JRIIN : Jurnal Riset Informatika dan Inovasi
Volume 4, No. 4 Tahun 2026
ISSN 3025-0919 (media online)
Hal 1020-1029

Deteksi Dini Website Phising Berbasis Karakteristik URL Menggunakan Algoritma Random Forest sebagai Upaya Penegakan Keamanan Siber

Muhammad Faqih Alharits[1*], Hanif Maulana Ar Rasyid[2], Rifky Firmansyah[3], Abdullah Rendra Zuriansyah[4], Ardiansyah Maulana[5], Firza Aditiya Ardiansah[6], Rahmawati [7]

[1,2,3,4,5,6,7]Fakultas Ilmu Komputer, Teknik Informatika, Universitas Pamulang, Kota Tangerang Selatan, Indonesia
Email: [1*]alharitsfaqih@gmail.com, [2]hanifmaulanaarrasyid@gmail.com, [3]rfky.frmansyah@gmail.com, [4]rendra.zuriansyah.edu@gmail.com, [5]ardimaulana1232@gmail.com, [6]firzaaditiya22@gmail.com, [7]dosen02394@unpam.ac.id
(* : coressponding author)

Abstrak−*Phising* merupakan salah satu ancaman keamanan siber yang paling merugikan, dengan modus menyamar sebagai *website* terpercaya untuk mencuri data sensitif pengguna. Penleitian ini mengembangkan sistem deteksi dini *phising website* berbasis karakteristik URL menggunakan algoritma *Random Forest*. *Dataset* yang digunakan terdiri dari 10.000 sampel URL (50% *legitimate*, *50% phising*) dengan 48 fitur berbasis karakteristik URL. Model *baseline Random Forest* mencapai akurasi 98,55%, presisi 98,60%, *recall* 98,50%, F1-*score* 98,55%, dan AUC-ROC 99,09%. *Hyperparameter tuning* menggunakan *RandomizedSearchCV* menghasilkan model dengan performa serupa namun sedikit lebih rendah pada *recall* (98,20%). Analisis *feature importance* menunjukkan bahwa PctExtHyperlinks (20,69%), PctExtNullSelfRedirectHyperlinksRT (16,77%), dan FrequentDomainNameMismatch (7,78%) merupakan fitur paling informatif. Kajian etika profesi dilakukan terhadap implikasi false positive dan false negative, tanggung jawab profesional pengembang, serta kepatuhan terhadap regulasi Indonesia (UU ITE, UU PDP No. 27 Tahun 2022). Penelitian ini menunjukkan bahwa pendekatan machine learning berbasis Random Forest efektif untuk deteksi phishing dan dapat berkontribusi pada penegakan keamanan siber di Indonesia.

Kata Kunci: Phising; Random Forest; Keamanan Siber; Deteksi URL; Etika Profesi

Abstract− *Phishing is one of the most damaging cybersecurity threats, disguising malicious websites as trusted ones to steal sensitive user data. This study develops an early detection system for phishing websites based on URL characteristics using the Random Forest algorithm. The dataset consists of 10,000 URL samples (50% legitimate, 50% phishing) with 48 URL-based features. The baseline Random Forest model achieved 98.55% accuracy, 98.60% precision, 98.50% recall, 98.55% F1-score, and 99.90% AUC-ROC. Hyperparameter tuning using RandomizedSearchCV produced a model with similar but slightly lower recall performance (98.20%). Feature importance analysis revealed that PctExtHyperlinks (20.69%), PctExtNullSelfRedirectHyperlinksRT (16.77%), and FrequentDomainNameMismatch (7.78%) are the most informative features. A professional ethics review was conducted on false positive and false negative implications, developer responsibilities, and compliance with Indonesian regulations (UU ITE, UU PDP No. 27/2022). This study demonstrates that Random Forest-based machine learning is effective for phishing detection and can contribute to cybersecurity enforcement in Indonesia.*

Keywords: *Phising; Random Forest; Cybersecurity; URL Detection; Professional Ethics*

1. PENDAHULUAN

Perkembangan teknologi informasi dan internet yang pesat telah membawa manfaat besar bagi masyarakat, namun di sisi lain juga membuka peluang bagi kejahata siber. Salah satu bentuk kejahatan siber yang paling merugikan adalah *phising*, yaitu upaya penipuan untuk memperoleh informasi sensitif seperti *username*, *password*, dan data kartu kredit dengan menyamar sebagai entitas terpercaya melalui komunikasi elektronik (Jain & Gupta, 2019). Penelitian terbaru menunjukkan bahwa serangan *phising* terus meningkat secara signifikan dengan memanfaatkan teknik manipulasi URL yang semakin canggih (Karim et al., 2023).

Di Indonesia, ancaman *phising* semakin mengkhawatirkan seiring dengan meningkatnya penetrasi internet. Berbagai studi menunjukkan bahwa *phising* menjadi salah satu serangan siber tertinggi di Indonesia, dengan implikasi hukum yang serius terkait perlindungan data pribadi dan keamanan siber (Sumartono et al., 2024). Kerugian yang ditimbulkan tidak hanya bersifat finansial, tetapi juga mencakup pencurian identitas dan pelanggaran privasi data pribadi.

JRIIN : Jurnal Riset Informatika dan Inovasi
Volume 4, No. 4 Tahun 2026
ISSN 3025-0919 (media online)
Hal 1020-1029

Berbagai pendekatan telah dikembangkan untuk mendeteksi website *phising*, mulai dari *blacklisting*, heuristik, hingga pendekatan berbasis *machine learning* (Sahingoz et al., 2019). Pendekatan *machine learning* menawarkan keunggulan dalam hal kemampuan generalisasi terhadap pola-pola *phising* baru yang belum pernah teridentifikasi sebelumnya. Beberapa penelitian terbaru juga mengkombinasikan *deep learning* dengan *ensemble learning* untuk meningkatkan akurasi deteksi (Yang et al., 2021). Di antara berbagai algoritma *machine learning*, *Random Forest* dikenal memiliki performa yang sangat baik untuk tugas klasifikasi dengan dimensi fitur tinggi dan mampu menangani *overfitting* secara efektif (Breiman, 2001).

Penelitian ini bertujuan untuk mengembangkan sistem deteksi dini website *phising* berbasis karakteristik URL menggunakan algoritma *Random Forest*. Karakteristik URL dipilih karena dapat diekstraksi tanpa perlu mengakses konten halaman web, sehingga lebih efisien dan aman (Rao & Pais, 2018). Selain aspek teknis, penelitian ini juga mengkaji dimensi etika profesi dalam pengembangan sistem deteksi *phising*, mencakup tanggung jawab profesional pengembang, implikasi kesalahan klasifikasi, serta kepatuhan terhadap regulasi yang berlaku di Indonesia.

2. METODE

2.1 Dataset

Dataset yang digunakan berasal dari Kaggle dengan nama "*Phising Dataset for Machine Learning*" (Shashwat, 2020) berisi 10.000 sampel URL yang terbagi rata, yaitu 5.000 URL *legitimate* (label 0) dan 5.000 URL *phising* (label 1). Distribusi yang seimbang ini mencegah model bias terhadap salah satu kelas.

Setiap URL direpresentasikan oleh 48 fitur numerik yang diekstrak tanpa mengakses konten halaman yang mencakup panjang URL dan komponennya (UrlLength, HostLength, PathLength), jumlah karakter khusus (NumDash, NumDots, NumUnderscore), informasi domain (NumSubDomain, FrequentDomainNameMismatch), persentase *hyperlink* eksternal (PctExtHyperlinks, PctExtResourceUrls), serta indikator keamanan (InsecureForms, IframeOrFrame).

2.2 *Preprocessing* dan Pembagian Data

Sebelum diproses, data terlebih dahulu diperiksa kualitasnya. Hasilnya menunjukkan tidak ada data yang kosong (*missing values*) pada seluruh 48 fitur, sehingga tidak diperlukan penanganan khusus seperti pengisian atau penghapusan data.

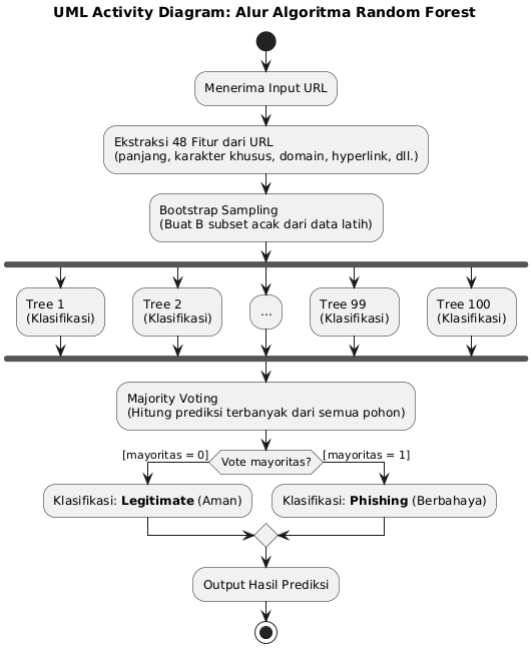
Selanjutnya, *dataset* dibagi menjadi dua bagian, yakni 80% untuk data latih (8.000 sampel) dan 20% untuk data uji (2.000 sampel). Pembagian menggunakan teknik *stratified splitting* yang menjamin proporsi kelas *phising* dan *legitimate* tetap seimbang di kedua bagian. Data latih digunakan untuk melatih model, sedangkan data uji digunakan untuk menguji performa model pada data yang belum pernah dilihat sebelumnya.

2.3 Algoritma Random Forest

Random Forest adalah algoritma *machine learning* yang bekerja dengan cara membuat banyak pohon keputusan (*decision tree*) sekaligus, kemudian menggabungkan hasil prediksi dari semua pohon tersebut melalui sistem voting mayoritas (Breiman, 2001). Konsepnya sederhana, yakni jika satu pohon keputusan bisa salah, maka dengan membuat ratusan pohon dan mengambil jawaban terbanyak, hasilnya akan lebih akurat. Algoritma ini dipilih karena cocok untuk *dataset* dengan banyak fitur (48 fitur), tidak mudah *overfitting*, dan tidak memerlukan normalisasi data terlebih dahulu.

Secara ringkas, alur kerja algoritma *Random Forest* dalam penelitian ini diilustrasikan pada Gambar 1. Dimulai dari data URL masuk, kemudian diekstrak fiturnya, lalu diproses oleh banyak pohon keputusan secara bersamaan, dan hasil akhirnya ditentukan berdasarkan voting mayoritas.

JRIIN : Jurnal Riset Informatika dan Inovasi
Volume 4, No. 4 Tahun 2026
ISSN 3025-0919 (media online)
Hal 1020-1029



Gambar 1. Diagram Alur Algoritma *Random Forest* untuk Deteksi *Phising*

Secara formal, jika kita memiliki B buah pohon keputusan, maka prediksi akhir $\hat{y}$ untuk sebuah URL x ditentukan berdasarkan jawaban terbanyak dari seluruh pohon:

$$\hat{y} = mode\{h_1(x), h_2(x), \dots, h_{13}(x)\}$$

Artinya, setiap pohon h_b memberikan prediksinya (*phising* atau *legitimate*), lalu jawaban yang paling banyak dipilih menjadi keputusan akhir. Misalnya, jika dari 100 pohon, 87 pohon memprediksi "*phising*" dan 13 pohon memprediksi "*legitimate*", maka URL tersebut diklasifikasikan sebagai *phising*.

Setiap pohon dibangun dari sampel data yang diambil secara acak (*bootstrap sampling*), dan pada setiap percabangan (*node*), hanya sebagian fitur yang dipilih secara acak sebanyak $m = \lfloor\sqrt{p}\rfloor$ dari total p fitur. Pemilihan percabangan terbaik menggunakan kriteria *Gini Impurity* yang mengukur seberapa "campuran" komposisi kelas pada suatu *node*:

$$Gini(t) = 1 - \sum_{k=1}^{K} P_k^2$$

di mana p_k adalah proporsi sampel kelas k pada *node* t. Nilai *Gini* berkisar antara 0 (sempurna murni, hanya satu kelas) hingga 0,5 (campuran merata). Sebagai contoh sederhana, jika sebuah *node* berisi 60 URL *phising* dan 40 URL *legitimate*, maka $Gini = 1 - (0{,}6)^2 - (0{,}4)^2 = 1 - 0{,}36 - 0{,}16 = 0{,}48$. Nilai 0,48 yang mendekati 0,5 berarti komposisi masih sangat campuran, sehingga pohon akan terus mencari percabangan yang lebih baik untuk memisahkan kedua kelas.

2.4 Eksperimen

Penelitian ini dilakukan dalam dua tahap. Tahap pertama adalah membangun model *baseline*, yaitu model *Random Forest* dengan pengaturan default (100 pohon keputusan) tanpa optimasi apapun. Model ini berfungsi sebagai titik awal untuk melihat seberapa baik algoritma bekerja "apa adanya". Tahap kedua adalah *hyperparameter tuning*, yaitu proses mencari kombinasi pengaturan terbaik agar model bisa bekerja lebih optimal. Pencarian dilakukan menggunakan *RandomizedSearchCV* yang mencoba 50 kombinasi parameter secara acak, masing-masing divalidasi dengan 5-*fold cross-validation* (data latih dibagi menjadi 5 bagian, bergantian dijadikan data validasi). Parameter yang dicari beserta rentang nilainya disajikan pada Tabel 1.

JRIIN : Jurnal Riset Informatika dan Inovasi
Volume 4, No. 4 Tahun 2026
ISSN 3025-0919 (media online)
Hal 1020-1029

Tabel 1. Ruang Pencarian *Hyperparameter*

Parameter	Rentang Nilai
n_estimators	100, 200, 300, 500
max_depth	5, 10, 15, 20, None
min_samples_split	2, 3, 5, 10
min_samples_leaf	1, 2, 4
max_features	Sqrt, log2

2.5 Metrik Evaluasi

Untuk menilai seberapa baik model bekerja, digunakan beberapa metrik evaluasi. *Accuracy* adalah metrik paling dasar yang mengukur berapa persen prediksi model yang benar dari seluruh data uji:

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

di mana TP (*True Positive*) adalah URL phising yang berhasil dideteksi, TN (*True Negative*) adalah URL *legitimate* yang benar diklasifikasikan, FP (*False Positive*) adalah URL *legitimate* yang salah dianggap *phising*, dan FN (*False Negative*) adalah URL *phising* yang lolos dari deteksi.

Namun *accuracy* saja tidak cukup. *Precision* mengukur dari semua URL yang diprediksi *phising*, berapa banyak yang memang benar *phising*. *Recall* mengukur dari semua URL *phising* yang sebenarnya ada, berapa banyak yang berhasil terdeteksi. F1-*score* menggabungkan keduanya menjadi satu angka yang seimbang:

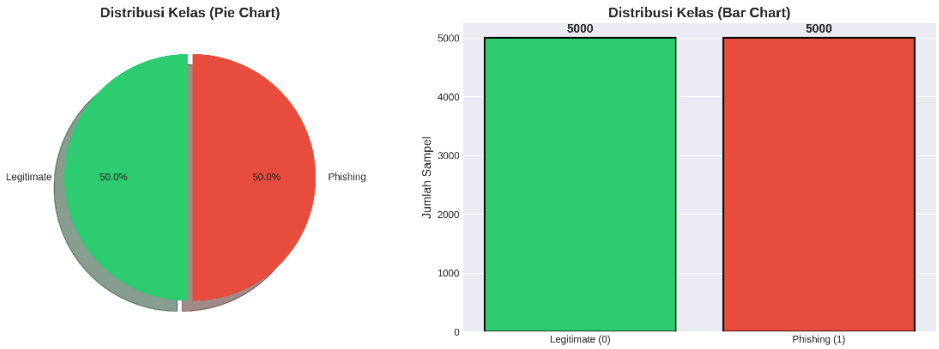
$$F1 = 2 \times \frac{Precision \times Recall}{Precision \times Recall}$$

Selain itu, AUC-ROC digunakan untuk mengukur kemampuan model untuk membedakan antara URL *phising* dan *legitimate* secara keseluruhan. Nilai AUC-ROC berkisar 0 hingga 1, di mana nilai mendekati 1 berarti model sangat baik dalam membedakan kedua kelas. Analisis *overfitting* juga dilakukan melalui *learning curves* untuk memastikan model tidak hanya menghafal data latih tetapi juga mampu menggeneralisasi ke data baru.

