

Penerapan Algoritma Random Forest Untuk Analisis Dan Prediksi Phishing E-Mail Menggunakan Machine Learning

Implementation Of Random Forest Algorithm For Analysis And Prediction Of Phishing E-Mail Using Machine Learning

Ilham Ramadhan¹, Budi Triandi²

^{1,2} Program Studi Informatika, Fakultas Teknik dan Ilmu Komputer, Universitas Potensi Utama
K.L. Yos Sudarso KM 6,5 No. 3A Tj.Mulia – Medan
Email : ir.ilhamofficial@gmail.com¹, Email : buditriandi@gmail.com²

ABSTRAK

Perkembangan teknologi informasi yang pesat membawa kemudahan dalam komunikasi digital, namun juga memunculkan ancaman keamanan siber, salah satunya serangan phishing melalui email. Metode deteksi tradisional seperti blacklist dan rule-based sudah tidak memadai menghadapi serangan phishing yang semakin canggih. Oleh karena itu, diperlukan pendekatan adaptif berbasis machine learning untuk mendeteksi phishing email secara otomatis dan akurat. Penelitian ini bertujuan menerapkan algoritma Random Forest untuk analisis dan prediksi phishing email. Dataset yang digunakan berisi berbagai fitur penting dari email, seperti struktur URL, penggunaan HTTPS, panjang alamat, kata-kata mencurigakan, dan metadata pengirim. Proses penelitian mencakup pengumpulan dan eksplorasi data, pembersihan dan praproses, rekayasa fitur, pemisahan data latih dan uji, pembangunan model Random Forest, serta evaluasi performa menggunakan metrik akurasi, presisi, recall, f1-score, dan ROC-AUC. Hasil pengujian menunjukkan bahwa model Random Forest mampu mengklasifikasikan email phishing dan non-phishing dengan tingkat akurasi di atas 90%. Model ini juga terbukti robust terhadap ketidakseimbangan data serta mampu menangani berbagai tipe fitur secara efektif. Dengan demikian, penerapan algoritma Random Forest dapat menjadi solusi yang menjanjikan dalam meningkatkan sistem deteksi phishing email yang adaptif, efisien, dan akurat.

Kata Kunci : *Phising E-mail, Random Forest, Machine Learning, Klasifikasi, Keamanan Siber.*

ABSTRACT

The rapid development of information technology has brought convenience to digital communication, but it has also given rise to cybersecurity threats, one of which is phishing attacks via email. Traditional detection methods such as blacklists and rule-based methods are no longer adequate to deal with increasingly sophisticated phishing attacks. Therefore, an adaptive, machine-learning-based approach is needed to detect phishing emails automatically and accurately. This research aims to apply the Random Forest algorithm to the analysis and prediction of phishing emails. The dataset used contains various important email features, such as URL structure, HTTPS usage, address length, suspicious words, and sender metadata. The research process included data collection and exploration, cleaning and preprocessing, feature engineering, separation of training and test data, building a Random Forest model, and performance evaluation using accuracy, precision, recall, f1-score, and ROC-AUC metrics. The test results show that the Random Forest model is capable of classifying phishing and non-phishing emails with an accuracy rate above 90%. This model also proved robust to data imbalance and was able to handle various feature types effectively. Thus, the application of the Random Forest algorithm can be a promising solution in improving the adaptive, efficient, and accurate email phishing detection system.

Kata Kunci : *Phising E-mail, Random Forest, Machine Learning, Classification, Cyber Security.*

1. PENDAHULUAN

Perkembangan teknologi informasi melaju pesat membawa banyak benefit kemudahan dalam kehidupan sehari-hari, terutama dalam komunikasi digital melalui *email*. Namun, seiring dengan meningkatnya penggunaan *email*, muncul pula berbagai ancaman siber yang mengintai pengguna, salah satunya adalah serangan *phishing*. *Phishing* merupakan bentuk serangan rekayasa sosial dengan mengajak korban untuk melakukan *click* dan melakukan kunjungan ke web yang seolah dikenal lalu diminta untuk memasukkan data yang bertujuan untuk memperoleh informasi pribadi atau sensitif,

seperti *username*, *password*, dan informasi keuangan, dengan menyamar sebagai entitas yang terpercaya [1].

Menurut laporan *Anti-Phishing Working Group* (APWG), total jumlah insiden *phishing* global terus meningkat. Sektor surat elektronik atau *e-mail* menjadi target paling mudah dan sering diserang, dengan persentase mencapai 30,5% dari seluruh serangan *phishing* pada kuartal ketiga tahun 2024 [2]. Efektivitas serangan *phishing* yang tinggi menunjukkan bahwa melakukan metode deteksi tradisional seperti *blacklist* atau *filter* berbasis aturan sudah tidak lagi efektif untuk menghadapi ancaman yang semakin canggih dan dinamis.

Dalam upaya meningkatkan sistem *cyber security* terhadap *phishing*, teknologi *machine learning* (ML) hadir sebagai salah satu solusi yang menjanjikan. *Machine learning* memiliki kemampuan untuk mengenali pola dan anomali dari data yang besar dan kompleks secara otomatis dari *dataset* yang diberikan [3]. Berbagai teknik telah dikembangkan untuk tujuan ini. Salah satu algoritma yang cukup efektif digunakan dalam klasifikasi dan prediksi adalah algoritma *Random Forest*.

Beberapa studi telah mengeksplorasi penerapan berbagai *algoritma machine learning* untuk mendeteksi serangan *phishing*. Sebuah studi komparatif menemukan bahwa *Random Forest* memberikan hasil yang sangat baik dalam hal akurasi, presisi, dan *recall*. Penelitian lain menunjukkan bahwa *Random Forest* memiliki akurasi deteksi yang signifikan dan mampu menangani data yang tidak seimbang. Sementara itu, algoritma lain seperti *Support Vector Machine* (SVM), *Naive Bayes*, dan *K-Nearest Neighbors* (KNN) juga telah diterapkan, namun *Random Forest* seringkali menunjukkan keunggulan dalam analisis dan prediksi *phishing email*.

Menurut [4], *Cyber Security* merupakan bidang keilmuan yang melibatkan perlindungan sistem komputer, jaringan, dan data dari ancaman dan serangan digital yang dapat merusak atau mencuri informasi. Dengan berkembangnya teknologi digital dan penggunaan internet secara global, ancaman terhadap keamanan data pribadi dan organisasi semakin meningkat. *Phishing*, sebagai salah satu metode penipuan paling umum di dunia maya, menargetkan individu atau organisasi untuk mencuri informasi sensitif. Analisis tekstual *email* adalah salah satu metode untuk mendeteksinya.

Menurut [5], *Phishing* adalah jenis penipuan *online* di mana penyerang menyamar sebagai entitas atau sosok yang dikenal tepercaya untuk memperoleh informasi pribadi korban, biasanya melalui *email* yang tampak sah meniru dengan format yang sangat detail. *Email phishing* umumnya berisi tautan yang mengarah ke arah situs palsu yang sudah dirancang untuk menipu pengguna agar memasukkan data sensitif mereka, seringkali dengan menganalisis URL.

Menurut [6], *Random Forest* adalah *algoritma* pembelajaran mesin yang termasuk dalam kategori *ensemble learning*, yang menggunakan banyak pohon keputusan (*decision trees*) untuk membuat prediksi lebih akurat. *Random Forest* membangun sejumlah pohon keputusan dari subset acak data pelatihan dan kemudian menggabungkan hasilnya untuk memperoleh prediksi final. Keunggulan *algoritma* ini adalah kemampuannya untuk mengatasi *overfitting* (sehingga lebih tahan terhadap data yang tidak sempurna) dan efektivitasnya dalam menangani data besar dan variabel yang kompleks.

Dalam upaya meningkatkan sistem keamanan terhadap *phishing*, teknologi *machine learning* (ML) hadir sebagai salah satu solusi yang menjanjikan. *Machine learning* memiliki kemampuan untuk mengenali pola dan anomali dari data yang besar dan kompleks secara otomatis. Berbagai teknik telah dikembangkan untuk tujuan ini. Salah satu algoritma yang cukup efektif digunakan dalam klasifikasi dan prediksi adalah algoritma *Random Forest*.

Penelitian yang dilakukan [7], dengan judul “*Detection of Phishing Emails using Machine Learning and Deep Learning*”. Memperoleh hasil analisis dengan menggunakan berbagai *algoritma machine learning* untuk mendeteksi *phishing email*. Mereka menemukan bahwa *Random Forest* memberikan hasil yang sangat baik dari segi akurasi, presisi, dan *recall*, menjadikannya pilihan yang sangat baik untuk analisis dan prediksi ancaman dari *phishing email*. *Random Forest* dapat mengenali pola yang lebih kompleks dalam *data phishing* dibandingkan dengan algoritma lain yang jauh lebih sederhana.

Hal ini dikarenakan *Random Forest* memiliki kemampuan untuk menangani data dengan ukuran dimensi yang besar dan jumlah variabel yang sangat beragam.

Penelitian yang dilakukan [8], dengan judul “*Phishing Detection System using Random Forest Algorithm*”. Dengan penelitian ini bertujuan untuk meningkatkan metode dari deteksi situs web *phishing* menggunakan teknologi *machine learning*. Penelitian yang dilakukan mencapai akurasi deteksi 97,14% dengan *algoritma random forest* memiliki tingkat *false positive* terendah. Selain itu, hasilnya menunjukkan bahwa pengklasifikasian menunjukkan kinerja yang lebih baik ketika menggunakan banyak data sebagai data pelatihan. Di masa depan, teknologi *hybrid* diterapkan untuk mendeteksi situs web *phishing* lebih akurat dan akan digunakan *algoritma random forest* dari teknologi pembelajaran mesin dan metode pemblokiran.

Penelitian yang dilakukan [9], dengan judul “*Performance Analysis of Support Vector Classification and Random Forest in Phishing Email Classification*”. Dengan penelitian ini menunjukkan akurasi yang meningkat signifikan untuk model *Random Forest* dan *support vector classification* di berbagai persentase pengolahan data. Secara spesifik, *vector classification* secara konsisten mengungguli model *Random Forest*, mencapai tingkat akurasi yang lebih tinggi. Pada persentase pembagian 70:30, *Support Vector Classification* mencapai akurasi tertinggi sebesar 97,52%, diikuti oleh 97,37% pada persentase pembagian 60:40 dengan ini menunjukkan bahwa *random forest* hanya memiliki selisih sedikit dari metode lainnya.

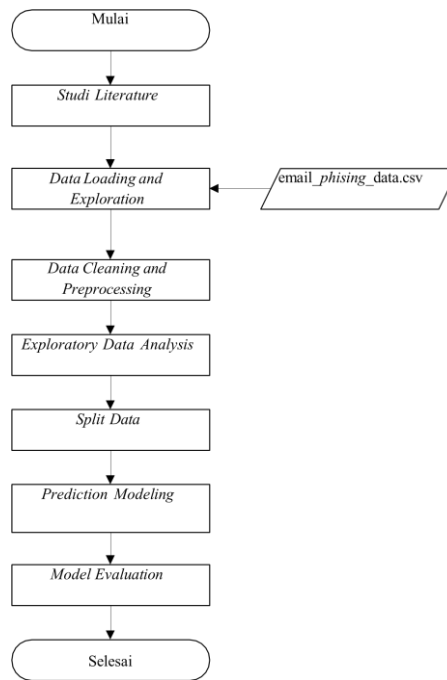
Penelitian yang dilakukan [10], dengan judul “*A Comparative Study of Machine Learning Techniques for Phishing Website Detection*”. Menemukan bahwa menguji beberapa *algoritma machine learning* berbasis fitur struktural website untuk mendeteksi situs *phishing* ditemukan bahwa *algoritma random forest* menunjukkan akurasi tertinggi di dibandingkan dengan beberapa algoritma seperti *Logistic Regression* dan *SVM*. *Algoritma* berbasis pohon (*Random Forest* dan *Decision Tree*) menunjukkan lebih efektif dalam hal menangani kompleksitas fitur dibandingkan metode linear sehingga *Random Forest* direkomendasikan sebagai model paling andal untuk deteksi *phishing website* karena mampu meminimalkan kesalahan klasifikasi dan memberikan hasil yang stabil.

Penelitian yang dilakukan [11], dengan judul “*Phishy? Detecting Phishing Emails Using Machine Learning and Natural Language Processing*”. Penelitian ini menunjukkan bahwa pendekatan berbasis NLP dan *machine learning* sangat efektif untuk mendeteksi *email phishing* dilakukan pengujian enam *algoritma ML* pada dua *dataset* publik dan ditemukan bahwa metode ini mampu mencapai akurasi, presisi, dan *recall* yang tinggi hasil yang didapatkan ini menegaskan bahwa kombinasi NLP dan ML dapat meningkatkan kemampuan sistem keamanan *email* untuk mengenali dan mencegah serangan *phishing*.

Berdasarkan latar belakang tersebut, penerapan *algoritma Random Forest* untuk analisis dan memprediksi *phishing email* melalui pendekatan *machine learning* menjadi penting untuk dikaji lebih lanjut untuk menganalisis dan mengidentifikasi fitur-fitur penting yang ada di dalam email yang digunakan sebagai indikator dari *phishing*. Membangun model klasifikasi *phishing email* menggunakan *algoritma Random Forest*, Mengukur dan mengevaluasi kinerja *algoritma Random Forest* dalam mendeteksi *email phishing*, menggunakan metrik seperti akurasi, presisi, *recall*, dan *F1-score* sehingga *phishing* dapat dideteksi kemudian dilakukan tindakan sehingga dapat melindungi dari celah serangan yang ditargetkan.

2. METODE

Metodologi penelitian ini dirancang secara sistematis untuk memastikan model yang dihasilkan memiliki performa yang optimal dalam mendeteksi email phishing. Tahapan penelitian diilustrasikan pada Gambar 1.



Gambar 1. Rancangan Penelitian Penerapan Algoritma Random Forest untuk Analisis dan Prediksi Phishing Email Menggunakan Machine Learning.

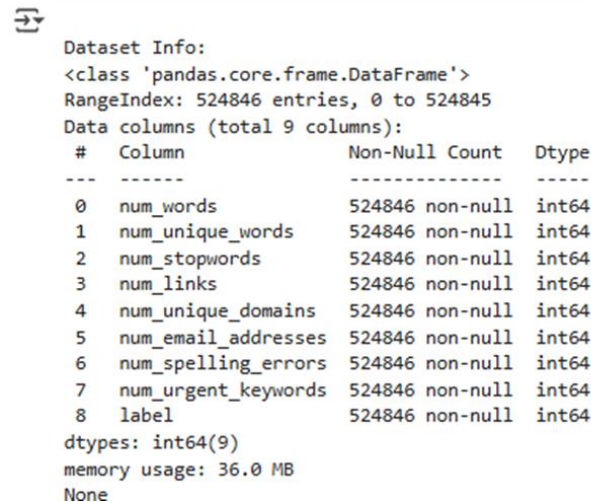
2.1. Bahan Penelitian (Dataset)

Bahan utama yang digunakan dalam penelitian ini adalah *dataset* cuaca yang bersumber dari platform online Kaggle dengan jumlah data awal 524.846. *Dataset* ini menggabungkan lebih dari ~500.000 email *non-phishing* ("aman") dari *Enron Email Dataset* dan ~20.000 email *phishing* dan aman dari *Phishing Email Dataset* [12].

	A	B	C	D	E	F	G	H	I
1	num_words	num_unique_words	num_stopwords	num_links	num_unique_domains	num_email_addresses	num_spelling_errors	num_urgent_keywords	label
2	140	94	52	0	0	0	0	0	0
3	5	5	1	0	0	0	0	0	0
4	34	32	15	0	0	0	0	0	0
5	6	6	2	0	0	0	0	0	0
6	9	9	2	0	0	0	0	0	0
7	37	29	5	0	0	3	7	1	0
8	4	4	1	0	0	0	0	0	0
9	22	21	4	4	1	0	7	0	0
10	289	176	66	0	0	3	28	2	0
11	32	30	9	0	0	0	2	0	0
12	50	32	19	0	0	0	2	0	0
13	923	355	309	0	0	0	51	0	0
14	923	355	309	0	0	0	51	0	0
15	19	17	7	0	0	0	0	0	0
16	5	5	1	0	0	0	0	0	0
17	130	75	12	0	0	0	39	0	0
18	25	24	8	0	0	0	1	0	0
19	275	159	94	0	0	4	21	0	0
20	93	56	15	0	0	0	16	0	0
21	26	24	10	0	0	0	0	0	0
22	13	17	4	0	0	0	1	0	0

Gambar 2. Dataset Email Phishing Data.

Setiap *email* telah melalui proses *feature extraction* berbasis *Natural Language Processing* (NLP). *Dataset* ini menyediakan 9 kolom fitur yang merepresentasikan karakteristik penting dari email, yaitu:



```

Dataset Info:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 524846 entries, 0 to 524845
Data columns (total 9 columns):
#   Column                Non-Null Count  Dtype
---  -
0   num_words              524846 non-null int64
1   num_unique_words       524846 non-null int64
2   num_stopwords          524846 non-null int64
3   num_links              524846 non-null int64
4   num_unique_domains     524846 non-null int64
5   num_email_addresses    524846 non-null int64
6   num_spelling_errors    524846 non-null int64
7   num_urgent_keywords    524846 non-null int64
8   label                  524846 non-null int64
dtypes: int64(9)
memory usage: 36.0 MB
None

```

Gambar 3. Informasi Dataset Email_Phishing_Data.

1. *num_words* : Jumlah total kata.
2. *num_unique_words* : Jumlah kata unik.
3. *num_stopwords* : Jumlah kata umum (cth: "the", "and").
4. *num_links* : Jumlah tautan (*hyperlink*).
5. *num_unique_domains* : Jumlah *domain* unik dalam tautan.
6. *num_email_addresses* : