Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer e-ISSN: 2548-964X
Vol. 9, No. 2, Februari 2025, hlm. x-x http://j-ptiik.ub.ac.id

Analisis Klasifikasi Spam Email Menggunakan Metode Extreme Gradient Boosting (XGBoost)

Lintang Gladyza Adinda Putri[1], Satrio A. Wicaksono[2], Bayu Rahayudi[3]

Program Studi Teknologi Informasi, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: [1]gladyza14@ub.ac.id, [2]satrio@ub.ac.id, [3]ubay1@ub.ac.id

Abstrak

Peningkatan penggunaan email telah menyebabkan lonjakan spam yang merugikan, seperti penipuan, phishing, dan iklan tidak sah. Untuk mengatasi hal ini, diperlukan sistem deteksi yang mampu mengklasifikasikan email dengan akurat sebagai spam atau ham. Penelitian ini mengusulkan metode Extreme Gradient Boosting (XGBoost) untuk klasifikasi spam email. Evaluasi dilakukan menggunakan Stratified K-Fold Cross Validation dan Confusion Matrix. Hasil evaluasi Klasifikasi spam email menggunakan metode *Extreme Gradient Boosting* menunjukkan bahwa model yang diusulkan memiliki akurasi sebesar 95,3%, precision 95,1%, recall 95,6%, dan F1-score 95,2%. Analisis confusion matrix mengungkapkan bahwa model berhasil mengklasifikasikan 326 email spam dengan benar (*True Positive*) dan 323 email non-spam dengan benar (*True Negative*), sementara tingkat kesalahan yang tercatat relatif kecil, yaitu 17 email non-spam salah diklasifikasikan sebagai spam (*False Positive*) dan 15 email spam salah diklasifikasikan sebagai non-spam (*False Negative*). Hasil ini menggambarkan keseimbangan yang baik antara kemampuan model untuk mengenali email spam dan menghindari kesalahan klasifikasi pada email non-spam. Secara keseluruhan, hasil analisis matriks evaluasi ini membuktikan bahwa metode *Extreme Gradient Boosting* adalah pendekatan yang efektif dalam mengklasifikasikan spam email.

Kata kunci: *Klasifikasi Spam Email, Machine Learning, Extreme Gradient Boosting (XGBoost), Stratified K-Fold Cross Validation, Confusion Matrix*

Abstract

The increase in email usage has led to a surge in harmful spam, such as scams, phishing, and unauthorized advertisements. To address this, a detection system capable of accurately classifying emails as spam or ham is required. This research proposes the Extreme Gradient Boosting (XGBoost) method for email spam classification. Evaluation is done using Stratified K-Fold Cross Validation and Confusion Matrix. The evaluation results of email spam classification using the Extreme Gradient Boosting method show that the proposed model has an accuracy of 95.3%, precision 95.1%, recall 95.6%, and F1-score 95.2%. Confusion matrix analysis revealed that the model successfully classified 326 spam emails correctly (True Positive) and 323 non-spam emails correctly (True Negative), while the error rate recorded was relatively small, with 17 non-spam emails misclassified as spam (False Positive) and 15 spam emails misclassified as non-spam (False Negative). These results illustrate a good balance between the model's ability to recognize spam emails and avoid misclassification of non-spam emails. Overall, the results of this evaluation matrix analysis prove that the Extreme Gradient Boosting method is an effective approach in classifying email spam.

Keywords: *Email Spam Classification, Machine Learning, Extreme Gradient Boosting (XGBoost), Stratified K-Fold Cross Validation, Confusion Matrix*

1. PENDAHULUAN

Menurut laporan Badan Pusat Statistik (BPS) (2023), sekitar 10,73% pengguna internet di Indonesia menggunakan email sebagai komunikasi utama. Menurut Čavor (2021), peningkatan penggunaan email telah menyebabkan lonjakan spam email secara signifikan yang mengganggu komunikasi, membebani kapasitas jaringan, serta menyulitkan pemisahan antara email penting dan email yang tidak diinginkan. Pelaku spam sering

memanfaatkan email untuk menyebarkan *malware*, *phishing*, dan penipuan digital lainnya (Ma et al., 2020). Dampak ancaman tersebut tercermin pada peningkatan insiden kejahatan siber yang signifikan bahwa terdapat 1.433 insiden siber pada Tahun 2022 (Badan Sandi dan Siber Negara (BSSN), 2022). Sementara menurut laporan Kominfo menunjukkan adanya peningkatan ancaman siber sebesar 40% sejak Tahun 2019(Kementerian Komunikasi dan Informatika, 2024). Hal ini menegaskan perlunya pengembangan sistem deteksi email spam yang efektif untuk melindungi pengguna.

