DATA ENTHUSIAST JOURNAL

OPTIMALISASI AKURASI DETEKSI URL PHISHING DENGAN HYPERPARAMETER-TUNING RFECV DAN GRID SEARCH PADA ALGORITMA RANDOM FOREST

Catherine Vanya Pangemanan[1]

1. Universitas Koperasi Indonesia
Kawasan Pendidikan Tinggi Jl. Raya Jatinangor
No.KM. 20, RW.5, Cibeusi, Kec. Jatinangor,
Kabupaten Sumedang, Jawa Barat 45363
Email : catherinevannya742@gmail.com

ABSTRAK

Dalam konteks metode supervised learning, penelitian ini bertujuan meningkatkan akurasi model klasifikasi dalam mendeteksi URL phishing menggunakan algoritma Random Forest yang dikombinasikan dengan teknik hyperparameter tuning, yaitu Recursive Feature Elimination with Cross-Validation (RFECV) dan Grid Search. Dataset PhiUSIIL Phishing URL yang diakuisisi dari UCL Machine Learning hanya dimanfaatkan sebanyak 10.000 baris data dalam penelitian ini agar mempermudah proses. Data dibagi menjadi 80% data training dan 20% data testing. Model dilatih menggunakan Random Forest dan dioptimalkan dengan hyperparameter tuning RFECV lalu Grid Search, yang menghasilkan akurasi, precision, recall, dan F1 score sebesar 100%. Fitur-fitur 'URLSimilarityIndex', 'LineOfCode', dan 'NoOfExternalRef' memberikan kontribusi terbesar terhadap prediksi. Hasil penelitian ini menunjukkan bahwa pendekatan teknik hyperparameter tuning dan algoritma yang dipilih lebih efektif dibandingkan penelitian sebelumnya, yang hanya mencapai akurasi tertinggi 99,97%. Selain itu, penelitian ini juga mengidentifikasi pentingnya fitur 'URLLength' dalam meningkatkan kinerja model. Temuan ini menegaskan bahwa teknik hyperparameter tuning yang tepat dapat meningkatkan kinerja model klasifikasi URL phishing secara signifikan dan memberikan kontribusi penting dalam bidang keamanan siber.

Kata kunci: *URL phising, Random Forest, RFECV, Grid Search*

1. PENDAHULUAN

Banyak publikasi ilmiah dan akademik dari situs seperti *Google Scholar*, Crossref, dan *ScienceDirect* telah membahas penggunaan berbagai algoritma dalam metode *supervised learning* untuk menangani *dataset* PhiUSIIL Phishing URL *Website* dari tahun 2010 hingga 2024 [4,2,8,3,9,7,5] sebagai konsekuensi atas berkembangnya trik *phishing* yang semakin kompleks, sehingga memerlukan pendekatan komputasi tingkat lanjut.

Penelitian dengan metode *supervised learning* khususnya algoritma *Random Forest* untuk memproses *dataset* PhiUSIIL Phising URL Website masih terbatas. Hal ini mungkin disebabkan *Random Forest* termasuk metode *ensamble*, yang menjadikan algoritma ini memiliki tingkat kompleksitas komputasi yang tinggi dan menghabiskan banyak waktu.

Sebuah penelitian mencakup algoritma *Random Forest* dan *XGBoost* menghasilkan akurasi model sebesar 97,44% dan 98,27% [1]. Sementara penelitian lain menggunakan kombinasi Regresi Logistik dan *Mutual Information* mencapai akurasi 99,97% [9], mendekati akurasi ideal.

Hasil-hasil tersebut menunjukkan dataset ini cocok untuk diolah baik menggunakan algoritma klasifikasi seperti *Random Forest*, prediksi seperti Regresi Logistik maupun gabungan keduanya seperti *Gradient Boosting Machines*. Namun algoritma *Random forest* tergolong rentan menghasilkan angka akurasi model yang rendah jika tanpa melalui tahap *fine-tuning*.

