah membantu model belajar dari data yang seimbang antara kelas mayoritas dan minoritas. Untuk menyeimbangkan data Churn, dilakukan SMOTE dengan hasil sebagai berikut.



Gambar 2. Distribusi Data Churn Sebelum dan Setelah Churn

Gambar 2 menunjukkan distribusi data churn sebelum dan sesudah penerapan metode Synthetic Minority Over-sampling Technique (SMOTE). Sebelum SMOTE, data menunjukkan ketidakseimbangan kelas yang signifikan, dengan jumlah pelanggan yang tidak churn (label 0) jauh lebih banyak dibandingkan pelanggan yang churn (label 1). Setelah SMOTE diterapkan, jumlah sampel pada kedua kelas menjadi seimbang, sehingga model dapat dilatih secara lebih adil tanpa bias terhadap mayoritas kelas.

3.5 Klasifikasi XGBoost

Dalam XGBoost, turunan pertama dari fungsi loss terhadap prediksi saat ini disebut sebagai gradien (g), sedangkan turunan keduanya disebut sebagai hessian (h). Berikut proses perhitungan fungsi kerugian (loss function) yang dioptimalkan oleh XGBoost melalui metode gradient boosting.

Tabel 5. Perhitungan XGBoost

y true	y pred	gradien (g)	hessian (h)	Region
0	0.132	0.132	0.1146	Left
0	0.132	0.132	0.1146	Left
1	0.132	-0.868	0.1146	Left
0	0.132	0.132	0.1146	Left
0	0.132	0.132	0.1146	Left
0	0.132	0.132	0.1146	Left
0	0.132	0.132	0.1146	Left
0	0.132	0.132	0.1146	Left
0	0.132	0.132	0.1146	Left
0	0.132	0.132	0.1146	Left
0	0.132	0.132	0.1146	Left

Pada Tabel 5, ditampilkan data awal hasil prediksi dari model XGBoost, yang meliputi nilai aktual (y_{true}), prediksi awal (y_{pred}), serta nilai gradien (g) dan hessian (h) untuk setiap data observasi. Nilai prediksi awal untuk seluruh data adalah konstan (0.132), yang mencerminkan output awal model sebelum pohon dibentuk. Berdasarkan nilai tersebut Untuk instance dengan $y_{true} = 0$, gradien dihitung sebagai $g = 0.132 - 0 = 0.132$. Untuk instance dengan $y_{true} = 1$, gradien menjadi negatif: $g = 0.132 - 1 = -0.868$. Hessian untuk semua baris adalah $h = 0.132 * (1 - 0.132) \approx 0.1146$. Kolom Region menyatakan bahwa semua data pada iterasi ini berada di satu node (region) yang sama, yakni Left, sebelum dilakukan pemisahan berdasarkan gain.

3.6 Hasil Evaluasi

3.6.1 Confusion Matrix

Pada penelitian ini, dilakukan perbandingan performa antara dua model klasifikasi menggunakan algoritma XGBoost Model A menggunakan semua fitur tanpa proses seleksi fitur, Model B menggunakan fitur hasil seleksi dengan algoritma Boruta SHAP. Confusion matrix digunakan untuk mengevaluasi performa klasifikasi dengan membandingkan antara hasil prediksi dan label aktual. Matrix ini membantu mengidentifikasi jenis kesalahan yang dilakukan model, seperti false positive (FP) dan false negative (FN), yang sangat penting dalam konteks prediksi churn.

Tabel 6. Confusion Matix

Model		Pred 0	Pred 1
Model A	True 0	788	245
	True 1	105	269
Model B	True 0	819	214
	True 1	127	247

Tabel 6 menyajikan confusion matrix dari Model A dan Model B yang menunjukkan jumlah prediksi benar dan salah untuk masing-masing kelas. Pada Model A, dari total data uji, terdapat 788 data kelas 0 (tidak churn) yang diprediksi dengan benar, sementara 245 data kelas 0 salah diklasifikasikan sebagai churn. Untuk kelas 1 (churn), terdapat 269 data yang diprediksi dengan benar, dan 105 data salah diklasifikasikan sebagai tidak churn. Sementara itu, Model B menunjukkan peningkatan pada prediksi kelas 0 dengan 819 data yang diprediksi benar dan 214 salah prediksi. Namun, pada kelas 1, jumlah prediksi benar menurun menjadi 247, dengan 127 data churn salah diklasifikasikan sebagai tidak churn. Hasil ini mengindikasikan bahwa Model B lebih baik dalam mengenali kelas mayoritas (tidak churn), namun sedikit menurun dalam mengidentifikasi kelas minoritas (churn), jika dibandingkan dengan Model A.

3.6.2 Hasil Akurasi, Presisi, Recall, dan F1-Score

Dalam penelitian ini, digunakan beberapa metrik evaluasi, yaitu akurasi, precision, recall, dan F1-score. Akurasi digunakan untuk memberikan gambaran umum tentang seberapa besar proporsi prediksi yang benar terhadap seluruh data. Namun, karena data churn bersifat tidak seimbang, akurasi saja tidak cukup merepresentasikan performa model secara menyeluruh. Oleh karena itu, precision, recall, dan F1-score juga dihitung untuk masing-masing kelas guna memberikan penilaian yang lebih mendalam, terutama pada performa model dalam mendeteksi kelas minoritas (churn).

Table 7. Hasil Akurasi

Model A	Model B
0.751244	0.757640

Tabel 7 menunjukkan perbandingan hasil akurasi antara dua model yang diuji. Model A memperoleh akurasi sebesar 0.7512, sedangkan Model B menunjukkan peningkatan performa dengan akurasi sebesar 0.7576. Hasil ini mengindikasikan bahwa Model B memiliki kemampuan klasifikasi yang sedikit lebih baik dibandingkan Model A dalam mengidentifikasi pelanggan churn, meskipun perbedaannya relatif kecil. Selain akurasi, dilakukan evaluasi menggunakan tiga metrik utama, yaitu precision, recall, dan F1-score. Metrik-metrik ini dihitung secara terpisah untuk masing-masing kelas (kelas 0 = tidak churn, kelas 1 = churn).

Tabel 8. Hasil Precision, Recall, F1-Score

Kelas	Metrik	Model A	Model B
0	Precision	0.88	0.87
0	Recall	0.76	0.79
0	F1-Score	0.82	0.83
1	Precision	0.52	0.54
1	Recall	0.72	0.66
1	F1-Score	0.61	0.59

Berdasarkan hasil evaluasi performa model yang ditunjukkan pada Tabel 8, kedua model dibandingkan menggunakan metrik precision, recall, dan F1-score untuk masing-masing kelas (kelas 0: tidak churn, kelas 1: churn). Pada kelas 0, Model A memiliki precision sebesar 0.88 dan recall sebesar 0.76, menghasilkan F1-score sebesar 0.82. Sementara itu, Model B menunjukkan precision sebesar 0.87 dan recall yang sedikit lebih tinggi, yaitu 0.79, dengan F1-score sebesar 0.83. Hal ini menunjukkan bahwa Model B lebih baik dalam mendeteksi pelanggan yang tidak churn, meskipun dengan sedikit penurunan pada precision. Untuk kelas 1 (churn), Model A memiliki precision sebesar 0.52 dan recall sebesar 0.72, menghasilkan F1-score sebesar 0.61. Sebaliknya, Model B menunjukkan peningkatan precision menjadi 0.54, namun mengalami penurunan pada recall menjadi 0.66, sehingga menghasilkan F1-score yang sedikit lebih rendah, yaitu 0.59. Meskipun demikian, kedua model masih menghadapi tantangan dalam mendeteksi kelas minoritas secara optimal, yang tercermin dari nilai precision dan F1-score yang lebih rendah dibandingkan kelas mayoritas.

3.7 Pembahasan

Hasil evaluasi yang ditunjukkan pada Tabel 6, Tabel 7, dan Tabel 8 mengindikasikan bahwa penerapan feature selection menggunakan Boruta-SHAP pada algoritma XGBoost (Model B) memberikan pengaruh berbeda terhadap performa klasifikasi churn pelanggan dibandingkan model tanpa seleksi fitur (Model A). Secara umum, Model B menunjukkan peningkatan akurasi dari 0.7512 menjadi 0.7576. Peningkatan ini konsisten dengan temuan studi [5] dan [3], yang melaporkan bahwa XGBoost mampu memberikan kinerja superior pada klasifikasi churn setelah dilakukan optimisasi atau penyesuaian fitur. Berdasarkan confusion matrix, Model B lebih unggul dalam mendeteksi kelas mayoritas (tidak churn) dengan peningkatan recall dari 0.76 menjadi 0.79 dan F1-score dari 0.82 menjadi 0.83. Hal ini sejalan dengan karakteristik XGBoost yang mampu mempelajari pola dari data mayoritas dengan baik [2], [4]. Namun, performa Model B pada kelas minoritas (churn) menunjukkan penurunan recall dari 0.72 menjadi 0.66, meskipun precision meningkat dari 0.52 menjadi 0.54. Fenomena ini sejalan dengan permasalahan yang sering dihadapi pada dataset tidak seimbang, di mana proses seleksi fitur dapat menghilangkan beberapa atribut minor yang relevan bagi identifikasi kelas minoritas, sebagaimana dijelaskan dalam studi [6], [7].

Metode Boruta-SHAP yang digunakan pada penelitian ini secara teoritis bertujuan mempertahankan fitur-fitur yang benar-benar relevan sekaligus memberikan interpretasi kontribusi fitur melalui nilai SHAP [10], [11]. Hasil penelitian ini menunjukkan bahwa meskipun pendekatan ini meningkatkan efisiensi model dan kemampuan deteksi kelas mayoritas, tantangan tetap muncul pada deteksi kelas minoritas. Temuan ini konsisten dengan studi [11], yang melaporkan bahwa integrasi SHAP dapat mengidentifikasi fitur kunci secara efektif, tetapi performa terhadap kelas minoritas memerlukan dukungan tambahan seperti penyesuaian threshold atau metode penyeimbangan data lanjutan. Dengan demikian, penelitian ini mengonfirmasi temuan sebelumnya bahwa XGBoost merupakan algoritma yang andal untuk prediksi churn dan penerapan Boruta-SHAP mampu meningkatkan efisiensi serta interpretabilitas model. Namun, hasil juga menegaskan perlunya strategi tambahan, seperti pengaturan bobot kelas atau kombinasi dengan metode balancing data seperti SMOTE, untuk mengoptimalkan kinerja deteksi churn pada kelas minoritas, sebagaimana disarankan oleh penelitian [5].

4. KESIMPULAN

Hasil penelitian ini menunjukkan bahwa algoritma XGBoost memiliki kinerja yang baik dalam klasifikasi churn pada dataset telekomunikasi, khususnya setelah penerapan teknik penyeimbangan data SMOTE. Integrasi metode Boruta dan SHAP pada XGBoost (Model B) tidak hanya meningkatkan akurasi model dari 0,7512 menjadi 0,7576, tetapi juga memberikan interpretasi yang lebih jelas mengenai kontribusi masing-masing fitur terhadap prediksi churn. Model B lebih unggul dalam mendeteksi kelas mayoritas (tidak churn), yang ditunjukkan dengan peningkatan recall dari 0,76 menjadi 0,79 dan F1-score dari 0,82 menjadi 0,83. Namun, performa pada kelas minoritas (churn) mengalami penurunan, dengan recall menurun dari 0,72 menjadi 0,66 meskipun precision meningkat dari 0,52 menjadi 0,54. Temuan ini mengindikasikan bahwa masih terdapat tantangan dalam mendeteksi seluruh pelanggan churn secara optimal. Untuk mengatasi tantangan tersebut, penelitian selanjutnya disarankan melakukan penyesuaian decision threshold, penambahan class weight, atau penerapan metode penyeimbangan lanjutan seperti ADASYN maupun ensemble sampling untuk meningkatkan deteksi pada kelas minoritas. Selain itu, pengujian kombinasi XGBoost-Boruta-SHAP dengan algoritma alternatif seperti LightGBM atau CatBoost dapat memberikan wawasan lebih lanjut terkait stabilitas model dan efisiensi komputasi. Penggunaan data real-time serta penambahan variabel eksternal, seperti data perilaku pelanggan atau catatan transaksi terbaru, juga berpotensi meningkatkan akurasi model dan efektivitas strategi retensi pelanggan.

REFERENCES

- [1] J. Ahn, J. Hwang, D. Kim, H. Choi, and S. Kang, "A Survey on Churn Analysis in Various Business Domains," *IEEE Access*, vol. 8, pp. 220816–220839, 2020, doi: 10.1109/ACCESS.2020.3042657.
- [2] S. M. Shrestha and A. Shakya, "A Customer Churn Prediction Model using XGBoost for the Telecommunication Industry in Nepal," *Procedia Comput Sci*, vol. 215, pp. 652–661, 2022, doi: 10.1016/j.procs.2022.12.067.



- [3] K. Peng and Y. Peng, “Research on Telecom Customer Churn Prediction Based on GA-XGBoost and SHAP,” *Journal of Computer and Communications*, vol. 10, no. 11, pp. 107–120, 2022, doi: 10.4236/jcc.2022.1011008.
- [4] E. H. Yulianti, O. Soesanto, and Y. Sukmawaty, “Penerapan Metode Extreme Gradient Boosting (XGBOOST) pada Klasifikasi Nasabah Kartu Kredit,” *JOMTA Journal of Mathematics: Theory and Applications*, vol. 4, no. 1, 2022, doi: <https://doi.org/10.31605/jomta.v4i1.1792>.
- [5] L. N. Wakhidah, A. Khanif Zyen, and B. B. Wahono, “Evaluation of Telecommunication Customer Churn Classification with SMOTE Using Random Forest and XGBoost Algorithms,” *Journal of Applied Informatics and Computing (JAIC)*, vol. 9, no. 1, p. 89, 2025, doi: <https://doi.org/10.30871/jaic.v9i1.8740>.
- [6] X. Chen, X. Qiu, Y. Ma, L. Wang, and L. Fang, “Boruta-XGBoost Electricity Theft Detection Based on Features of Electric Energy Parameters,” *J Phys Conf Ser*, vol. 2290, no. 1, 2022, doi: 10.1088/1742-6596/2290/1/012121.
- [7] M. A. Ganaie, M. Hu, A. K. Malik, M. Tanveer, and P. N. Suganthan, “Ensemble deep learning: A review,” *Eng Appl Artif Intell*, vol. 115, Aug. 2022, doi: 10.1016/j.engappai.2022.105151.
- [8] R. Sinaga and S. Widiyanto, “Understanding Telecommunication Customer Churn: Insights from LightGBM Predictive Modelling and SHAP Feature Interpretation,” *ASEAN Marketing Journal*, vol. 15, no. 1, Jun. 2023, doi: 10.7454/amj.v15i1.1229.
- [9] E. AKKUR and A. C. ÖZTÜRK, “PREDICTING LUNG CANCER USING EXPLAINABLE ARTIFICIAL INTELLIGENCE AND BORUTA-SHAP METHODS,” *Journal of Engineering Sciences*, vol. 27(3), pp. 792–803, 2024, doi: <https://doi.org/10.17780/ksujes.1425483>.
- [10] J. Maan and H. Maan, “Customer Churn Prediction Model using Explainable Machine learning,” *International Journal of Computer Science Trends and Technology*, vol. 11, doi: <https://doi.org/10.48550/arXiv.2303.00960>.
- [11] Y. Li and K. Yan, “Prediction of bank credit customers churn based on machine learning and interpretability analysis,” *Data Science in Finance and Economics*, vol. 5, no. 1, pp. 19–34, 2025, doi: 10.3934/dsfe.2025002.
- [12] Y. Xia, S. Jiang, L. Meng, and X. Ju, “XGBoost-B-GHM: An Ensemble Model with Feature Selection and GHM Loss Function Optimization for Credit Scoring,” *Systems*, vol. 12, no. 7, Jul. 2024, doi: 10.3390/systems12070254.
- [13] S. Keputusan Dirjen Penguatan Riset dan Pengembangan Ristek Dikti, A. Nikmatul Kasanah, U. Pujianto, T. Elektro, F. Teknik, and U. Negeri Malang, “Penerapan Teknik SMOTE untuk Mengatasi Imbalance Class dalam Klasifikasi Objektivitas Berita Online Menggunakan Algoritma KNN,” *masa berlaku mulai*, vol. 1, no. 3, pp. 196–201, 2017, doi: 10.29207/resti.v3i2.945.
- [14] X. Yuan *et al.*, “A novel feature susceptibility approach for a PEMFC control system based on an improved XGBoost-Boruta algorithm,” *Energy and AI*, vol. 12, Apr. 2023, doi: 10.1016/j.egyai.2023.100229.
- [15] A. Farzipour, R. Elmi, and H. Nasiri, “Detection of Monkeypox Cases Based on Symptoms Using XGBoost and Shapley Additive Explanations Methods,” *Diagnostics*, vol. 13, no. 14, Jul. 2023, doi: 10.3390/diagnostics13142391.
- [16] Z. Fan *et al.*, “XGBoost-SHAP-based interpretable diagnostic framework for knee osteoarthritis: a population-based retrospective cohort study,” *Arthritis Res Ther*, vol. 26, no. 1, Dec. 2024, doi: 10.1186/s13075-024-03450-2.
- [17] H. Sahlaoui, E. A. A. Alaoui, A. Nayyar, S. Agoujil, and M. M. Jaber, “Predicting and Interpreting Student Performance Using Ensemble Models and Shapley Additive Explanations,” *IEEE Access*, vol. 9, pp. 152688–152703, 2021, doi: 10.1109/ACCESS.2021.3124270.
- [18] A. A. Saputra, B. N. Sari, C. Rozikin, U. Singaperbangsa, and K. Abstrak, “Penerapan Algoritma Extreme Gradient Boosting (Xgboost) Untuk Analisis Risiko Kredit,” *Jurnal Ilmiah Wahana Pendidikan*, vol. 10, no. 7, pp. 27–36, 2024, doi: 10.5281/zenodo.10960080.
- [19] A. Ibrahim Ahmed Osman, A. Najah Ahmed, M. F. Chow, Y. Feng Huang, and A. El-Shafie, “Extreme gradient boosting (Xgboost) model to predict the groundwater levels in Selangor Malaysia,” *Ain Shams Engineering Journal*, vol. 12, no. 2, pp. 1545–1556, Jun. 2021, doi: 10.1016/j.asej.2020.11.011.
- [20] R. Soelistijadi, T. Dwiati Wismarini, S. Eniyati, and S. I4, “Pemodelan Prediktif Menggunakan Metode Ensemble Learning XGBoost dalam Peningkatan Akurasi Klasifikasi Penyakit Ginjal,” *KESATRIA Jurnal Penerapan Sistem Informasi (Komputer & Manajemen)*, vol. 5, no. 4, pp. 1866–1875, 2024, doi: <https://doi.org/10.30645/kesatria.v5i4.507>.
- [21] H. Apriyani, “Perbandingan Metode Naïve Bayes Dan Support Vector Machine Dalam Klasifikasi Penyakit Diabetes Melitus,” *Journal of Information Technology Ampera*, vol. 1, no. 3, pp. 2774–2121, 2020, doi: 10.51519/journalita.volumel.issuue3.year2020.page133-143.
- [22] T. A. Y. Siswa and N. A. Verdikha, “Komparasi Algoritma Klasifikasi Untuk Menentukan Evaluasi Kinerja Terbaik Pada Status Akreditasi Sekolah/Madrasah Kalimantan Timur Berdasarkan IASP 2020,” *JINTEKS (Jurnal Informatika Teknologi dan Sains)*, vol. Vol. 4 No. 3, pp. 185–192, Aug. 2022, doi: 10.51401/jinteks.v4i3.1807.
- [23] N. Suarna and W. Prihartono, “Analisis Sentimen Ulasan Aplikasi Threads Di Google Playstore Menggunakan Algoritma Naïve Bayes,” *Jurnal Mahasiswa Teknik Informatika*, vol. 8, no. 1, 2024, doi: <https://doi.org/10.29100/jipi.v9i1.4929>.
- [24] I. Habib Kusuma and N. Cahyono, “Analisis Sentimen Masyarakat Terhadap Penggunaan E-Commerce Menggunakan Algoritma K-Nearest Neighbor,” *Jurnal Informatika Jurnal pengembangan IT (JPIT)*, vol. 8, no. 3, 2023, doi: 10.30591/jpit.v8i3.5734.
- [25] C. N. Daiman, A. Y. Rahman, and F. Nudiyansyah, “Klasifikasi Teks Berita Breaking News Di Manggarai Menggunakan Long Short Term Memory (LSTM),” *Jurnal MNEMONIC*, vol. 7, no. 2, 2024, doi: 10.36040/mnemonic.v7i2.9939.