Comparative Analysis of Random Forest and XGBoost for Detecting Phishing Websites: A Machine Learning Approach

Yogi Perdana[1*]

[1]*Universitas Jambi, Jambi, Indonesia*

yogi.perdana@unja.ac.id[*]

Received: 20/11/2025 | *Revised: 09/12/2025* | *Accepted: 16/12/2025*

Abstrak

Serangan phishing merupakan salah satu ancaman keamanan siber yang signifikan di era digital dengan lebih dari 300.000 keluhan dilaporkan secara global pada tahun 2023. Di Indonesia, Badan Siber dan Sandi Negara melaporkan trafik anomali terkait *phishing* mencapai 47.231.390 insiden pada tahun 2023, menjadikannya salah satu ancaman terbesar terhadap ekosistem digital nasional. Kompleksitas serangan *phishing* modern yang semakin canggih membutuhkan pendekatan deteksi otomatis berbasis *machine learning* untuk mengatasi keterbatasan metode deteksi manual yang kurang efektif. Penelitian ini menyajikan analisis komparatif algoritma Random Forest dan XGBoost untuk deteksi website *phishing* otomatis menggunakan teknik *machine learning*. Meskipun kedua algoritma telah terbukti efektif dalam domain keamanan siber, perbandingan komprehensif yang mempertimbangkan aspek performa, interpretabilitas, dan efisiensi komputasi dalam konteks deteksi *phishing* masih terbatas, menciptakan *gap* penelitian yang perlu diisi untuk mengoptimalkan sistem deteksi *phishing* nasional. Metodologi penelitian mencakup pengumpulan data, *preprocessing*, implementasi model, optimisasi *hyperparameter* menggunakan *randomized search* dengan *5-fold stratified cross-validation* dan analisis komparatif. Hasil eksperimen menunjukkan XGBoost teroptimasi memberikan performa terbaik dengan akurasi 97,78% dan waktu training 73% lebih cepat, sementara Random Forest menawarkan keunggulan interpretabilitas dengan akurasi 97,65%. Analisis *feature importance* mengungkapkan status sertifikat SSL dan karakteristik *anchor URL* sebagai fitur diskriminatif paling kritis. Penelitian ini menyimpulkan bahwa XGBoost teroptimasi menjadi pilihan yang lebih optimal untuk *deployment production* sistem deteksi *phishing real-time*, sementara Random Forest lebih sesuai untuk skenario yang memerlukan transparansi model. Temuan ini memberikan kontribusi untuk pengembangan sistem deteksi *phishing* nasional yang mendukung program digitalisasi pemerintah Indonesia dan melindungi masyarakat dari ancaman keamanan siber yang terus meningkat.

Kata kunci: Deteksi Phishing, Machine Learning, Random Forest, XGBoost, Keamanan Siber

Abstract

Phishing attacks represent one of the most significant cybersecurity threats in the digital era, with over 300,000 complaints reported globally in 2023. In Indonesia, the National Cyber and Crypto Agency reported anomalous traffic related to phishing reaching 47,231,390 incidents in 2023, making it one of the greatest threats to the national digital ecosystem. The complexity of increasingly sophisticated modern phishing attacks requires machine learning-based automatic detection approaches to overcome the limitations of ineffective manual detection methods. This study presents a comparative analysis of Random Forest and XGBoost algorithms for automatic phishing website detection using machine learning techniques. Although both algorithms have proven effective in the cybersecurity domain, comprehensive comparisons considering aspects of performance, interpretability, and computational efficiency in the context of phishing detection remain limited, creating a research gap that needs to be filled to optimize national phishing detection systems. The research methodology includes data collection, preprocessing, model implementation, hyperparameter optimization using randomized search with 5-fold stratified cross-validation, and comparative analysis. Experimental results demonstrate that optimized XGBoost delivers the best performance with 97.78% accuracy and 73% faster training time, while Random Forest offers interpretability advantages with 97.65% accuracy. Feature importance analysis reveals SSL certificate status and anchor URL characteristics as the most critical discriminative features. This study concludes that optimized XGBoost is the more optimal choice for production deployment of real-time phishing detection systems, while Random Forest is more suitable for scenarios requiring model transparency. These findings contribute to the development of national phishing detection systems that support the Indonesian government's digitalization program and protect the public from increasing cybersecurity threats.

Keywords: Phishing Detection, Machine Learning, Random Forest, XGBoost, Cybersecurity

Pendahuluan

Era tekonologi informasi saat ini telah mengubah lanskap kehidupan masyarakat secara fundamental, dengan teknologi informasi menjadi tulang punggung aktivitas ekonomi dan sosial di seluruh dunia (Fang & Liu, 2024). Indonesia sebagai negara dengan populasi digital terbesar di Asia Tenggara mengalami pertumbuhan signifikan dalam adopsi teknologi digital, dengan penetrasi internet pada tahun 2022-2023 mencapai 78,19 persen dari total populasi Indonesia yang meningkat 1,7% jika dibandingkan dengan data pada 2021-2022 (Kementerian Komunikasi dan Digital, 2023). Program digitalisasi nasional yang digalakkan pemerintah Indonesia telah mendorong transformasi di berbagai sektor, mulai dari layanan publik hingga perdagangan elektronik yang berkontribusi signifikan terhadap pertumbuhan ekonomi digital nasional.

Meskipun pertumbuhan digitalisasi terus meningkat, percepatan digitalisasi ini juga diikuti oleh meningkatnya ancaman keamanan siber yang semakin banyak dan kompleks. Serangan siber telah menjadi ancaman signifikan di era digital, di mana phishing menjadi jenis

serangan yang paling banyak dilaporkan dengan lebih dari 300,000 keluhan (Federal Bureau of Investigation, 2023). Di Indonesia, Badan Siber dan Sandi Negara melaporkan serangan *phishing* merupakan salah satu insiden terbesar yang terjadi pada tahun 2023 dengan total trafik anomali sebanyak 47.231.390 (Badan Siber dan Sandi Negara, 2024). Serangan *phishing* mer upakan teknik rekayasa sosial yang memanfaatkan kelemahan faktor manusia untuk mencuri informasi sensitif melalui website palsu yang menyerupai situs aslinya (Rajalim, 2025).

Kompleksitas serangan *phishing* modern yang semakin canggih membuat deteksi manual menjadi tidak efektif dan membutuhkan pendekatan otomatis berbasis *machine learning* (Safi & Singh, 2023). Lebih lanjut, penelitian tersebut menunjukkan bahwa pemanfaatan *machine learning* sebagai alat deteksi *phising* rata-rata memiliki akurasi di atas 90%. Penelitian lain juga menunjukkan bahwa *machine learning* telah terbukti efektif dalam mengatasi tantangan deteksi phishing karena kemampuannya menganalisis pola kompleks dari berbagai fitur website dan menghasilkan model prediktif yang akurat (Dutta, 2021)

Algoritma ensemble learning seperti Random Forest dan XGBoost telah mendapat perhatian khusus dalam domain *cybersecurity* karena kemampuannya menghasilkan model yang *robust* dan *resistant* terhadap *overfitting* (Amora et al., 2025). Random Forest menggunakan teknik *bootstrap aggregating* untuk mengkombinasikan *multiple decision trees* dan menunjukkan efektivitas tinggi dalam deteksi *phishing* dengan kemampuan menghasilkan *feature importance* yang *interpretable* (Abdeljaber et al., 2014; Ovi et al., 2024). XGBoost sebagai implementasi *optimized* dari *gradient boosting framework* menunjukkan performa yang luar biasa dalam berbagai aplikasi keamanan siber (Abdul Samad et al., 2023).

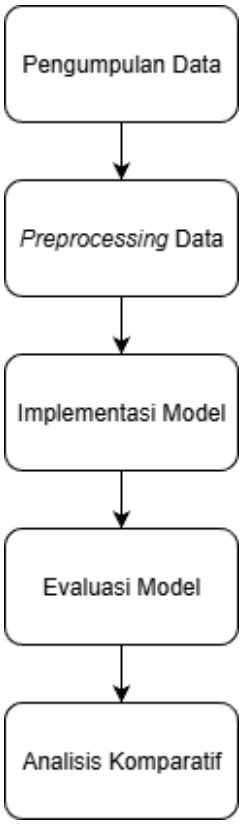
Penelitian terdahulu menunjukkan potensi dari kedua algoritma untuk deteksi phishing. (Abdeljaber et al., 2014) mencapai akurasi 97,14% menggunakan Random Forest, sementara (Abdul Samad et al., 2023) melaporkan XGBoost dapat mencapai akurasi hingga 97,47%. Namun, perbandingan komprehensif yang mempertimbangkan aspek performa, interpretabilitas, dan efisiensi komputasi masih terbatas. (Sahingoz et al., 2019) menemukan Random Forest menunjukkan performa terbaik dengan akurasi 97,98%. Meskipun berbagai penelitian telah mengevaluasi algoritma-algoritma ini secara terpisah, terdapat beberapa *gap* penelitian yang signifikan. Pertama, perbandingan langsung antara Random Forest dan XGBoost dalam konteks deteksi *phishing* masih terbatas, terutama yang menggunakan dataset dan metodologi evaluasi yang konsisten. Kedua, sebagian besar studi fokus hanya pada aspek akurasi, namun mengabaikan faktor-faktor penting lainnya seperti interpretabilitas model dan efisiensi komputasi yang krusial untuk implementasi praktis. Ketiga, analisis mendalam tentang feature importance dan kontribusi relatif setiap fitur dalam kedua algoritma belum banyak dieksplorasi secara komparatif.

Analisis *feature importance* menjadi aspek kritis dalam pengembangan sistem deteksi *phishing* yang praktis karena memberikan wawasan tentang karakteristik website yang paling mempunyai indikasi terhadap aktivitas *phishing* (Marchal et al., 2014)Penelitian terdahulu mengidentifikasi bahwa fitur *URL structure*, *domain characteristics*, dan *HTML content* merupakan indikator paling *reliable* untuk deteksi *phishing* (Somesha et al., 2020). Aspek efisiensi komputasi juga penting untuk implementasi dunia nyata, terutama untuk aplikasi yang memerlukan *response time* cepat seperti *web browser extensions* (Wu et al., 2023)

Dalam konteks Indonesia, pengembangan sistem deteksi phishing yang efektif menjadi prioritas strategis untuk mendukung program digitalisasi nasional dan melindungi masyarakat dari ancaman kemanan siber yang meningkat. Penelitian ini bertujuan memberikan analisis komparatif antara algoritma Random Forest dan XGBoost untuk deteksi website *phishing* otomatis dengan fokus pada aspek performa, interpretabilitas, dan *feasibility* implementasi. Hasil penelitian diharapkan dapat memberikan kontribusi untuk pengembangan sistem deteksi *phishing* dan mendukung penguatan infrastruktur keamanan siber Indonesia.

Metodologi Penelitian

Penelitian ini menggunakan pendekatan eksperimental komparatif untuk membandingkan performa algoritma *Random Forest* dan XGBoost dalam deteksi website *phishing* sebagaimana diilustrasikan pada Gambar 1. Metodologi dirancang untuk memastikan komparasi dengan pendekatan sistematis dan memiliki *practical relevance* untuk implementasi aplikasi keamanan siber.



Gambar 1. Metodologi Penelitian

Pengumpulan Data

Dataset yang digunakan adalah *UCI Machine Learning Repository Phishing Websites Dataset* (Abdeljaber et al., 2014) yang merupakan *benchmark* standar untuk penelitian deteksi *phishing*. Dataset berisi 11.055 instance website yang dikarakterisasi oleh 30 fitur berbeda, mencakup karakteristik berbasis URL, domain, dan HTML. Distribusi dataset terdiri dari 6.157

website phishing (55,7%) dan 4.898 website legitimate (44,3%), memberikan representasi yang seimbang untuk pelatihan model machine learning.

Pemilihan dataset ini didasarkan pada beberapa pertimbangan strategis. Pertama, dataset telah menjadi *benchmark* standar dalam penelitian deteksi *phishing* dan digunakan secara luas dalam literatur ilmiah yang memungkinkan komparasi hasil dengan penelitian sebelumnya. Kedua, kualitas anotasi data telah diverifikasi melalui proses validasi yang ketat, dengan fitur-fitur yang diekstrak berdasarkan karakteristik teknis website yang telah terbukti diskriminatif untuk membedakan *phishing* dan *legitimate websites*. Ketiga, ukuran dataset yang mencukupi (>11.000 *instance*) memungkinkan pembagian data yang optimal untuk *training, validation,* dan *testing* sambil mempertahankan representatifitas distribusi kelas.