Permasalahan tersebut memunculkan celah perbaikan yaitu dengan menambahkan teknik *hyperparameter tuning* yang tepat ke dalam *Random Forest*. Maka hal tersebut memotivasi

penelitian ini terlaksana dengan tujuan utama meningkatkan angka akurasi model klasifikasi *Random Forest* dalam mengenali URL *phising* dan resmi namun model tetap mampu terhindar dari *overfitting*, dengan menawarkan kebaruan metode berupa penggabungan *Random Forest* dengan teknik *hyperparameter tuning* lainnya.

2. TINJAUAN PUSTAKA

Abdul Samad et al. [1] telah melakukan penelitian terhadap beragam penggunaan algoritma machine learning, termasuk *Random Forest, XGBoost, Gradient Boosting,* dan Regresi Logistik, untuk dataset PhiUSIIL Phising URL *Website* ini. Dalam penelitiannya, Abdul Samad et al menegaskan bahwa *fine-tuning hyperparameter* sangat penting untuk meningkatkan kinerja model. Algoritma *ensamble* seperti *Random Forest* dan *XGBoost* menunjukkan peningkatan performa yang signifikan setelah *fine-tuning. XGBoost* mencapai akurasi tertinggi sebesar 97,2% pasca *fine-tuning*, dengan *precision* 96,9%, *recall* 97,0%, dan *F1-score* 97,0%. Sementara *Random Forest* mencapai akurasi model pasca *fine-tuning* sebesar 95,3%, dengan *precision* 94,8%, *recall* 95,1%, dan *F1-score* 95,0%.

Abdul Samad et al. mengungkapkan URL *phishing* biasanya memiliki ciri-ciri seperti panjang yang berlebihan, penggunaan simbol-simbol khusus, domain mencurigakan, banyak subdomain, penggunaan protokol "http://" alih-alih "https://", teks yang meniru merek terkenal, domain baru dengan umur pendek, kata kunci sensitif, penggunaan alamat IP alih-alih nama domain, serta tanda-tanda penipuan seperti ejaan salah dan karakter Unicode yang mirip. Algoritma seperti *Random Forest* dan *XGBoost* dapat mendeteksi URL phishing dengan akurat ketika fitur-fitur ini dianalisis [1].

Selain itu penelitian yang dicetuskan oleh penemu metode PhiUSIIL phising URL *Website data maining,* Prasad et al, hanya mencapai akurasi model sebesar 99,24%. [6,7].

Lalu terdapat penelitian yang sejalan dengan [1,6,7], menggunakan algoritma Regresi Linier Logistik dan *Mutual Information* [9] menghasilkan tingkat akurasi tertinggi daripada penelitian [1,7] yaitu mencapai 99,97% dengan *precision* 99,97%, *F1-score* 99,97% dan *recall* 99,97%. Vjrobol et al [9] juga menyoroti bahwa 'Indeks Kemiripan URL', 'Garis Kode', *'No Of External Ref', 'No Of Image', dan 'No Of Self Ref',* merupakan fitur terpilih karena memiliki angka persentase kontribusi tertinggi pada model prediksi Regresi Linear Logistik pada kasus *phising URL website* yang berkorespondensi dengan ciri adanya upaya *phishing* [1,6,7,9].

Berdasarkan [1,9] walaupun algoritma *machine learning* seperti *Random Forest* memiliki kemampuan untuk mengelola dataset besar dan kompleks serta mengurangi risiko *overfitting* dan *underfitting* melalui *ensemble* namun *Hyperparameter tuning (ht)* menjadi hal krusial dalam mengoptimalkan performa model dengan mencari kombinasi parameter yang optimal sebelum proses pelatihan dimulai.

Tujuan ht adalah meningkatkan akurasi, *precision*, *recall*, dan metrik evaluasi lainnya, sambil menjaga keseimbangan antara bias dan varians untuk menghindari *overfitting*. Proses ini juga meningkatkan generalisasi model agar dapat bekerja dengan baik pada data baru yang tidak pernah dilihat sebelumnya. Penelitian ini berhipotesis jika kita menggunakan algoritma dan teknik *hyperparameter tuning* yang tepat, kita dapat menghasilkan model klasifikasi dengan kinerja terbaik dalam mengidentifikasi *URL phishing* dan resmi.