3. ANALISA DAN PEMBAHASAN

3.1 Exploratory Data Analysis

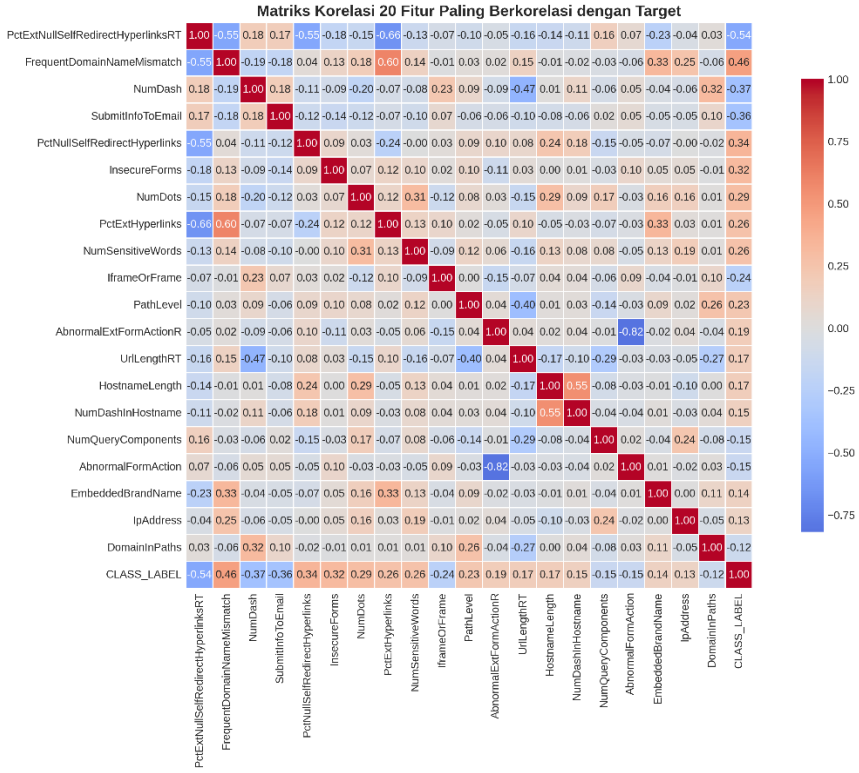
Langkah pertama sebelum membangun model adalah memeriksa kondisi data. Hasil analisis menunjukkan bahwa seluruh 10.000 sampel dalam *dataset* terisi lengkap tanpa ada data yang kosong (*missing values*), sehingga data dapat langsung digunakan tanpa perlu diperbaiki terlebih dahulu. Distribusi kelas juga seimbang sempurna, yakni 5.000 URL *legitimate* dan 5.000 URL *phising* (50:50). Keseimbangan ini penting karena jika satu kelas jauh lebih banyak, model cenderung "malas" dan hanya memprediksi kelas mayoritas saja. Visualisasi distribusi kelas ditunjukkan pada Gambar 2.



Gambar 2. Distribusi Kelas pada *Dataset* (*Legitimate* vs *Phising*)

JRIIN : Jurnal Riset Informatika dan Inovasi
Volume 4, No. 4 Tahun 2026
ISSN 3025-0919 (media online)
Hal 1020-1029

Selanjutnya dilakukan analisis korelasi, yaitu mengukur seberapa kuat hubungan antara masing-masing fitur dengan label target (*phising* atau *legitimate*). Semakin tinggi nilai korelasi suatu fitur, semakin berguna fitur tersebut sebagai "petunjuk" bagi model untuk membedakan kedua kelas. Hasil analisis ditunjukkan pada Gambar 3 dan Tabel 2.



Gambar 3. Matriks Korelasi 20 Fitur Paling Berkorelasi dengan Target

Tabel 2. Top 10 Fitur dengan Korelasi Tertinggi terhadap CLASS_LABEL

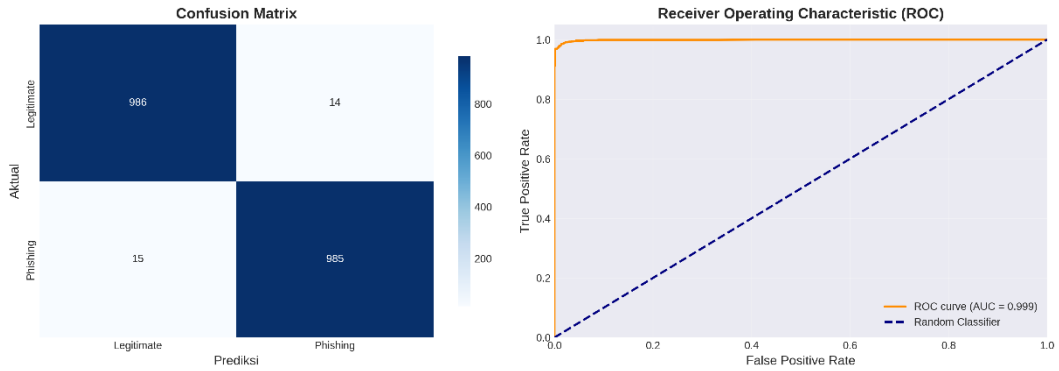
No.	Parameter	Rentang Nilai
1	PctExtNullSelfRedirectHyperlinksRT	-0.54
2	FrequentDomainNameMismatch	0.46
3	NumDash	-0.37
4	SubmitInfoToEmail	-0.36
5	PctNullSelfRedirectHyperlinks	0.34
6	InsecureForms	0.32
7	NumDots	0.29
8	PctExtHyperlinks	0.26
9	NumSensitiveWords	0.26
10	IframeOrFrame	-0.24

Fitur dengan korelasi tertinggi (nilai absolut) adalah PctExtNullSelfRedirectHyperlinksRT (-0,54), yang mengukur persentase *hyperlink* dalam halaman yang bersifat *null*, *self-redirect*, atau mengarahkan ke domain eksternal. Korelasi negatif menunjukkan bahwa semakin tinggi nilai fitur ini, semakin besar kemungkinan URL tersebut merupakan *phising*. Situs *phising* cenderung memiliki peresntasi tinggi pada fitur ini karena halaman palsu meniru tampilan situs asli namun *link*-nya tidak berfungsi atau mengarah ke server penyerang.

JRIIN : Jurnal Riset Informatika dan Inovasi
Volume 4, No. 4 Tahun 2026
ISSN 3025-0919 (media online)
Hal 1020-1029

3.2 Performa Model Baseline

Model *Random Forest baseline* menggunakan 100 pohon keputusan dengan pengaturan bawaan tanpa optimasi apapun. Meskipun tanpa penyesuaian khusus, model ini sudah menunjukkan perfoma yang sangat baik pada data uji (2.000 sampel yang belum pernah dilihat model selama pelatihan). Gambar 4 menyajikan *confusion matrix* dan kurva ROC, sedangkan hasil evaluasi disajikan pada Tabel 3.



Gambar 4. *Confusion Matrix* dan Kurva ROC Model *Baseline*

Tabel 3. Hasil Evaluasi Model Baseline Random Forest

Metrik	Nilai
Accuracy	0.9855
Precision	0.9860
Recall	0.9850
F1-Score	0.9855
AUC-ROC	0.9990

Dari 2.000 sampel uji, *confusion matrix* menunjukkan 986 TN, 985 TP, 14 FP, dan 15 FN. AUC-ROC sebesar 0,999 menunjukkan kemampuan diskriminasi yang sangat tinggi antara kelas *legitimate* dan *phising*.

3.3 Hyperparameter Tuning

Setelah melihat hasil *baseline* yang sudah baik, langkah selanjutnya adalah mencoba meningkatkan performa model melalui *hyperparameter tuning*. Proses ini ibarat mencari "resep terbaik" untuk model – mencoba berbagai kombinasi pengaturan (seperti jumlah pohon, kedalaman pohon, dan lain-lain) untuk menemukan kombinasi yang menghasilkan prediksi paling akurat. Pencarian dilakukan menggunakan *RandomizedSearchCV* yang mencoba 50 kombinasi pengaturan secara acak, masing-masing diuji dengan 5-*fold cross-validation*. Konfigurasi optimal yang ditemukan disajikan pada Tabel 4.