Berbagai pendekatan telah digunakan untuk mengklasifikasikan email sebagai spam atau ham (non-spam). Studi sebelumnya menunjukkan bahwa metode seperti *Naive Bayes*, *SVM*, dan *Random Forest* menghasilkan akurasi tinggi, tetapi masih terdapat ruang untuk peningkatan, terutama dalam menangani dataset besar dan kompleks (Sulochana et al., 2023). Sedangkan menurut penelitian yang dilakukan oleh Sumithra et al. (2022) metode *Extreme Gradient Boosting (XGBoost)* sebagai metode *ensemble learning*, telah terbukti unggul dalam menangani dataset tidak seimbang dengan akurasi mencapai 96,05%. Meskipun demikian, tantangan tetap ada dalam meningkatkan performa klasifikasi spam di lingkungan yang dinamis dan kompleks.

Berdasarkan hal tersebut, penelitian ini berfokus pada pengklasifikasian spam email dengan menerapkan algoritme *XGBoost*. Pendekatan ini diharapkan dapat memberikan kontribusi signifikan dalam bidang *text mining* dengan mengembangkan sistem deteksi spam yang lebih efektif, khususnya dalam mengatasi tantangan klasifikasi yang belum optimal.

2. DASAR TEORI

2.1 Extreme Gradient Boosting

Metode *Extreme Gradient Boosting* yang merupakan salah satu contoh penerapan *ensemble learning*. *XGBoost* dikenal karena efisiensi dan kecepatannya dalam mengatasi dataset besar dan kompleks (Anirudh et al., 2024). Metode *XGBoost* bekerja dengan prinsip *gradient boosting* yang iteratif, dimana setiap pohon yang dibangun akan memperbaiki kesalahan dari pohon sebelumnya. Model ini mengoptimalkan fungsi *loss* untuk meminimalisir kesalahan prediksi dengan menghitung gradien loss pada setiap iterasi menggunakan *binary logistic loss* dengan Persamaan 1:

$$L = -\sum_{i=1}^{n}(y_i \log(p_i) + (1-)\log(1-p_i)) \quad (1)$$

Dalam metode *XGBoost*, prediksi setiap sampel dihitung sebagai penjumlahan dari pohon keputusan yang dibangun setiap iterasi dengan Persamaan 2:

$$\hat{y}i = \sum_{k=1}^{K} fk(Xi) \quad (2)$$

2.2 TF-IDF

TF-IDF (Term Frequency-Inverse Document Frequency) adalah metode untuk menghitung bobot kata yang penting dalam representasi teks, yang memiliki berbagai aplikasi dalam pencarian informasi, penambangan teks, dan bidang terkait lainnya. Algoritma ini bekerja untuk meningkatkan bobot kata-kata tertentu dalam dokumen dan menyusun vektor kata berbobot untuk meningkatkan akurasi representasi teks (Sun, Bao and Bu, 2022). Pada *TF-IDF*, bobot kata dihitung menggunakan dua komponen utama: *TF (Term Frequency)* dan *IDF (Inverse Document Frequency)*. *Term Frequency (TF)* pada Persamaan 3:

$$TF_{i,j} = \frac{n_{i,j}}{\sum_k n_{i,j}} \quad (3)$$

Pada persamaan 3, $n_{i,j}$ merepresentasikan frekuensi kemunculan kata i dalam dokumen j, sedangkan $\sum_k n_{i,j}$ menunjukkan total jumlah kata dalam dokumen j. *Inverse Document Frequency (IDF)* digunakan untuk menentukan seberapa umum atau jarang suatu kata dalam semua dokumen, sebagaimana ditunjukkan dalam Persamaan 4:

$$IDF_i = log\frac{|D|}{1+|j:t_i \in d_j|} \quad (4)$$

Di mana $|D|$ adalah jumlah total dokumen dan $|j: t_i \in d_j|$ adalah jumlah dokumen yang mengandung kata t_i Penggabungan nilai TF dan IDF untuk setiap kata menghasilkan bobot TF-IDF, sebagaimana dijelaskan pada Persamaan 5:

$$TF - IDF = TF \times IDF \quad (5)$$

2.3 SMOTETomek

Salah satu metode yang dapat mengatasi ketidakseimbangan tersebut adalah dengan menggunakan metode *Synthetic minority*

oversampling technique (SMOTE) yang bertujuan untuk menghasilkan sampel kelas minoritas dengan menginterpolasi dua kelas yang berdekatan antara sampel minoritas. Salah satu cara untuk meningkatkan performa klasifikasi adalah dengan menyeimbangkan data dengan menambahkan data minoritas sehingga sebanding dengan data mayoritas (Hairani et al., 2023). Menurut I. Tomek (1976 disitasi oleh Hairani, Anggrawan and Priyanto (2023) salah satu metode untuk menghilangkan *noise* data pada kelas mayoritas adalah dengan *Tomek Links.*

2.4 Confusion Matrix

Confusion matrix merupakan alat yang bermanfaat untuk menilai kinerja model klasifikasi dalam *machine learning.* Matriks ini memberikan visualisasi yang jelas tentang performa model dengan menampilkan jumlah prediksi yang benar dan salah untuk setiap kelas. Dalam klasifikasi dua kelas, *Confusion Matrix* dapat ditunjukkan dengan:

1. *True Positif (TP)* adalah jumlah prediksi positif yang sesuai dengan nilai aktual positif.
2. *False Positif (FP)* adalah jumlah prediksi yang dinyatakan positif, namun nilai aktualnya negatif.
3. *False Negatif (FN)* adalah jumlah prediksi yang dinyatakan negatif, padahal nilai aktualnya positif
4. *True Negatif (TN)* adalah jumlah prediksi negative yang sesuai dengan nilai aktual negatif

Confusion matrix menunjukkan perhitungan matriks evaluasi yang penting untuk menilai kinerja model klasifikasi. Akurasi adalah proporsi total prediksi yang benar dari keseluruhan prediksi yang dibuat. Nilai akurasi dapat diperoleh menggunakan Persamaan 6:

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

Precision menghitung proporsi prediksi positif yang sesuai dengan nilai positif sebenarnya. Nilai *precision* dapat dihitung menggunakan Persamaan 7:

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

Recall, juga dikenal sebagai sensitivitas, mengukur proporsi kasus positif yang benar-benar diidentifikasi oleh model. Nilai *recall* dapat diperoleh menggunakan persamaan 8:

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

F1-Score merupakan rata-rata harmonis antara *precision* dan *recall*, yang digunakan untuk menciptakan keseimbangan antara kedua metrik tersebut. Nilai *F1-Score* dapat diperoleh menggunakan persamaan 9:

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

2.5 K-Fold Cross Validation

Cross Validation adalah metode untuk memperkirakan performa model klasifikasi dengan membagi data menjadi beberapa lipatan (*fold*) untuk pelatihan dan pengujian secara bergantian (Kalra et al., 2022; Soundrapandian & Geetha, 2022). Pada *K-Fold Cross Validation* dengan K=5, setiap fold secara bergantian menjadi data testing, sementara sisanya digunakan untuk data training. Setelah lima iterasi, performa model dievaluasi berdasarkan rata-rata metrik dari semua split, memastikan seluruh data terpakai dan meningkatkan generalisasi model.

Untuk dataset yang tidak seimbang, digunakan *Stratified K-Fold Cross Validation*, yang membagi lipatan berdasarkan distribusi target agar proporsinya konsisten antara data *training* dan *testing*. Metode ini memberikan representasi yang merata untuk setiap kelas dalam setiap *fold*, sehingga mengurangi bias dan menghasilkan evaluasi model yang lebih akurat (Soundrapandian & Geetha, 2022).