3. METODE PENELITIAN

Pemrosesan dataset menggunakan bahasa pemrograman Python. Setelah pembersihan data dari fitur-fitur kategorikal, data dinormalisasi dan distandarisasi. *Dataset* dibagi menjadi 80% data training dan 20% data testing. Menggunakan teknik *hyperparameter tuning* berupa *Recursive Feature Elimination with Cross-Validation (RFECV)* untuk pemilihan fitur optimal, model *Random Forest* dilatih dan dioptimalkan melalui *Grid Search*. Evaluasi model melibatkan metrik seperti *accuracy, precision, recall, dan F1 score* menggunakan *library ScikitLearn*, serta analisis *Confusion Matrix*. Fitur-fitur yang paling penting diidentifikasi berdasarkan skor "*Importances*" dari model *Random Forest*.

3.1. Deskripsi Dataset

Studi ini menggunakan dataset *PhiUSIIL Phishing URL* yang diakuisisi dari *UCI Machine Learning* Repository [1], dengan penggunaan 10.000 baris data untuk mempermudah proses dan mengurangi kompleksitas. Akuisisi dataset *PhiUSIIL Phishing URL* bersumber dari situs UCI Machine Learning Repository [1], yang diakses pada bulan Juli tahun 2024. Sebagian besar *URL website* saat pengumpulan dan analisis dataset, merupakan URL terbaru pada tahun 2024. Dataset terdiri atas 134.850 URL resmi dan 100.945 URL *phising.* Jumlah dari kedua dataset URL resmi dan *phising* yang digunakan pada penelitian ini hanya 10.000 baris data karena mempertimbangkan lamanya durasi dan kerumitan pengeksekusian algoritma *Random Forest, Recrusive Feature Elimination Cross Validation* dan *Grid Search* yang akan digunakan untuk pemrosesan dataset ini. *Dataset* memiliki 56 fitur, dengan 1 fitur target bernama *label.* Di dalam fitur *labes* terdapat nilai 1 dan 0, dimana 1 berkorespondensi dengan *URL* resmi dan 0 dengan *URL phising*.

Prasad et al [6,7] sebagai pencetus metode baru pendeteksian *URL phising* mengemukakan bahwa, seluruh fitur dalam *dataset* ini telah diekstrak dari kode sumber halaman web dan *URL* menggunakan metode PhiUSIIL, yaitu kerangka deteksi *URL phishing*, yang didukung oleh beragam profil keamanan berdasarkan indeks kesamaan dan pembelajaran bertahap. Kerangka kerja ini memanfaatkan model pembelajaran mesin untuk mengenali *URL* yang mencurigakan dan membedakannya dari *URL* resmi.

3.2. Optimalisasi Pemilihan Fitur dengan Recrusive Feature Elimination with Cross-Validation

RFECV menentukan jumlah fitur optimal dalam model Random Forest eliminasi fitur kurang penting secara bertahap dan berulang kemudian mempertahankan fitur paling berpengaruh di setiap iterasinya sebagai hasil dari validasi silang untuk mengevaluasi kinerja model hingga mencapai kombinasi fitur terbaik. RFECV menghasilkan jumlah fitur optimal dan mengurangi risiko *overfitting* model. Sehingga model menjadi lebih sederhana dan mungkin lebih akurat.

3.3. Hyperparameter tuning dengan dan Grid Search

Grid Search pencarian parameter optimal dalam model menggunakan kombinasi parameter yang telah ditentukan sebelumnya berdasarkan matrik evaluasi dengan validasi silang. Metode ini memungkinkan kita menemukan kombinasi parameter yang terbaik untuk Random Forest, sehingga performa model dapat dimaksimalkan.

3.4. Random Forest

Metode *ensamble* yang memanfaatkan kombinasi dari banyak *Decision Tree* yang dibentuk secara acak dengan subset acak dari fitur dan observasi di dalamnya. Seperti *Decision Tree,* algoritma *Random Forest* memiliki fungsi klasifikasi dengan mekanisme *voting* (menentukan keputusan didasari suara terbanyak) yang menentukan output mana yang akan dipakai hasil dari validasi silang *data test. Random Forest* menangani dataset besar, banyak fitur dan kompleks tanpa memerlukan *tuning parameter* yang rumit (Breiman, 2001).

3.5. Evaluasi Model dengan Confussion Matrix

Mengukur kinerja model yang menghasilkan *binary classification* dapat menggunakan matriks evaluasi ini karena *Confussion Matrix* pada model *Random Forest* terdapat nilai *precision* (seberapa baik model dapat menangkap prediksi positif), *sensitivity* atau *recall* (seberapa banyak model telah lalai dalam menangkap data yang seharusnya diprediksi positif), dan skor F1 (menggambarkan kinerja model secara seimbang dengan mempertimbangkan kedua jenis kesalahan klasifikasi tersebut). *Precision* tinggi belum pasti diikuti *recall* tinggi maupun sebaliknya. Jika ada dua model yang memiliki *precision* hampir sama-sama tinggi, maka bandingkan nilai *recall* tertingginya.

Nomor	Nama Variabel	Deskripsi Singkat	Tipe Fitur	Alasan Pemilihan Fitur
1	FILENAME	Nama file dari URL	Kategori kal	Nama file tertentu mungkin lebih sering digunakan oleh situs phishing.
2	URL	URL lengkap dari situs web	Teks	URL dapat dianalisis untuk mengidentifikasi pola yang mencurigakan.
3	URLLength	Panjang dari URL	Numerik	URL yang sangat panjang seringkali digunakan untuk menyembunyikan niat jahat.
4	Domain	Domain dari URL	Teks	Domain dapat dianalisis untuk keaslian dan kepercayaan.
5	DomainLength	Panjang dari domain URL	Numerik	Domain yang sangat panjang mungkin mencurigakan.
6	IsDomainIP	Apakah domain merupakan alamat IP (1 jika ya, 0 jika tidak)	Kategori kal	Penggunaan alamat IP alih-alih nama domain seringkali terkait dengan situs phishing.
7	TLD	Top-Level Domain (misalnya .com, .org)	Kategori kal	TLD tertentu mungkin lebih sering digunakan oleh situs phishing.
8	URLSimilarityIndex	Indeks kesamaan URL terhadap URL sah	Numerik	Phishing URL seringkali mirip dengan URL sah untuk mengecoh pengguna.
9	CharContinuationRate	Tingkat kelanjutan karakter dalam URL	Numerik	URL phishing mungkin memiliki pola karakter yang tidak biasa.
10	TLDLegitimateProb	Probabilitas bahwa TLD adalah sah	Numerik	TLD dengan reputasi buruk lebih mungkin digunakan untuk phishing.
11	URLCharProb	Probabilitas karakter dalam URL	Numerik	Analisis frekuensi karakter dapat membantu mengidentifikasi URL phishing.
12	TLDLength	Panjang dari TLD	Numerik	TLD yang sangat panjang bisa menjadi tanda perlu diperiksa.
13	NoOfSubDomain	Jumlah subdomain dalam URL	Numerik	Banyak subdomain bisa digunakan untuk menyembunyikan niat jahat.
14	HasObfuscation	Apakah URL memiliki pengaburan (1 jika ya, 0 jika tidak)	Kategori kal	Pengaburan karakter sering digunakan untuk menghindari deteksi.
15	NoOfObfuscatedChar	Jumlah karakter yang diobfuscate dalam URL	Numerik	Lebih banyak karakter yang diobfuscate dapat menandakan URL phishing.
16	ObfuscationRatio	Rasio pengaburan karakter dalam URL	Numerik	Rasio pengaburan yang tinggi dapat menjadi indikator phishing.
17	NoOfLettersInURL	Jumlah huruf dalam URL	Numerik	Analisis jumlah huruf dapat mengungkap pola mencurigakan.
18	LetterRatioInURL	Rasio huruf dalam URL	Numerik	Rasio huruf yang tidak biasa bisa menjadi tanda phishing.
19	NoOfDegitsInURL	Jumlah digit dalam URL	Numerik	Banyak digit dalam URL bisa digunakan untuk menyamarkan niat jahat.
20	DegitRatioInURL	Rasio digit dalam URL	Numerik	Rasio digit yang tinggi bisa menandakan phishing.
21	NoOfEqualsInURL	Jumlah tanda sama dengan (=) dalam URL	Numerik	Penggunaan simbol "=" bisa menjadi indikasi URL phishing.
22	NoOfQMarkInURL	Jumlah tanda tanya (?) dalam URL	Numerik	Tanda tanya sering digunakan dalam URL phishing untuk menyamarkan parameter.
23	NoOfAmpersandInURL	Jumlah tanda dan (&) dalam URL	Numerik	Tanda "&" bisa menambahkan parameter mencurigakan dalam URL.
24	NoOfOtherSpecialCharsInURL	Jumlah karakter khusus lainnya dalam URL	Numerik	Karakter khusus sering digunakan dalam URL phishing untuk pengaburan.
25	SpecialCharRatioInURL	Rasio karakter khusus yang umum dalam URL	Numerik	Rasio karakter khusus yang tinggi bisa menjadi tanda peringatan.
26	IsHTTPS	Apakah URL menggunakan HTTPS (1 jika ya, 0 jika tidak)	Kategori kal	Situs phishing sering kali tidak menggunakan HTTPS.
27	LineOfCode	Jumlah baris kode dalam halaman	Numerik	Situs phishing mungkin memiliki lebih sedikit baris kode.
28	LargestLineLength	Panjang baris terbesar dalam kode	Numerik	Analisis panjang baris terbesar dapat mengungkap pola mencurigakan.
29	HasTitle	Apakah halaman memiliki judul (1 jika ya, 0 jika tidak)	Kategori kal	Halaman tanpa judul mungkin mencurigakan.
30	Title	Judul dari halaman	Teks	Judul halaman dapat dibandingkan dengan domain untuk keabsahan.
31	DomainTitleMatchScore	Skor kecocokan judul domain	Numerik	Skor kecocokan rendah dapat mengindikasikan situs phishing.
32	URLTitleMatchScore	Skor kecocokan judul URL	Numerik	Skor kecocokan rendah dapat mengindikasikan URL phishing.
33	HasFavicon	Apakah halaman memiliki favicon (1 jika ya, 0 jika tidak)	Kategori kal	Halaman tanpa favicon bisa jadi situs phishing.
34	Robots	Apakah halaman memiliki file robots.txt (1 jika ya, 0 jika tidak)	Kategori kal	Situs phishing mungkin tidak memiliki file robots.txt.
35	IsResponsive	Apakah halaman responsif (1 jika ya, 0 jika tidak)	Kategori kal	Halaman yang tidak responsif bisa menjadi tanda peringatan.
36	NoOfURLRedirect	Jumlah pengalihan URL	Numerik	Banyak pengalihan URL dapat digunakan untuk menyembunyikan niat jahat.
37	NoOfSelfRedirect	Jumlah pengalihan diri	Numerik	Pengalihan diri yang berlebihan bisa menjadi indikasi phishing.
38	HasDescription	Apakah halaman memiliki deskripsi (1 jika ya, 0 jika tidak)	Kategori kal	Halaman tanpa deskripsi mungkin mencurigakan.
39	NoOfPopup	Jumlah popup dalam halaman	Numerik	Popup berlebihan sering digunakan dalam situs phishing.
40	NoOfIFrame	Jumlah iframe dalam halaman	Numerik	Banyak iframe bisa digunakan untuk menyembunyikan konten berbahaya.
41	HasExternalFormSubmit	Apakah ada pengiriman form eksternal (1 jika ya, 0 jika tidak)	Kategori kal	Pengiriman form eksternal dapat menjadi tanda phishing.
42	HasSocialNet	Apakah ada tautan ke jejaring sosial (1 jika ya, 0 jika tidak)	Kategori kal	Halaman tanpa tautan jejaring sosial bisa mencurigakan.
43	HasSubmitButton	Apakah halaman memiliki tombol submit (1 jika ya, 0 jika tidak)	Kategori kal	Halaman tanpa tombol submit bisa jadi situs phishing.
44	HasHiddenFields	Apakah halaman memiliki field tersembunyi (1 jika ya, 0 jika tidak)	Kategori kal	Field tersembunyi dapat digunakan untuk mengumpulkan data tanpa sepengetahuan pengguna.
45	HasPasswordField	Apakah halaman memiliki field kata sandi (1 jika ya, 0 jika tidak)	Kategori kal	Field kata sandi pada situs tidak resmi bisa mencurigakan.
46	Bank	Apakah terkait dengan bank (1 jika ya, 0 jika tidak)	Kategori kal	URL yang mengaku terkait bank perlu diperiksa dengan lebih hati-hati.
47	Pay	Apakah terkait dengan pembayaran (1 jika ya, 0 jika tidak)	Kategori kal	URL terkait pembayaran perlu diperiksa untuk keamanan.
48	Crypto	Apakah terkait dengan cryptocurrency (1 jika ya, 0 jika tidak)	Kategori kal	URL terkait cryptocurrency sering dijadikan target phishing.
48	HasCopyrightInfo	Apakah halaman memiliki informasi hak cipta (1 jika ya, 0 jika tidak)	Kategori kal	Halaman tanpa informasi hak cipta bisa mencurigakan.
50	NoOfImage	Jumlah gambar dalam halaman	Numerik	Situs phishing mungkin memiliki lebih sedikit gambar.
51	NoOfCSS	Jumlah file CSS dalam halaman	Numerik	Situs phishing mungkin memiliki lebih sedikit file CSS.
52	NoOfJS	Jumlah file JS dalam halaman	Numerik	Situs phishing mungkin memiliki lebih sedikit file JS.
53	NoOfSelfRef	Jumlah referensi diri dalam halaman	Numerik	Banyak referensi diri bisa menjadi tanda peringatan.
54	NoOfEmptyRef	Jumlah referensi kosong dalam halaman	Numerik	Referensi kosong sering digunakan untuk menyembunyikan niat jahat.
55	NoOfExternalRef	Jumlah referensi eksternal dalam halaman	Numerik	Banyak referensi eksternal bisa menjadi tanda peringatan.
56	label	abel klasifikasi, apakah phishing (1) atau bukan (0)	Kategorikal	Label ini digunakan untuk pelatihan dan evaluasi model klasifikasi.