Preprocessing Data

Tahap preprocessing dilakukan secara sistematis untuk memastikan kualitas data dan kesiapan untuk pemodelan. Proses dimulai dengan evaluasi kualitas data melalui pemeriksaan *missing values, outliers*, dan distribusi fitur. Analisis eksploratori dilakukan untuk memahami karakteristik dataset, termasuk korelasi antar fitur, distribusi kelas, dan identifikasi potensi bias. Transformasi fitur mencakup *encoding* variabel kategorikal dan normalisasi untuk memastikan semua fitur berada dalam skala yang tepat. Dataset kemudian dibagi dengan rasio 80:20 untuk *training* dan *testing* dengan *stratified sampling* untuk mempertahankan proporsi kelas yang konsisten. Semua tahap *preprocessing* dirancang untuk menjaga integritas data sambil mempersiapkan format yang optimal untuk kedua algoritma *machine learning* yang akan dibandingkan. Detail spesifik implementasi *preprocessing* akan dibahas pada bagian hasil dan pembahasan.

Implementasi Model

Implementasi dilakukan dengan pendekatan analisis komparatif yang sistematis untuk memastikan *fair comparison* antara Random Forest dan XGBoost. Penelitian ini mengatasi *gap* dalam literatur yang menunjukkan bahwa meskipun kedua algoritma telah terbukti efektif secara individual dalam domain keamanan siber, perbandingan langsung yang menggunakan dataset dan metodologi evaluasi yang konsisten masih terbatas. Sebagian besar studi sebelumnya fokus pada optimisasi satu algoritma tanpa mempertimbangkan *trade-off* antara performa, interpretabilitas, dan efisiensi komputasi yang krusial untuk implementasi praktis. Setiap algoritma diimplementasikan dengan konfigurasi *baseline* terlebih dahulu untuk *establishing performance benchmark,* kemudian dilanjutkan dengan *optimized configurations.*

Random Forest

Random Forest dipilih sebagai representative dari ensemble learning dengan pendekatan *bagging*. Algoritma ini mengimplementasikan *bootstrap aggregating* dengan random *feature selection* untuk menciptakan *ensemble classifier* yang *robust* (Ovi et al., 2024). Implementasi dilakukan menggunakan *framework* yang memungkinkan *extensive hyperparameter exploration* dengan fokus pada optimisasi performa deteksi *phishing*. Keunggulan Random Forest yang dimanfaatkan dalam penelitian ini meliputi *robustness* terhadap *overfitting*, *built-in feature importance calculation*, *native handling* untuk *missing values*, dan *parallelizable training process*.

XGBoost

XGBoost dipilih sebagai representatif dari *ensemble learning* dengan pendekatan *boosting*. Algoritma ini mengimplementasikan *gradient boosting* dengan optimisasi *advanced* dan regularisasi (Abdul Samad et al., 2023). Implementasi menggunakan konfigurasi yang memungkinkan komparasi dengan Random Forest dalam konteks deteksi *phishing*. Keunggulan XGBoost yang dimanfaatkan meliputi *sparsity-aware algorithm*, *advanced regularization techniques*, *cache-aware access patterns*, dan *built-in cross-validation capabilities*.

Evaluasi Model

Evaluasi model dilakukan menggunakan kerangka kerja komprehensif yang mencakup berbagai metrik dan pendekatan validasi untuk memastikan penilaian performa yang *robust*. Evaluasi dirancang untuk memberikan pandangan holistik tentang karakteristik performa dari kedua algoritma dalam konteks deteksi *phishing*.

Performance Metrics

Metrik utama berfokus pada penilaian kinerja keseluruhan, sedangkan metrik pendukung memberikan rincian tentang aspek-aspek spesifik dari performa model yang penting untuk pengembangan praktis. Pemilihan metrik didasarkan pada kebutuhan untuk aplikasi keamanan siber dimana *false positive* dan *false negative rates* memiliki implikasi yang berbeda. Rincian terkait perhitungan metrik, analisis statistik, dan interpretasi akan dijelaskan pada bagian hasil dan pembahasan.

Cross-Validation Strategy

Cross-validation strategy dirancang untuk memberikan estimasi performa yang *robust* dan validasi statistik. Pendekatan yang digunakan memastikan penilaian yang andal dari kinerja model sambil mempertahankan distribusi kelas yang konsisten di setiap lipatan validasi. Kerangka validasi meliputi pengujian statistik untuk menentukan signifikansi dari perbedaan performa dan penilaian stabilitas model. Hasil validasi terperinci, pengujian statistik, dan interval kepercayaan akan dianalisis pada bagian pembahasan.

Analisis Komparatif

Kerangka analisis komparatif dirancang untuk membandingkan kedua algoritma secara sistematis berdasarkan berbagai dimensi yang relevan untuk aplikasi keamanan siber. Analisis ini dilakukan untuk mengatasi *gap* penelitian terkait kurangnya evaluasi komprehensif yang mempertimbangkan tidak hanya akurasi, tetapi juga aspek interpretabilitas model dan efisiensi komputasi yang penting untuk *deployment* sistem deteksi *phishing real-time* Analisis meliputi perbandingan performa, penilaian efisiensi komputasi, dan analisis *feature importance*.

Hasil dan Pembahasan

Hasil

Pengumpulan Data

Dataset UCI Machine Learning Repository Phishing Websites digunakan sebagai dataset penelitian dengan menggunakan *tools* Google Colab. Dataset memiliki dimensi 11.055 *instance* dengan 31 kolom, termasuk 30 fitur prediktif dan 1 variabel target. Total penggunaan memori

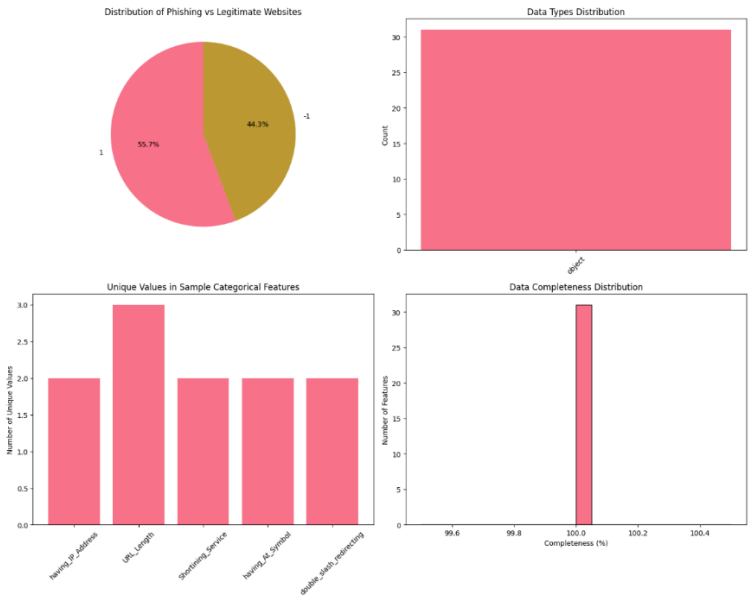
dataset adalah 16,44 MB dengan semua kolom bertipe data objek. Analisis distribusi target menunjukkan dataset memiliki distribusi yang relatif seimbang dengan 6.157 *instance website legitimate* (55,7%) dan 4.898 *instance website phishing* (44,3%). Evaluasi kualitas data mengkonfirmasi tidak adanya *missing values* pada seluruh dataset, menunjukkan kelengkapan data yang sangat baik (100% *completeness*) pada semua fitur.

Dataset mencakup 30 fitur yang terbagi dalam tiga kategori karakteristik: berbasis URL (seperti *having_IP_Address, URL_Length, Shortining_Service*), berbasis domain (seperti *Domain_registeration_length, age_of_domain, DNSRecord*), dan berbasis HTML (seperti *Iframe, popUpWidnow, Links_in_tags*). Setiap fitur memiliki representasi kategorikal dengan nilai unik berkisar antara 2-3 kategori per fitur.

***Preprocessing* Data**

Proses *preprocessing* dilaksanakan secara sistematis dengan transformasi seluruh 30 fitur kategorikal. Analisis *unique values* menunjukkan distribusi yang bervariasi: 24 fitur memiliki 2 *unique values*, dan 6 fitur memiliki 3 *unique values*. Fitur dengan kompleksitas tertinggi adalah *URL_Length, having_Sub_Domain, SSLfinal_State, URL_of_Anchor, Links_in_tags, SFH, web_traffic,* dan *Links_pointing_to_page* yang masing-masing memiliki 3 kategori. *Encoding target variable* ditransformasi dari representasi *string* ('-1', '1') menjadi format *numerik binary* (0, 1) dengan *mapping*: website *phishing* = 0 dan website *legitimate* = 1. Hasil akhir *preprocessing* menghasilkan *feature matrix* berukuran (11.055, 30) dan *target vector* berukuran (11.055). Hal ini mengkonfirmasi konsistensi dimensi data yang siap untuk tahap pemodelan *machine learning*.

Evaluasi data *completeness* menunjukkan distribusi sempurna pada tingkat 100% untuk seluruh fitur, mengindikasikan tidak diperlukan perlakuan khusus untuk *missing values* atau *outlier*. Struktur data hasil *preprocessing* telah optimal untuk implementasi algoritma Random Forest dan XGBoost dengan format *numerical* yang konsisten. Untuk memahami karakteristik dataset secara mendalam, dilakukan analisis eksploratori menggunakan visualisasi statistik. Gambar 2. menunjukkan empat aspek penting dari dataset yang telah di-*preprocessing*.



Gambar 2. Visualisasi EDA (*Exploratory Data Analysis*)

Distribusi kelas target (kiri atas) mengkonfirmasi proporsi yang relatif seimbang antara website *legitimate* (55,7%) dan website *phishing* (44,3%). Hal ini menunjukkan dataset tidak mengalami *severe class imbalance* yang dapat mempengaruhi performa model. Distribusi tipe data (kanan atas) memvalidasi bahwa seluruh 30 fitur telah berhasil ditransformasi ke format numerik yang diperlukan untuk pemodelan *machine learning*.

Analisis unique values pada fitur kategorikal (kiri bawah) mengungkapkan bahwa sebagian besar fitur memiliki 2 *unique values*, kecuali fitur URL_Length yang memiliki 3 *unique values*. Hal tersebut menunjukkan bahwa kompleksitas representasi yang moderat. Distribusi kelengkapan data (kanan bawah) mengkonfirmasi data *completeness* 100% pada seluruh fitur yang mengindikasikan kualitas dataset yang sudah baik. Hasil analisis eksploratori ini menjadi dasar validasi bahwa dataset telah siap untuk tahap implementasi model dengan kualitas data yang optimal dan distribusi yang mendukung *supervised learning* yang efektif.

Implementasi Model

Implementasi model dilakukan dengan membagi data *training-testing* sebesar 80:20. Dataset berhasil dibagi menjadi *training set* sebanyak 8.844 sampel dan *test set* sebanyak 2.211 sampel. Distribusi kelas pada *training set* menunjukkan proporsi yang seimbang dengan 4.926 *instance legitimate websites* dan 3.918 *instance phishing websites*.

Random Forest

Model Random Forest *baseline* diimplementasikan dengan performa yang baik pada tahap *initial training*. Model menunjukkan *training time* yang efisien selama 1,38 detik dengan memanfaatkan *bootstrap aggregating* dan *random feature selection*. Hasil evaluasi *baseline* menunjukkan akurasi sebesar 97,42% dan AUC score mencapai 99,77%. Hal ini mengindikasikan kemampuan *discriminative* yang sangat tinggi dalam membedakan website *phishing* dan *legitimate*.

Optimisasi *hyperparameter* dilakukan menggunakan *RandomizedSearchCV* dengan *5-fold stratified cross-validation* untuk memastikan *robustness* hasil. Parameter *grid* mencakup *n_estimators [100, 200, 300], max_depth [10, 15, 20, None], min_samples_split [2, 5, 10], min_samples_leaf [1, 2, 4], max_features ['sqrt', 'log2'],* dan *class_weight ['balanced', None]*. Proses *grid search* mengeksplorasi 50 kombinasi parameter dengan total 250 *fits* selama 235,87 detik.

Hasil optimisasi mengidentifikasi konfigurasi optimal: *n_estimators=200, max_depth=20, min_samples_split=2, min_samples_leaf=1, max_features='log2',* dan *class_weight=None*. Model teroptimasi mencapai *best CV score (AUC)* sebesar 99,60%, menunjukkan sedikit menurun dibandingkan *baseline* namun dengan *improved generalization capability* melalui *hyperparameter tuning*.

XGBoost

Model XGBoost *baseline* diimplementasikan dengan performa yang kompetitif dan efisiensi *computational* yang lebih baik. Model menunjukkan *training time* yang lebih cepat selama 0,84 detik dengan memanfaatkan *gradient boosting optimization*. Hasil evaluasi *baseline* menunjukkan akurasi sebesar 97,38% dan *AUC score* mencapai 99,75%, mendemonstrasikan

kemampuan prediktif yang sangat tinggi dengan sedikit perbedaan dibandingkan dengan Random Forest.

Optimisasi *hyperparameter* XGBoost menggunakan parameter *grid* yang komprehensif meliputi *n_estimators [100, 200, 300], max_depth [3, 6, 9], learning_rate [0.01, 0.1, 0.2], subsample [0.8, 0.9, 1.0], colsample_bytree [0.8, 0.9, 1.0], min_child_weight [1, 3, 5], reg_alpha [0, 0.1, 0.5],* dan *reg_lambda [1, 1.5, 2]*. Proses *grid search* mengeksplorasi 50 kombinasi parameter dengan total 250 *fits* selama 63,93 detik. Hal ini menunjukkan efisiensi *computational* yang lebih baik dibandingkan Random Forest.

Hasil optimisasi mengidentifikasi konfigurasi *optimal: n_estimators=200, max_depth=9, learning_rate=0.1, subsample=0.8, colsample_bytree=0.9, min_child_weight=1, reg_alpha=0.1,* dan *reg_lambda=2*. Model teroptimasi mencapai *best CV score (AUC)* sebesar 99,61% yang menunjukkan peningkatan dibandingkan *baseline* dan sedikit lebih baik dibandingkan Random Forest yang telah dioptimasi.

Evaluasi Model

Evaluasi model dilakukan untuk memberikan penilaian kinerja yang kuat pada kedua algoritma yang telah dioptimasi. Evaluasi menggunakan data uji yang sepenuhnya tidak terlihat selama proses pelatihan dan optimasi *hiperparameter* untuk memastikan penilaian yang tidak bias dari kemampuan generalisasi model.

Performance Metrics

Model Random Forest teroptimasi menunjukkan kinerja yang sangat baik pada evaluasi data uji dengan akurasi mencapai 97,65%. Model mencapai skor AUC-ROC sebesar 99,79% yang mengindikasikan kemampuan diskriminatif yang sangat tinggi dalam membedakan website *phishing* dan *legitimate*. Skor rata-rata presisi mencapai 99,83% yang menunjukkan keseimbangan *precision-recall* yang optimal untuk aplikasi keamanan siber.

Hasil evaluasi mengkonfirmasi bahwa optimasi *hyperparameter* berhasil meningkatkan kinerja dibandingkan *baseline* dengan peningkatan pada akurasi dari 97,42% menjadi 97,65%. Konsistensi antara skor validasi silang (99,60%) dan AUC data uji (99,79%) menunjukkan tidak adanya *overfitting* dan kinerja model yang kuat pada distribusi data yang berbeda.

Model XGBoost teroptimasi mendemonstrasikan kinerja yang superior dengan akurasi mencapai 97,78%. Model mencapai skor AUC-ROC sebesar 99,76% yang menunjukkan kemampuan prediktif yang sangat baik dan sedikit lebih unggul dari Random Forest. Skor rata-rata presisi mencapai 99,80% yang mengindikasikan kinerja yang seimbang antara presisi dan *recall* yang kritis untuk aplikasi deteksi *phishing*.

Evaluasi menunjukkan peningkatan dibandingkan *baseline* dengan peningkatan akurasi dari 97,38% menjadi 97,78%. Kemampuan generalisasi yang sangat baik terbukti dengan celah minimal antara skor validasi silang (99,61%) dan kinerja data uji (99,76%). Hal ini mengkonfirmasi efektivitas teknik regularisasi dalam XGBoost untuk mencegah *overfitting*.

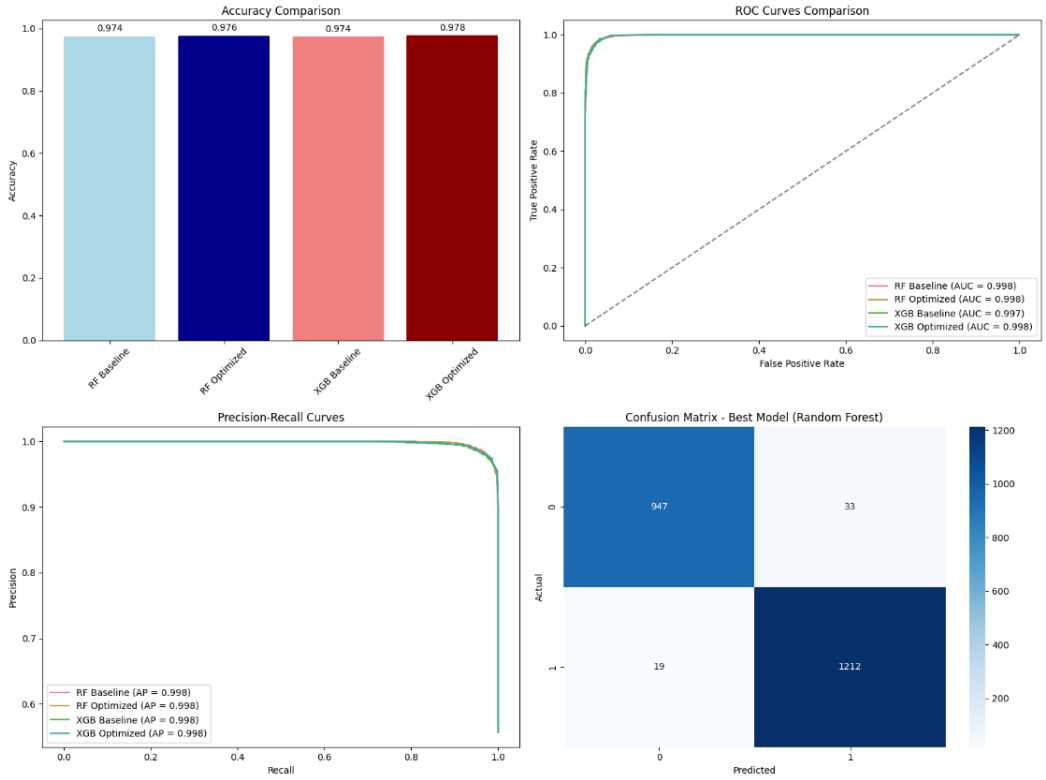
Hasil akurasi yang sangat tinggi (di atas 97%) ini wajar mengingat dataset UCI yang digunakan sudah sangat berkualitas dan bersih dari *noise*. Perbedaan yang kecil antara model dasar dan model teroptimasi (hanya 0,23% untuk Random Forest dan 0,40% untuk XGBoost)

menunjukkan bahwa pengaturan *default* dari kedua algoritma sudah sangat efektif untuk dataset yang berkualitas tinggi seperti ini.

Temuan ini memberikan panduan praktis yang berharga untuk implementasi di dunia nyata dengan dataset serupa, praktisi dapat memulai dengan pengaturan *default* terlebih dahulu. Proses optimasi yang membutuhkan waktu 171 kali lebih lama mungkin tidak selalu diperlukan jika peningkatan performa yang kecil tidak sebanding dengan biaya komputasi yang dikeluarkan.

Analisis mendalam menggunakan kurva ROC dan kurva *Precision-Recall* menunjukkan kinerja yang luar biasa pada kedua model teroptimasi. Gambar 3. menunjukkan perbandingan komprehensif meliputi perbandingan akurasi, kurva ROC, kurva *precision-recall*, dan analisis *confusion matrix*.

Analisis kurva ROC mengkonfirmasi skor AUC yang luar biasa Random Forest teroptimasi (99,8%), XGBoost teroptimasi (99,8%), dengan model *baseline* juga menunjukkan kinerja yang sangat baik di atas 99,7%. Kurva *precision-recall* menunjukkan skor rata-rata presisi yang konsisten di 99,8% untuk semua model, mengindikasikan kinerja yang kuat di berbagai ambang keputusan.



Gambar 3. Visualisasi Komparasi Performa

Analisis *confusion matrix* pada model terbaik (Random Forest) menunjukkan distribusi prediksi yang optimal dengan 1212 *true positives*, 947 *true negatives*, 19 *false negatives*, dan 33 *false positives*. Hasil ini menghasilkan tingkat *false positive* yang sangat rendah (3,4%) dan tingkat *false negative* yang minimal (1,5%), yang kritis untuk aplikasi keamanan siber.

Cross-Validation

Analisis *cross-validation* menggunakan pendekatan *5-fold stratified* yang mengkonfirmasi ketahanan dan kemampuan generalisasi kedua model. Tabel 1. menunjukkan kinerja statistik detail dari hasil *cross-validation*.

Tabel 1. Mean dan Standar Deviasi Metrik Evaluasi dari *Cross-Validation*

Metrik	RF_Mean	RF_Std	XGB_Mean	XGB_Std
Accuracy	0.9697	0.0024	0.9694	0.0024
ROC-AUC	0.9960	0.0005	0.9961	0.0005
Precision	0.9662	0.0031	0.9701	0.0020
Recall	0.9799	0.0043	0.9750	0.0037
F1-Score	0.9730	0.0022	0.9726	0.0021

Standar deviasi yang sangat rendah (< 0.005) pada semua metrik mengkonfirmasi stabilitas model dan kinerja yang konsisten di berbagai fold data. XGBoost menunjukkan sedikit keunggulan dalam presisi (97,01% vs 96,62%) dan konsistensi ROC-AUC, sementara Random Forest unggul dalam kinerja *recall* (97,99% vs 97,50%).

Gambar 4. Hasil Perbandingan Metrik Evaluasi Model Menggunakan *Cross-Validation*

Pembahasan

Perbandingan Model *Baseline* vs Teroptimasi

Analisis komparatif menunjukkan peningkatan yang melalui optimasi *hyperparameter* pada kedua algoritma. Random Forest mengalami peningkatan akurasi dari 97,42% (*baseline*) menjadi 97,65% (teroptimasi), dengan peningkatan AUC dari 99,77% menjadi 99,79%. XGBoost menunjukkan peningkatan yang lebih substansial dengan akurasi meningkat dari 97,38% menjadi 97,78%, dan AUC dari 99,75% menjadi 99,76%.

Perbandingan waktu pelatihan mengungkapkan *trade-off* antara peningkatan kinerja dan biaya komputasi. Optimasi Random Forest membutuhkan 235,87 detik (171x lebih lama dari *baseline*), sementara optimasi XGBoost membutuhkan 63,93 detik (76x lebih lama dari *baseline*).

XGBoost menunjukkan efisiensi komputasi yang lebih baik dengan waktu optimasi 73% lebih cepat dibandingkan Random Forest.

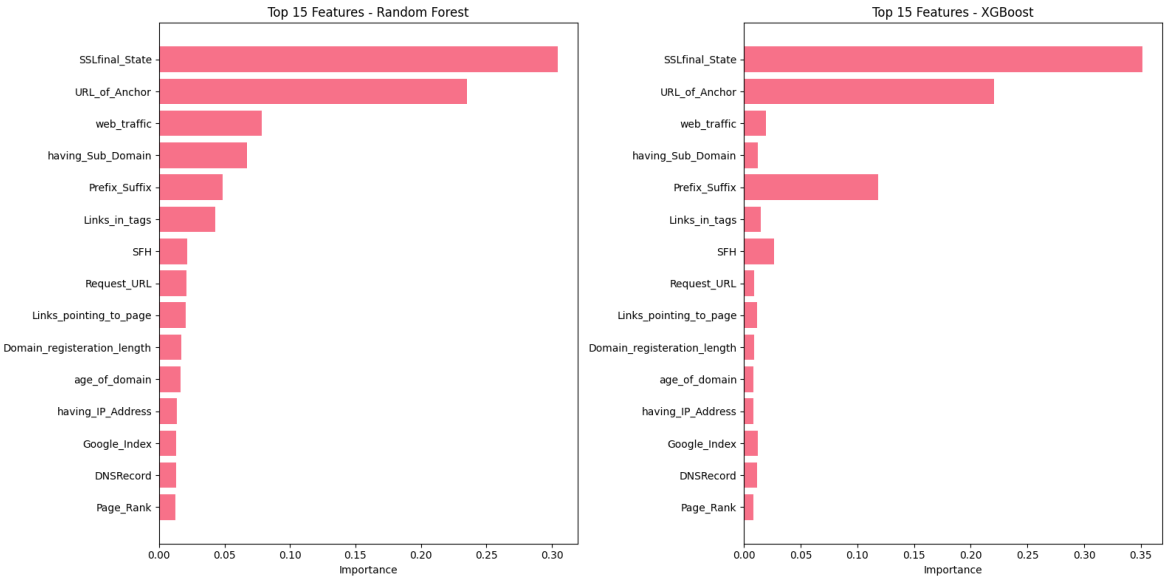
Perbandingan Kinerja *Random Forest* vs XGBoost

Perbandingan komprehensif antara model Random Forest dan XGBoost teroptimasi menunjukkan kinerja yang sangat kompetitif. XGBoost menunjukkan sedikit keunggulan dalam akurasi (97,78% vs 97,65%), sementara Random Forest unggul dalam AUC-ROC (99,79% vs 99,76%) dan rata-rata presisi (99,83% vs 99,80%). Perbedaan kinerja berada dalam range yang minimal namun konsisten di berbagai metrik.

Analisis *cross-validation* mengkonfirmasi bahwa perbedaan kinerja antara kedua algoritma berada dalam *range* signifikansi statistik yang dapat diterima. Standar deviasi yang sangat rendah (< 0.005) pada metrik ROC-AUC menunjukkan konsistensi kinerja yang kuat dengan XGBoost menunjukkan kinerja rata-rata yang sedikit superior (99,61% vs 99,60%) namun dengan varians yang sebanding.

Analisis Komparatif *Feature Importance*

Analisis *feature importance* mengungkapkan konsistensi yang luar biasa antara kedua algoritma dalam mengidentifikasi indikator *phishing* yang kritis. Gambar 5. menunjukkan analisis komparatif dari 15 fitur teratas untuk kedua model.



Gambar 5. Perbandingan *Feature Importance* dari 15 Fitur Teratas

Kedua algoritma mengidentifikasi *SSLfinal_State* sebagai fitur yang paling diskriminatif dengan kontribusi 30,42% (Random Forest) dan 35,15% (XGBoost). *URL_of_Anchor* konsisten menjadi fitur kedua terpenting dengan kontribusi 23,54% (Random Forest) dan 22,06% (XGBoost). Kontribusi gabungan dari kedua fitur utama mencapai 53,96% (Random Forest) dan 57,21% (XGBoost).

Tabel 2. Tingkat Kepentingan Fitur pada Model Random Forest dan XGBoost

Fitur	***RF_Importance***	***XGB_Importance***
SSLfinal_State	0.3042	0.3515
URL_of_Anchor	0.2354	0.2206
web_traffic	0.0783	0.0192
having_Sub_Domain	0.0672	0.0121
Prefix_Suffix	0.0485	0.1180
Links_in_tags	0.0431	0.0147
SFH	0.0215	0.0263
Request_URL	0.0212	0.0091
Links_pointing_to_page	0.0203	0.0115
Domain_registeration_length	0.0171	0.0092

Dominasi fitur *SSLfinal_State* (30-35%) dan *URL_of_Anchor* (22-23%) sebagai indikator utama memberikan wawasan strategis yang penting bagi praktisi keamanan siber. Hal ini mengkonfirmasi bahwa meskipun teknik *phishing* terus berkembang, para pelaku masih mengandalkan metode penipuan fundamental pada tingkat infrastruktur website.

Konsistensi peringkat kedua fitur teratas pada kedua algoritma menunjukkan keandalan temuan ini dan memberikan panduan yang dapat dipercaya untuk pemilihan fitur dalam implementasi sistem nyata. Perbedaan pada fitur sekunder mencerminkan cara kerja masing-masing algoritma - Random Forest lebih peka terhadap pola statistik sementara XGBoost lebih fokus pada pola struktural, memberikan perspektif yang saling melengkapi untuk deteksi ancaman yang menyeluruh.

Perbedaan terletak pada peringkat fitur sekunder, dimana Random Forest memberikan *importance* yang lebih tinggi pada *web_traffic* (7,83% vs 1,92%) dan *having_Sub_Domain* (6,72% vs 1,21%), sementara XGBoost lebih menekankan *Prefix_Suffix* (11,80% vs 4,85%). Konsistensi pada 2 fitur teratas mengindikasikan ketahanan dalam mengidentifikasi indikator phishing primer.

Dalam konteks penerapan untuk keamanan siber Indonesia, hasil penelitian ini memberikan panduan yang dapat langsung diterapkan. Tingkat *false positive* 1,5% yang dicapai menunjukkan keseimbangan optimal antara keamanan dan kenyamanan pengguna - cukup rendah untuk menghindari terlalu banyak peringatan palsu namun tetap memberikan perlindungan yang menyeluruh.

Tabel 3. Perbandingan Performa Model Sebelum dan Sesudah Optimasi

Model	*Accuracy*	AUC-ROC	*Training_Time*(s)
RF *Baseline*	0.9742	0.9977	1.38
RF *Optimized*	0.9765	0.9979	235.87
XGB *Baseline*	0.9738	0.9975	0.84
XGB *Optimized*	0.9778	0.9976	63.93

Dari hasil yang ditunjukan oleh Tabel 3. Terdapat perbedaan waktu pelatihan yang antara Random Forest (235,87 detik) dan XGBoost (63,93 detik) untuk proses optimasi menjadi pertimbangan penting dalam siklus pengembangan yang cepat. XGBoost memberikan kemampuan iterasi yang lebih cepat untuk penyempurnaan model yang dapat berguna dalam lingkungan pengembangan yang dinamis.

Berdasarkan evaluasi komprehensif, model XGBoost yang telah dioptimasi menunjukkan performa lebih baik dengan akurasi tertinggi (97.78%), waktu training 73% lebih cepat (63.93s vs 235.87s), dan efisiensi komputasi yang lebih baik untuk retraining berkala dan menjadikannya pilihan optimal untuk *deployment production* sistem deteksi phishing *real-time*. Random Forest yang telah dioptimasi memberikan performa kompetitif (97.65% akurasi, 99.79% AUC-ROC) dengan keunggulan interpretabilitas yang lebih baik, cocok untuk skenario yang memprioritaskan *explainability* model untuk audit.

Kesimpulan

Penelitian ini berhasil melakukan analisis komparatif yang komprehensif antara algoritma Random Forest dan XGBoost untuk deteksi website phishing otomatis menggunakan teknik *machine learning*. Berdasarkan hasil eksperimen yang telah dilakukan pada dataset UCI *Machine Learning Repository* dengan 11.055 sampel website, dapat disimpulkan beberapa temuan penting.

Kedua algoritma menunjukkan kemampuan deteksi phishing yang sangat tinggi setelah optimisasi hyperparameter. XGBoost teroptimasi mencapai akurasi tertinggi 97,78% dengan AUC-ROC 99,76%, sementara Random Forest teroptimasi mencapai 97,65% akurasi dengan AUC-ROC 99,79%. XGBoost menunjukkan keunggulan dalam efisiensi komputasi dengan waktu optimasi 73% lebih cepat (63,93 detik vs 235,87 detik). Peningkatan minimal dari optimisasi mengindikasikan bahwa pengaturan default kedua algoritma sudah sangat efektif untuk dataset berkualitas tinggi.

Analisis feature importance mengungkapkan konsistensi antara kedua algoritma dalam mengidentifikasi SSLfinal_State dan URL_of_Anchor sebagai fitur paling diskriminatif dengan kontribusi gabungan lebih dari 53%. Lebih lanjut, Cross-validation mengkonfirmasi stabilitas kedua model dengan standar deviasi sangat rendah (<0,005) pada semua metrik performa. Hal ini memberikan kepercayaan tinggi terhadap reliabilitas model dalam implementasi produksi.

Berdasarkan evaluasi komprehensif, XGBoost teroptimasi menjadi pilihan optimal untuk deployment sistem deteksi phishing *real-time* karena akurasi tertinggi, efisiensi komputasi lebih superior, dan kemampuan *retraining* yang lebih cepat. Sementara itu, Random Forest lebih sesuai

untuk skenario yang memerlukan interpretabilitas model dan transparansi dalam proses audit keamanan.

Penelitian ini memberikan kontribusi untuk pengembangan sistem keamanan siber nasional melalui validasi efektivitas *ensemble learning methods*, identifikasi fitur-fitur kritis yang dapat diandalkan, dan penyediaan kerangka kerja evaluasi yang dapat diterapkan langsung untuk penguatan infrastruktur keamanan siber Indonesia. Keterbatasan penelitian meliputi penggunaan dataset snapshot periode tertentu dan evaluasi dalam lingkungan laboratorium.

Penelitian lanjutan dapat difokuskan pada pengembangan sistem *hybrid* XGBoost-*Deep Learning* untuk analisis multimodal, implementasi *production real-time* dengan *continual learning*, dan eksplorasi arsitektur *state-of-the-art* untuk meningkatkan kemampuan deteksi ancaman phishing yang semakin canggih.

Daftar Pustaka

Abdeljaber, F., Mohammad, R. M., Thabtah, F., & McCluskey, L. (2014). Predicting phishing websites based on self-structuring neural network. *Neural Computing and Applications, 25*(2).

Abdul Samad, S. R., Balasubramanian, S., Al-Kaabi, A. S., Sharma, B., Chowdhury, S., Mehbodniya, A., Webber, J. L., & Bostani, A. (2023). Analysis of the performance impact of fine-tuned machine learning model for phishing URL detection. *Electronics, 12*(7). https://doi.org/10.3390/electronics12071642

Amora, E. N. O., Agoylo, J. C., Olaybar, J. A., Munasque, J. C., & Cerna, P. D. (2025). AI-driven *real-time* severity prediction for cyber attacks using machine learning. In *Proceedings of the 3rd International Conference on Self Sustainable Artificial Intelligence Systems (ICSSAS 2025)* (pp. 1467–1472). https://doi.org/10.1109/ICSSAS66150.2025.11081230

Badan Siber dan Sandi Negara. (2024). *Lanskap keamanan siber Indonesia 2023*. https://www.bssn.go.id/

Dutta, A. K. (2021). Detecting phishing websites using machine learning technique. *PLOS ONE, 16*(10), e0258361. https://doi.org/10.1371/journal.pone.0258361

Fang, X., & Liu, M. (2024). How does the digital transformation drive digital technology innovation of enterprises? Evidence from enterprise digital patents. *Technological Forecasting and Social Change, 204*, 123428. https://doi.org/10.1016/j.techfore.2024.123428

Federal Bureau of Investigation. (2023). *2022 Internet crime report*. https://www.ic3.gov/Media/PDF/AnnualReport/2022_IC3Report.pdf

Kementerian Komunikasi dan Digital Republik Indonesia. (2023). *Memenuhi layanan digital hingga pelosok*. https://www.komdigi.go.id/berita/artikel/detail/memenuhi-layanan-digital-hingga-pelosok

Marchal, S., François, J., State, R., & Engel, T. (n.d.). *PhishStorm: Detecting phishing with streaming analytics*. http://www.quadrodeofertas.com.br/www1.paypal-com/encripted/ssl218

Ovi, S., Rahman, M., & Hossain, M. (2024). PhishGuard: A multi-layered ensemble model for optimal phishing website detection. *Proceedings of the International Conference on Smart Technologies and Innovations*. https://doi.org/10.1109/STI64222.2024.10951075

Rajalim, S. (2025). Understanding human vulnerabilities: A study on social engineering techniques in cybersecurity. *International Journal for Research in Applied Science and Engineering Technology*, *13*(6), 1660–1666. https://doi.org/10.22214/ijraset.2025.72502

Safi, A., & Singh, S. (2023). A systematic literature review on phishing website detection techniques. *Journal of King Saud University – Computer and Information Sciences, 35*(2), 590–611. https://doi.org/10.1016/j.jksuci.2023.01.004

Sahingoz, O. K., Buber, E., Demir, O., & Diri, B. (2019). Machine learning based phishing detection from URLs. *Expert Systems with Applications, 117,* 345–357. https://doi.org/10.1016/j.eswa.2018.09.029

Wu, C.-Y., Kuo, C.-C., & Yang, C.-S. (2023). Phishing detection with browser extension based on machine learning. *Proceedings of the Asia Joint Conference on Information Security*. https://doi.org/10.1109/AsiaJCIS60284.2023.00023