3. METODOLOGI PENELITIAN

3.1 Diagram Alur Penelitian

Pada penelitian ini, penulis telah melakukan studi literatur untuk menemukan teori dan konsep yang menjadi pendukung penelitian. Konsep dan teori yang menjadi dasar penelitian berasal dari *paper*, buku, dan jurnal yang berkaitan. Berdasarkan hasil studi literatur, penulis mengidentifikasi masalah penelitian dan merumuskan rumusan masalah. Selanjutnya,

penulis menetapkan tujuan berdasarkan rumusan masalah yang telah ditetapkan. Berdasarkan hasil perumusan masalah dan tujuan, penulis melakukan analisis kebutuhan yang diperlukan dalam proses penelitian. Pada Gambar 1 merupakan alur penelitian yang dilakukan penulis:

Gambar 1. Alur Metodologi Penelitian

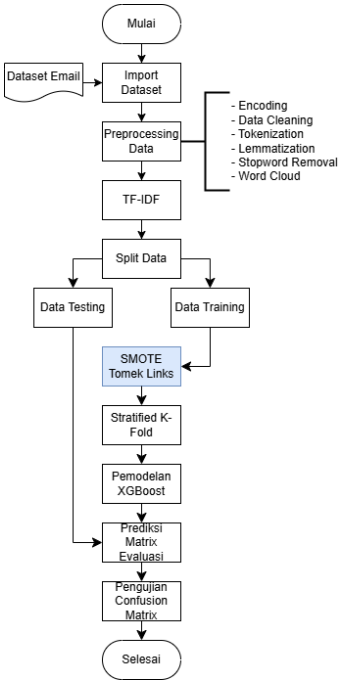
3.2 Analisis Kebutuhan

Setelah melakukan studi literatur, penulis melakukan analisis kebutuhan yang merupakan tahapan untuk mengidentifikasi kebutuhan dan spesifikasi yang harus dipenuhi oleh sistem. Pada tahapan ini, penulis melakukan analisis kebutuhan fungsional dan analisis pengumpulan data.

Analisis kebutuhan fungsional mencangkup identifikasi dan deksripsi fungsi-fungsi utama yang harus dimiliki sistem. Penulis melakukan identifikasi tersebut dengan metode analisis dokumen melalui studi literatur dan analisis terhadap sistem serupa yang sudah ada.

Pengumpulan data merupakan tahapan yang digunakan untuk mengumpulkan informasi yang diperlukan untuk perancangan sistem. Pada tahapan ini, penulis melakukan pencarian dataset yang relevan dengan topik, kemudian mengevaluasi konten dari dataset tersebut. Pada penelitian ini, penulis menggunakan dataset sekunder yang diunduh dari komunitas Kaggle yang diunggah oleh Ashfak Yeafi. Dataset yang digunakan dalam penelitian ini terdiri dari 5573 baris dan 2 kolom, yaitu kolom *message* dan *category*. Kolom *message* berisi teks email yang akan dianalisis, sementara kolom *category* berisi label yang menunjukkan apakah email tersebut termasuk dalam kategori "spam" atau "ham" (non-spam).

3.3 Implementasi



Gambar 2. Alur Perancangan Implementasi

Berdasarkan Gambar 2, penelitian ini menggunakan bahasa pemrograman *Python* dengan menerapkan *library numpy*, *pandas* dan *skicitlearn*. Penelitian ini dimulai dengan mengimpor dataset email, diikuti oleh proses *preprocessing* data untuk memastikan kualitas dan konsistensi. Tahap *preprocessing* mencakup proses *encoding*, *data cleaning*, *tokenization*, *lemmatization*, *stopword removal*, dan visualisasi *word cloud*. Selanjutnya, data teks diubah menjadi representasi numerik menggunakan metode *TF-IDF (Term Frequency-Inverse Document Frequency)*, sehingga informasi penting pada teks dapat diproses oleh algoritme *machine learning*.

Dataset dibagi menjadi 80% untuk data training dan 20% untuk data testing. Pada implementasi *Cross Validation*, penulis melakukan pengujian pada data training dengan teknik *Non-SMOTETomek* dan *undersampling* pada data mayoritas, serta *oversampling SMOTETomek* pada data minoritas. Hasil distribusi data setelah *resampling* diterapkan pada skema *Stratified K-Fold Cross-Validation* untuk menguji model *XGBoost* secara iteratif dengan K=5, yang berarti pembagian data dilakukan sebanyak lima kali. Setelah pemodelan *XGBoost* diterapkan, data testing

digunakan untuk menghasilkan prediksi model, dan nilai matriks evaluasi ditampilkan untuk setiap *fold*. Kemudian evaluasi lebih lanjut dilakukan menggunakan *Confusion Matrix*.

3.4 Evaluasi dan Analisis

Pada tahapan ini, penulis melakukan evaluasi terhadap pengujian *Confusion Matrix* dan *K-Fold Cross Validation*. Analisis pengujian *Confusion Matrix* menghasilkan nilai matriks sesuai keterangan berikut:

1. *True Positif (TP)* menampilkan hasil nilai email spam diklasifikasikan sebagai spam atau kelas positif.
2. *False Positif (FP)* menampilkan hasil nilai email non-spam diklasifikasikan sebagai spam.
3. *False Negatif (FN)* menampilkan hasil nilai email spam diklasifikasikan sebagai non-spam.
4. *True Negatif (TN)* menampilkan hasil nilai email non-spam diklasifikasikan sebagai non-spam atau kelas negatif.

Analisis pengujian *Cross Validation* menghasilkan nilai matriks sebanyak lima *fold* dengan hasil berupa nilai evaluasi matriks, seperti akurasi, *precision*, *recall*, dan *F1-Score*.

3.5 Saran dan Kesimpulan

Pada tahap akhir ini, kesimpulan dibuat berdasarkan temuan dari hasil uji data yang telah dilakukan. Hasil analisis diuraikan secara detail untuk menjawab rumusan masalah yang telah ditentukan di awal penelitian. Selanjutnya, penulis memberikan rekomendasi yang dapat digunakan untuk mengembangkan atau memperbaiki teknik penelitian di masa mendatang. Rekomendasi ini diharapkan dapat memberikan kontribusi signifikan terhadap peningkatan kualitas penelitian dan menawarkan solusi yang lebih efektif dan efisien dalam konteks penelitian yang relevan.

4. HASIL DAN PEMBAHASAN

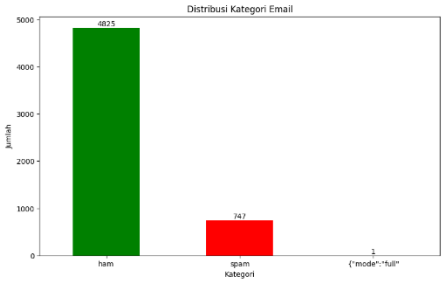
4.1 Hasil Analisis Kebutuhan

Berdasarkan hasil perancangan metodologi, penulis mengidentifikasi analisis kebutuhan yang dibutuhkan dalam pemodelan klasifikasi spam email menggunakan metode *XGBoost*. Hasil analisis kebutuhan fungsional dapat dilihat pada Tabel 1.

Tabel 1. Hasil Analisis Kebutuhan Fungsional

Kebutuhan	Definisi
Membuat Dataset	Memuat dataset email yang berisi teks pesan dan label kategori (spam/ham)
Melakukan konversi label	Menjadikan label ”spam” menjadi angka 0 dan ”ham” menjadi angka 1
Melakukan preprocessing data	membersihkan data (data cleaning), tokenisasi (tokenization), penghapusan kata umum (stopword removal), dan lemmatisasi (lemmatization)
Melakukan Ekstraksi Fitur	Mengonversi teks email menjadi vektor numerik menggunakan TF-IDF
Menangani ketidakseimbangan data	Menyeimbangkan hasil data preprocessing dengan teknik Undersampling dan SMOTETomek
Membagi dataset	Membagi dataset menjadi data training 80% dan data testing 20%
Melakukan pelatihan model	Melakukan implementasi model XGBoost pada data latih
Melakukan pengujian prediksi matriks evaluasi	Melakukan prediksi hasil akurasi, precision, recall, dan F1-Score XGBoost berdasarkan data uji
Evaluasi model	Melakukan evaluasi dengan menggunakan Stratified K-Fold Cross Validation dan Confusion Matrix untuk menghitung akurasi, precision, recall,

	serta F1-score untuk memberikan wawasan yang lebih tepat mengenai kinerja model.
Visualisasi Hasil	Menampilkan grafik Word Cloud dalam mengklasifikasi kata mengandung spam atau ham



Gambar 3 Distribusi Data Per Kategori

Pada Gambar 3 menunjukkan hasil pengumpulan data. Data yang berhasil dikumpulkan mencakup elemen-elemen utama yang dibutuhkan untuk pengembangan sistem. Dataset penelitian ini terdiri dari 5573 baris dengan dua kolom, yaitu *message* dan *category*. Data tersebut berisi 747 baris tergolong spam, 4825 baris tergolong ham, dan 1 baris tanpa kategori.

4.2 Visualisasi Wordcloud

Visualisasi *Word Cloud* dilakukan untuk menampilkan kata-kata yang sering muncul dalam pesan email pada dataset. *Word cloud* menunjukkan kata-kata yang digunakan dengan frekuensi tertinggi dalam bentuk visual di mana ukuran kata mencerminkan seberapa sering kata tersebut muncul di pesan. Semakin besar ukuran kata, semakin sering kata tersebut muncul dalam pesan. Hasil visualiasasi *Word Cloud* dapat dilihat pada Gambar 4 dan 5.

Gambar 4. Word Cloud Kata Spam

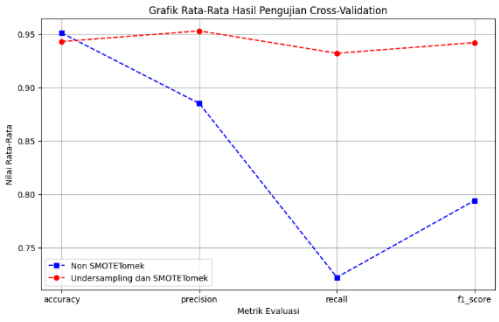
Gambar 5. Word Cloud Kata Ham

4.3 Hasil Pengujian 5-Fold Cross Validation

Tabel 2. Hasil Stratified Cross Validation

Matriks	Non SMOTETomek	Undersampling dan SMOTETomek
Akurasi	95,1%	94,3%
Precision	88,5%	95,3%
Recall	72,2%	92,2%
F1-Score	79,4%	94,2%

Berdasarkan Tabel 2, menunjukkan pengujian *Stratified K-Fold Cross Validation* tanpa menggunakan teknik *SMOTETomek.* Nilai *recall* yang rendah menunjukkan adanya kesenjangan yang signifikan antara akurasi dan performa pada kelas positif atau email spam, yang berdampak pada validitas model dalam mendeteksi email spam. Untuk mengatasi hal tersebut, dilakukan pengujian ulang dengan menerapkan teknik *undersampling* pada kelas mayoritas dan *oversampling SMOTETomek* pada kelas minoritas yang disesuaikan dengan distribusi data setelah proses *undersampling.* Hasil pengujian ini menunjukkan peningkatan performa yang mencerminkan keseimbangan antara deteksi kelas positif dan negatif.

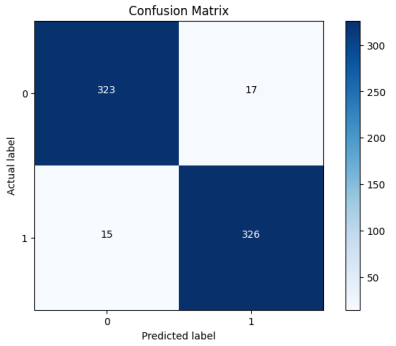


Gambar 6 Hasil Pengujian Stratified K-Fold Cross Validation

Berdasarkan Gambar 6, hasil pengujian *Stratified K-Fold Cross Validation* dengan teknik *SMOTETomek* menunjukkan performa yang konsisten dan stabil dengan rata-rata matriks yang lebih tinggi dibandingkan metode

sebelumnya. *SMOTETomek* mampu mencapai akurasi hingga 94%, dengan *precision*, *recall*, dan *F1-score* yang menunjukkan stabilitas performa yang baik. Meskipun teknik *Non-SMOTETomek* memiliki nilai akurasi yang tinggi, nilai *recall* hanya mencapai 72,2%, yang mengindikasikan bahwa model masih kesulitan mengenali email spam, menyebabkan beberapa email spam diklasifikasikan sebagai non-spam *(False Negative*). Penerapan *SMOTETomek* efektif dalam menyelesaikan masalah ini dengan menggabungkan sintesis data minoritas dan penghapusan sampel redundan, sehingga meningkatkan deteksi kelas positif (spam) secara signifikan. Dengan demikian, pemilihan teknik penanganan ketidakseimbangan data seperti *SMOTETomek* terbukti berpengaruh secara signifikan terhadap peningkatan performa model, khususnya dalam mendeteksi kelas positif yang sebelumnya sulit dikenali.

4.4 Hasil Pengujian Confusion Matrix



Gambar 7 Confusion Matrix Undersampling dan SMOTETomek

Berdasarkan hasil evaluasi menggunakan *confusion matrix* pada Gambar 7, model dengan penerapan teknik *undersampling* dan *SMOTETomek* menunjukkan performa yang optimal dalam mendeteksi kelas spam dan non-spam. Hal ini ditunjukkan oleh nilai *True Positive (TP)* sebesar 326 dan *True Negative (TN)* sebesar 323, yang mencerminkan tingkat akurasi tinggi dalam klasifikasi kedua kelas. Meskipun masih terdapat kesalahan *dengan False Positive (FP)* sebanyak 17 dan *False Negative (FN)* sebanyak 15, jumlah ini relatif kecil dibandingkan dengan total data yang diuji, sehingga kesalahan prediksi dapat dikategorikan minim.

Tabel 3. Matriks Evaluasi Confusion Matrix

Matriks	Nilai
Akurasi	95,3%
Precision	95,1%
Recall	95,6%
F1-Score	95,2%

Berdasarkan Tabel 3, nilai *precision* yang tinggi menunjukkan kemampuan model dalam menghindari kesalahan klasifikasi pada kelas non-spam, sedangkan nilai *recall* yang tinggi menunjukkan efektivitas model dalam mendeteksi kelas spam secara konsisten. Kombinasi *undersampling* dan *SMOTETomek* terbukti berhasil menangani ketidakseimbangan data dengan menyeimbangkan distribusi antara kelas mayoritas dan minoritas, sehingga menghasilkan model yang lebih stabil dan andal. Hal ini menggarisbawahi bahwa pemilihan metode penanganan ketidakseimbangan data seperti *SMOTETomek* berperan signifikan dalam meningkatkan performa model, terutama dalam memastikan akurasi, stabilitas, dan efektivitas deteksi kelas positif (spam) dan negatif (non-spam) secara optimal.

5. KESIMPULAN DAN SARAN

5.1 KESIMPULAN

Hasil evaluasi Klasifikasi spam email menggunakan metode *Extreme Gradient Boosting* menunjukkan bahwa model yang diusulkan memiliki akurasi sebesar 95,3%, precision 95,1%, recall 95,6%, dan F1-score 95,2%. Analisis confusion matrix mengungkapkan bahwa model berhasil mengklasifikasikan 326 email spam dengan benar (*True Positive*) dan 323 email non-spam dengan benar (*True Negative*), sementara tingkat kesalahan yang tercatat relatif kecil, yaitu 17 email non-spam salah diklasifikasikan sebagai spam (*False Positive*) dan 15 email spam salah diklasifikasikan sebagai non-spam (*False Negative*). Nilai *precision* yang tinggi mengindikasikan bahwa model mampu meminimalkan kesalahan dalam mengklasifikasikan email non-spam sebagai spam, sehingga memberikan keandalan tinggi dalam menjaga akurasi pengklasifikasian kelas negatif (non-spam). Di sisi lain, nilai *recall* yang juga tinggi mencerminkan kemampuan model dalam mendeteksi email spam secara konsisten, menunjukkan sensitivitas model terhadap kelas

positif (spam). Hasil ini menggambarkan keseimbangan yang baik antara kemampuan model untuk mengenali email spam dan menghindari kesalahan klasifikasi pada email non-spam. Secara keseluruhan, hasil analisis matriks evaluasi ini membuktikan bahwa metode *Extreme Gradient Boosting* adalah pendekatan yang efektif dalam mengklasifikasikan spam email.

5.2 Saran

Untuk penelitian selanjutnya, peneliti dapat membuat aplikasi *client server* atau sistem *real-time* yang dapat langsung menentukan apakah pesan merupakan spam atau tidak dengan menerapkan metode *machine learning*. Selain itu, peneliti dapat meningkatkan nilai matriks evaluasi dengan *tuning hyperparameter* pada model *XGBoost*, seperti *learning rate*, jumlah pohon *(n_estimators*), dan kedalaman pohon *(max_depth*), sehingga model dapat disesuaikan lebih baik dengan karakteristik data, menghasilkan performa yang lebih akurat dan efisien dalam mendeteksi spam.

6. DAFTAR PUSTAKA

Anirudh, S., Radha Nishant, P., Baitha, S., & Dinesh Kumar, K. (2024). An Ensemble Classification Model for Phishing Mail Detection. *Procedia Computer Science*, *233*, 970–978. https://doi.org/10.1016/j.procs.2024.03.286

Badan Pusat Statistik (BPS). (2023). *statistik-telekomunikasi-indonesia-2022*.

Badan Sandi dan Siber Negara (BSSN). (2022). *LANSKAP KEAMANAN SIBER INDONESIA*.

Čavor, I. (2021, February 16). Decision Tree Model for Email Classification. *2021 25th International Conference on Information Technology, IT 2021*. https://doi.org/10.1109/IT51528.2021.9390143

Hairani, H., Anggrawan, A., & Priyanto, D. (2023). Improvement Performance of the Random Forest Method on Unbalanced Diabetes Data Classification Using Smote-Tomek Link. *International Journal on Informatics Visualization*, *7*(1), 258–264. https://doi.org/10.30630/joiv.7.1.1069

Kalra, V., Kashyap, I., & Kaur, H. (2022). Effect of Ensembling over K-fold Cross-Validation with Weighted K-Nearest Neighbour for Classification in Medical Domain. *2022 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing, COM-IT-CON 2022*, 796–800. https://doi.org/10.1109/COM-IT-CON54601.2022.9850498

Kementerian Komunikasi dan Informatika. (2024). *Ancaman Siber Meningkat, 2Wamenkominfo Tekankan Pelindungan Data Pribadi*. https://www.kominfo.go.id/content/detail/55668/siaran-pers-no-243hmkominfo032024-tentang-ancaman-siber-meningkat-wamenkominfo-tekankan-pelindungan-data-pribadi/0/siaran_pers

Ma, T. M., Yamamori, K., & Thida, A. (2020). A Comparative Approach to Naïve Bayes Classifier and Support Vector Machine for Email Spam Classification. *2020 IEEE 9th Global Conference on Consumer Electronics, GCCE 2020*, 324–326. https://doi.org/10.1109/GCCE50665.2020.9291921

Soundrapandian, P. D., & Geetha, S. (2022). Ensemble Learning on a Weak Correlated Android Malware data using Stratified K-Fold. *3rd IEEE 2022 International Conference on Computing, Communication, and Intelligent Systems, ICCCIS 2022*, 187–192. https://doi.org/10.1109/ICCCIS56430.2022.10037646

Sulochana, B. C., Pragada, B. S., Lokesh, K., & Venugopalan, M. (2023). PySpark-Powered ML Models for Accurate Spam Detection in Messages. *2023 2nd International Conference on Futuristic Technologies, INCOFT 2023*. https://doi.org/10.1109/INCOFT60753.2023.10425231

Sumithra, A., Ashifa, A., Harini, S., & Kumaresan, N. (2022). Probability-based Naïve Bayes Algorithm for Email Spam Classification. *2022 International Conference on Computer Communication and Informatics, ICCCI 2022*. https://doi.org/10.1109/ICCCI54379.2022.9740792