Gambar 1 Penjelasan fitur pada Dataset PhiUSIIL Phising URL Website

4. HASIL PENELITIAN DAN PEMBAHASAN

Penggabungan Random Forest, RFECV (Recursive Feature Elimination with Cross-Validation), Grid Search, dan matriks kebingungan melewati beberapa tahap berikut.

4.1. *Preprocessing Data*

Pembersihan Data *(Data Cleaning)* sebagai langkah pertama dalam *preprocessing data*, dilakukan penghapusan fitur bertipe data kategorikal dari dataset yaitu fitur 'label', 'FILENAME', 'URL', 'Domain' dan 'TLD', 'Title'. Data yang telah dibersihkan kemudian dikelompokkan ke dalam variabel X (variabel prediktor) dan variabel Y (variabel target).

Dari seluruh nilai fitur variabel X dapat dilihat bahwa terdapat angka yang beragam. Mengatasi hal tersebut maka tahap proses selanjutnya adalah Transformasi Data *(Data Transformation)*, yaitu normalisasi dan standarisasi. Normalisasi berfungsi mengubah nilai fitur ke skala yang sama, misalnya menggunakan skala 0-1, untuk memastikan bahwa fitur dengan rentang nilai yang berbeda tidak mendominasi model. Standarisasi berfungsi mengubah nilai fitur sehingga memiliki mean 0 dan standar deviasi 1. Ini membantu beberapa algoritma pembelajaran mesin yang bekerja lebih baik dengan data yang memiliki distribusi normal.

Pembagian Data (Data Splitting) menjadi langkah terakhir pada preprocessing ini. *Data splitting* membagi dataset menjadi 20% data testing dan 80% data training. Digunakan random state 42 untuk pengambilan baris data untuk proses tersebut.

4.2. *Feature Selection*

Recursive Feature Elimination with Cross-Validation (RFECV), akan menggunakan $n_1 = 50$ (jumlah fitur awal setelah dibersihkan dari 6 fitur lainnya) untuk nilai cv pada *training* dan *testing* tahap pertama. Lalu didapatkan jumlah fitur optimal n_2 = 3 yaitu 'URLSimilarityIndex', 'LineOfCode' dan 'NoOfExternalRef'. Nilai n_2 ini selanjutnya akan digunakan sebagai nilai cv pada tahap Pelatihan Model menggunakan *Grid Search*.

4.3. *Model Training*

Metode Grid Search dimulai dengan inisialisasi algoritma Random Forest. Kemudian mendefinsikan parameter yang digunakan yaitu 'n_estimators': [100, 200, 300], 'max_depth': [None, 10, 20, 30] dan 'min_samples_split': [2, 5, 10]. Setelah itu perform Grid Search akan mengeksekusi *training* dan *testing* model sesuai proporsi yang telah ditetapkan di tahap sebelumnya agar dapat dipilih parameter terbaik. Parameter terbaik yang terpilih adalah 'max_depth': None, 'min_samples_split': 2, 'n_estimators': 100 dan dengan patokan ini algoritma Random Forest akan melatih model terhadap parameter tersebut.

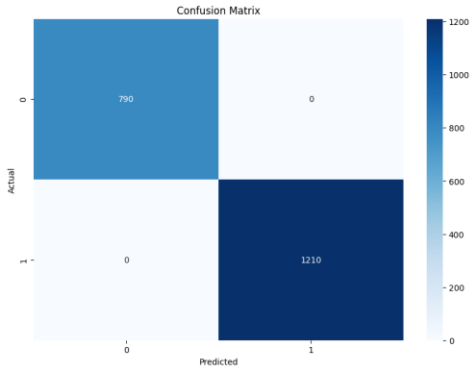
4.4. *Model Evaluation*

Menggunakan library ScikitLearn pada Python memungkinkan kita mengevaluasi model yang telah dilatih dengan parameter terbaik. Library ini menghitung *accuracy score*, *precission*, *recall*, dan F1.

Evaluasi Model	Nilai
Accuracy	1
Precision	1
Recall	1
F1 Score	1

Gambar 2 Evaluasi Model dengan Library ScikitLearn Python

Selain itu library ScikitLearn memperhitungkan *Matrix Confussion* yang dapat divisualisasikan menggunakan *heatmap.*



Gambar 3 Confussion Matrix

Confussion Matrix memberikan informasi mengenai nilai *true positive* = 790 kasus, *false positive* = 0 kasus, *false negative* = 0 kasus dan *true negative* = 1.210 kasus, yang dapat dijelaskan hasil pendeteksian dataset oleh model, yang telah dibangun sebelumnya melalui *hyperparameter tuning* REFCV& Grid Search dalam Random Forest. Angka 0 pada FP dan FN menunjukkan model telah bekerja dalam keadaan ideal karena tidak keliru dalam memprediksi untuk membedakan URL phising dan resmi dibandingkan yang terjadi sesungguhnya di kenyataan. Nampak hanya 2.000 baris data yang terdeteksi sebagai kasus URL phising dan resmi dari total 10.000 baris data. Namun tenang saja, jumlah tersebut adalah hasil pembagian proporsi data testing yang hanya sebesar 20% atau sama dengan 2.000 baris data.

4.5. *Feature Analysis*

Melengkapi hasil perhitungan metode REFCV hingga Random Forest, pada tahap ini kita akan menampilkan daftar skor dari *Future Importance.* Daftar fitur penting yang diurutkan berdasarkan nilai "*Importance*" dari model Random Forest adalah penjabaran lebih lanjut dari hasil analisis Recursive Feature Elimination with Cross-Validation (RFECV) sebelumnya (n_2). RFECV menunjukkan bahwa tiga fitur optimal adalah 'URLSimilarityIndex', 'LineOfCode', dan 'NoOfExternalRef', yang memberikan performa terbaik pada model.