Tabel 4. Hyperparameter Optimal Hasil Tuning

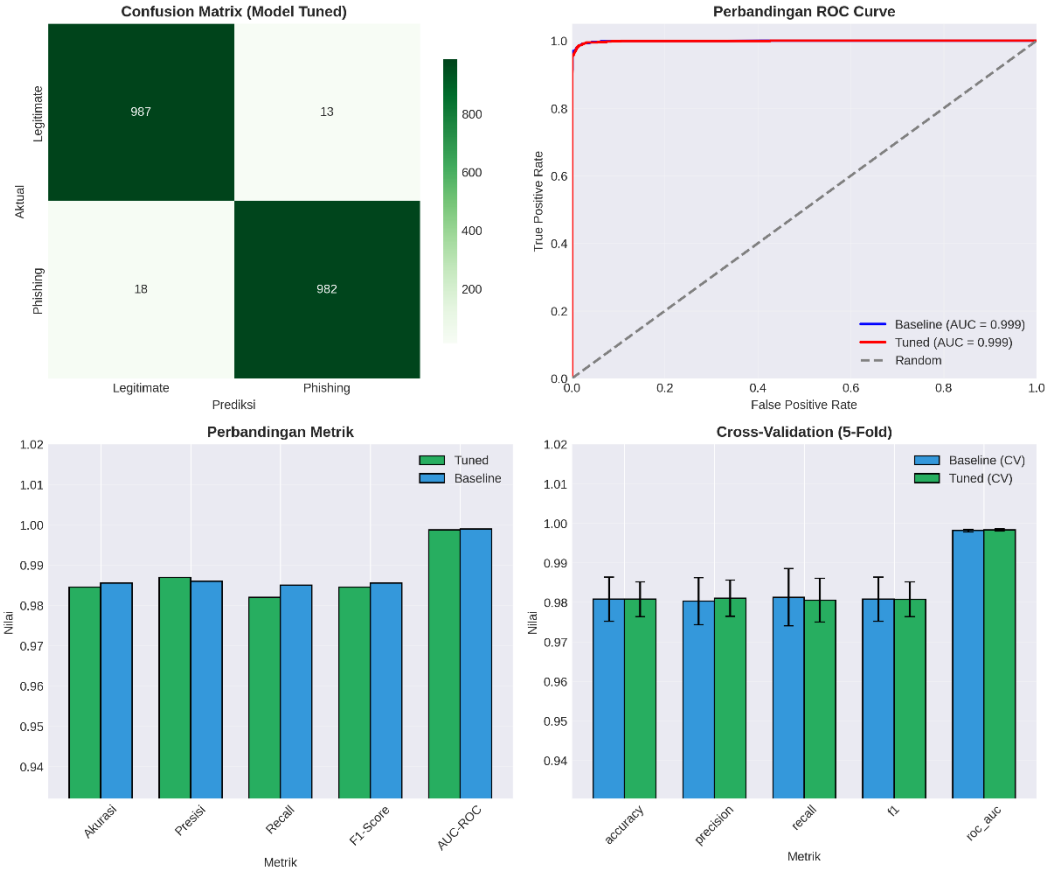
Parameter	Nilai Optimal
n_estimators	500
max_depth	15
min_samples_split	3
min_samples_leaf	1
max_features	Log2

Hasil tuning menunjukkan bahwa model optimal membutuhkan 500 pohon (5 kali lipat dari *baseline*) dengan kedalaman maksimum 15 level. Artinya, model yang lebih kompleks ini menggunakan lebih banyak pohon namun membatasi seberapa dalam setiap pohon boleh bercabang, sebagai upaya menjaga keseimbangan antara akurasi dan efisiensi.

JRIIN : Jurnal Riset Informatika dan Inovasi
Volume 4, No. 4 Tahun 2026
ISSN 3025-0919 (media online)
Hal 1020-1029

3.4 Perbandingan Model Baseline dan Tuned

Pertanyaan utama setelah melakukan *tuning* adalah mengenai apakah model yang sudah dioptimasi benar-benar lebih baik? Gambar 5 menyajikan perbandingan visual kedua model, dan Tabel 5 merangkum hasil evaluasinya.



Gambar 5. Perbandingan Performa Model *Baseline* vs *Tuned*

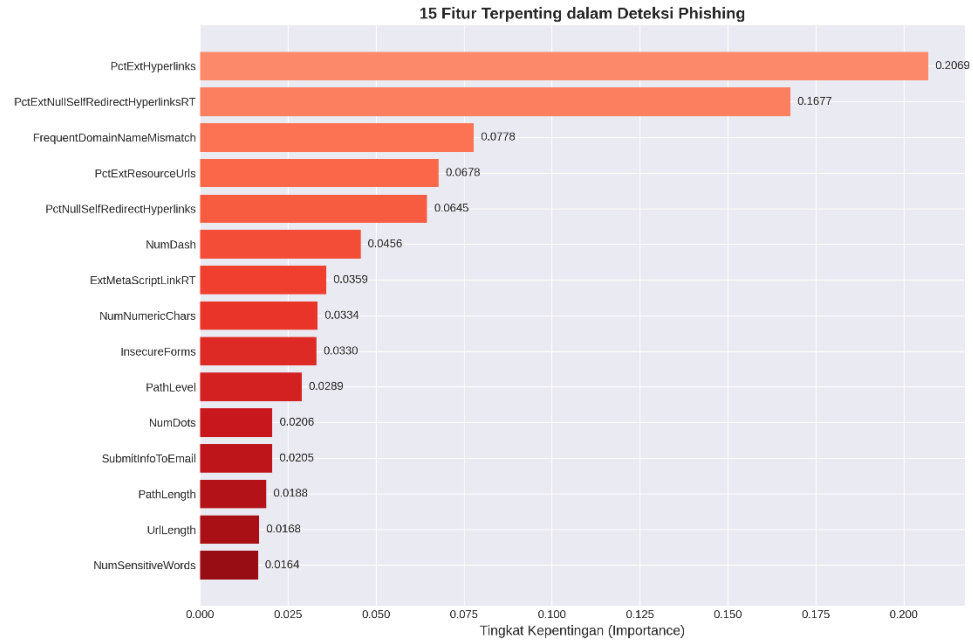
Tabel 5. Perbandingan Performa Model *Baseline* vs *Tuned*

Metrik	Baseline	Tuned
Accuracy	0.9855	0.9845
Precision	0.9860	0.9869
Recall	0.9850	0.9820
F1-Score	0.9855	0.9845
AUC-ROC	0.9990	0.9990

Model *tuned* tidak menunjukkan peningkatan signifikan. *Precision* naik +0,0009, namun *recall* turun -0,0030. *Confusion matrix* model *tuned* menunjukkan 987 TN, 982 TP, 13 FP, dan 18 FN, dibandingkan model *baseline* dengan 986 TN, 985 TP, 14 FP, dan 15 FN. Hal ini terjadi karena model *baseline* sudah mencapai akurasi di atas 98% dan *dataset* yang bersih serta seimbang membuat tugas klasifikasi relatif mudah bagi *Random Forest*. Dalam konteks kemanan siber, *recall* lebih penting daripada *precision* karena URL *phising* yang lolos deteksi berdampak lebih serius daripada URL *legitimate* yang salah diblokir. Oleh karena itu, model *baseline* dengan *recall* lebih tinggi (0,9850 vs 0,9820) direkomendasikan sebagai model final.

JRIIN : Jurnal Riset Informatika dan Inovasi
Volume 4, No. 4 Tahun 2026
ISSN 3025-0919 (media online)
Hal 1020-1029

3.5 Analisis Feature Importance



Gambar 6. Top 10 *Feature Importance* Model *Random Forest*

Salah satu keunggulan *Random Forest* adalah kemampuannya menunjukkan fitur mana yang paling berpengaruh dalam pengambilan keputusan. Ini penting agar kita memahami "apa yang sebenarnya dilihat model" ketika memutuskan sebuah URL berbahaya atau tidak.

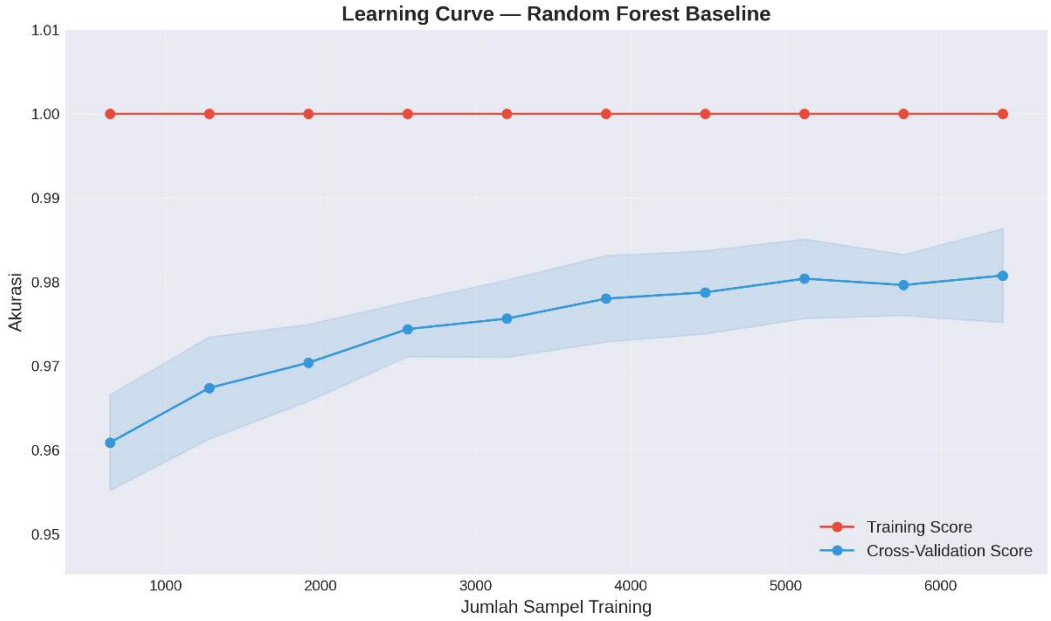
Tabel 6. Top 10 Fitur Berdasarkan Feature Importance

No	Fitur	Importance
1	PctExtHyperlinks	0.2069
2	PctExtNullSelfRedirectHyperlinksRT	0.1677
3	FrequentDomainNameMismatch	0.0778
4	PctExtResourceUrls	0.0678
5	PctNullSelfRedirectHyperlinks	0.0645
6	NumDash	0.0456
7	ExtMetaScriptLinkRT	0.0359
8	NumNumericChars	0.0334
9	InsecureForms	0.0330
10	PathLevel	0.0289

Tiga fitur teratas menyumbang 45,24% dari keseluruhan keputusan model. PctExtHyperlinks (20,69%) mengukur peresentase *hyperlink* yang mengarah ke domain eksternal. PctExtNullSelfRedirectHyperlinksRT (16,77%) mendeteksi *hyperlink* yang bersifat *null* atau melakukan *redirect* ke domain lain. FrequentDomainMismatch (7,78%) menandai ketidaksesuaian antara nama domain yang tertulis di *hyperlink* dengan domain tujuan sebenarnya. Fitur-fitur lainnya seperti PctExtResourceUrls (6,78%) dan PctNullSelfRedirectHyperlinks (6,45%) turut berkontribusi signifikan dalam klasifikasi.

JRIIN : Jurnal Riset Informatika dan Inovasi
Volume 4, No. 4 Tahun 2026
ISSN 3025-0919 (media online)
Hal 1020-1029

3.6 Analisis Overfitting



Gambar 7. *Learning Curve* Model *Random Forest Baseline*

Analisis *learning curves* menunjukkan *gap* antara skor latih dan validasi sebesar 0,0145 untuk model *baseline* dan 0,0131 untuk model *tuned*, keduanya di bawah batas aman 0,02. Hasil ini mengkonfirmasi bahwa kedua model tidak mengalami *overfitting* dan mampu menggeneralisasi dengan baik ke data baru.

3.7 Kajian Etika Profesi

3.7.1 Implikasi Kesalahan Klasifikasi

Dari hasil pengujian, model menghasilkan 14 kasus *False Positive* dan 15 kasus *False Negative*. *False Positive* menyebabkan *website* yang sah diblokir, merugikan pemilik situs dari segi pendapatan dan reputasi. *False Negative* menyebabkan *phising* lolos deteksi, memungkinkan pencurian data sensitif pengguna. Dampak *False Negative* dinilai lebih serius karena melanggar hak privasi pengguna sebagaimana diatur dalam UU PDP No.27 Tahun 2022.

3.7.2 Tanggung Jawab Profesional

Pengembang sistem keamanan memiliki tanggung jawab etis meliputi transparansi model melalui *feature importance*, akuntabilitas dengan menyediakan mekanisme pelaporan *false positive*, menjaga privasi dengan hanya menganalisis karakteristik URL tanpa mengumpulkan data pribadi, dan non-diskriminasi terhadap *website* dari kelompok tertentu.

3.7.3 Kepatuhan Regulasi Indonesia

Penerapan sistem deteksi *phising* di Indonesia harus mematuhi UU ITE (UU No. 11/2008 jo. UU No. 19/2016) sebagai dasar hukum transaksi elektronik, dan UU PDP (UU No. 27/2022) yang mewajibkan perlindungan data pribadi. Implementasi juga mengacu pada standar internasional ISO 27001, NIST *Cybersecurity Framework*, dan OWASP.

4. KESIMPULAN

Penelitian ini berhasil mengembangkan sistem deteksi dini *phising website* berbasis karakteristik URL menggunakan algoritma *Random Forest* dengan performa yang sangat baik. Model *baseline Random Forest* mencapai akurasi 98,55% dengan AUC-ROC 99,90%, menunjukkan efektivitas pendekatan ini dalam membedakan *legitimate website* dan *phising website*.

JRIIN : Jurnal Riset Informatika dan Inovasi
Volume 4, No. 4 Tahun 2026
ISSN 3025-0919 (media online)
Hal 1020-1029

Fitur-fitur yang paling berpengaruh dalam deteksi adalah PctExtHyperlinks, PctExtNullSelfRedirectHyperlinksRT, dan FrequentDomainNameMismatch, yang berkaitan dengan pola *hyperlink* dan ketidaksesuaian domain.

Hyperparameter tuning tidak menghasilkan peningkatan signifikan, mengkonfirmasi bahwa konfigurasi *default Random Forest* sudah optimal untuk *dataset* ini. Model *baseline* direkomendasikan karena memiliki *recall* yang lebih tinggi, yang krusial dalam konteks deteksi ancaman keamanan siber. Dari perspektif etika profesi, pengembangan sistem deteksi *phising* harus memperhatikan transparansi, akuntabilitas, privasi data, dan kepatuhan terhadap regulasi yang berlaku di Indonesia. Implikasi kesalahan klasifikasi harus dikelola secara profesional dan bertanggung jawab.

Pengembangan di masa depan dapat diarahkan pada penggunaan *dataset* yang lebih besar dan beragam agar model memiliki generalisasi yang lebih baik terhadap variasi *phising* terbaru. Integrasi fitur berbasis konten halaman web juga berpotensi meningkatkan akurasi deteksi karena dapat menangkap indikator visual yang tidak terepresentasi dalam fitur URL saja. Penerapan teknik *deep learning* dapat dieksplorasi untuk menangani pola yang lebih kompleks, sementara pengembangan sistem *real-time* dalam bentuk ekstensi *browser* atau API keamanan akan memungkinkan implementasi langsung yang bermanfaat bagi pengguna internet secara luas.

REFERENCES

Breiman, L. (2001). Random Forests. *Machine Learning*, *45*(1), 5–32. https://doi.org/10.1023/A:1010933404324

Jain, A. K., & Gupta, B. B. (2019). A machine learning based approach for phishing detection using hyperlinks information. *Journal of Ambient Intelligence and Humanized Computing*, *10*(5), 2015–2028. https://doi.org/10.1007/s12652-018-0798-z

Karim, A., Shahroz, M., Mustofa, K., Belhaouari, S. B., & Joga, S. R. K. (2023). Phishing Detection System Through Hybrid Machine Learning Based on URL. *IEEE Access*, *11*, 36805–36822. https://doi.org/10.1109/ACCESS.2023.3252366

Rao, R. S., & Pais, A. R. (2018). Detection of phishing websites using an efficient feature-based machine learning framework. *Neural Computing and Applications*. https://doi.org/10.1007/s00521-017-3305-0

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

Shashwat. (2020). *Phishing Dataset for Machine Learning*. https://www.kaggle.com/datasets/shashwatwork/phishing-dataset-for-machine-learning

Sumartono, E., Harliyanto, R., Situmeang, S. M. T., Siagian, D. S., & Septaria, E. (2024). The Legal Implications of Data Privacy Laws, Cybersecurity Regulations, and AI Ethics in a Digital Society. *The Journal of Academic Science*, *1*(2), 103–110. https://doi.org/10.59613/29qypw51

Yang, R., Zheng, K., Wu, B., Wu, C., & Wang, X. (2021). Phishing Website Detection Based on Deep Convolutional Neural Network and Random Forest Ensemble Learning. *Sensors*, *21*(24), 8281. https://doi.org/10.3390/s21248281