Daftar Feature Importances ini mengurutkan semua fitur berdasarkan skor pentingnya kontribusi mereka terhadap model. Fitur dengan nilai "Importance" yang tinggi memberikan kontribusi besar terhadap prediksi model. Dari daftar ini, fitur dengan skor kontribusi tertinggi adalah 'URLSimilarityIndex' (25%), 'NoOfExternalRef' (16%), dan 'LineOfCode' (11%). Hasil ini memperkuat temuan pada penelitian sebelumnya yang sama-sama menyebutkan bahwa ketiga fitur tersebut merupakan fitur dengan kontribusi tertinggi dalam model [9].

5. KESIMPULAN

Penelitian ini membuktikan bahwa kombinasi algoritma klasifikasi Random Forest dengan teknik hyperparameter tuning seperti RFECV dan Grid Search menghasilkan performa optimal dalam mengidentifikasi URL phishing dan resmi. Model ini mencapai akurasi, precision, recall, dan F1 score sebesar 100%, menunjukkan tidak adanya kesalahan dalam prediksi. Fitur 'URLSimilarityIndex' (25%), 'LineOfCode' (11%), dan 'NoOfExternalRef' (16%) memberikan kontribusi terbesar terhadap prediksi.

Hasil penelitian ini menggunguli penelitian sebelumnya. Abdul Samad et al. [1] melaporkan akurasi tertinggi 97,2% dengan XGBoost dan 95,3% dengan Random Forest. Prasad et al. [6,7] mencapai akurasi 99,24%, sementara Vjrobol et al. [9] mencapai 99,97%. Dengan demikian, penelitian ini menunjukkan bahwa penggunaan Random Forest dengan hyperparameter tuning efektif dalam meningkatkan kinerja model klasifikasi untuk dataset PhiUSIIL Phishing URL dan melampaui hasil penelitian sebelumnya.

6. DAFTAR PUSTAKA

[1]. Abdul Samad, S.R., Balasubaramanian, 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*.

[2]. Al-Ahmadi, S., & Alharbi, Y. (2020). A Deep Learning Technique for Web Phishing Detection Combined URL Features and Visual Similarity. *International journal of Computer Networks & Communications*, 12(5), 23-35. https://doi.org/10.5121/ijcnc.2020.12503.

[3]. Alani, M. M., & Tawfik, H. (2022). PhishNot: A Cloud-Based Machine-Learning Approach to Phishing URL Detection.

Computer Networks, 208, 109407. https://doi.org/10.1016/j.comnet.2022.109407 .

[4]. Blum, A., Wardman, B., Solorio, T., & Warner, G. (2010). Lexical feature based phishing URL detection using online learning. Dalam *Proceedings of the 3rd ACM workshop on Artificial intelligence and security* (hlm. 54-60). https://doi.org/10.1145/1866423.1866434.

[5]. Mangalam, K., & Subba, B. (2024). PhishDetect: A BiLSTM based phishing URL detection framework using FastText embeddings. Dalam *2024 16th International Conference on COMmunication Systems & NETworkS (COMSNETS)* (hlm. 230-235). https://doi.org/10.1109/comsnets59351.2024.10427067.

[6]. Prasad, A., & Chandra, S. (2024). PhiUSIIL Phishing URL (Website). UCI Machine Learning Repository. https://doi.org/10.1016/j.cose.2023.103545.

[7]. Prasad, A., & Chandra, S. (2024). PhiUSIIL: A diverse security profile empowered phishing URL detection framework based on similarity index and incremental learning. Computers & Security, 136, 103545. https://doi.org/10.1016/j.cose.2023.103545.

[8]. Jalil, S., Usman, M. & Fong, A. Highly accurate phishing URL detection based on machine learning. J Ambient Intell Human Comput 14, 9233–9251 (2023). https://doi.org/10.1007/s12652-022-04426-3

[9]. Tambe, Y. S. (2023). Phishing URL Detection Using Machine Learning. Journal of Advanced Research in Production and Industrial Engineering, 7(3), 185-195. https://doi.org/10.24321/2456.429x.202301

[9]. Vajrobol, V., Gupta, B. B., & Gaurav, A. (2024). Mutual information based logistic regression for phishing URL detection. Cyber Security and Applications, 2, 100044. https://doi.org/10.1016/j.csa.2024.100044.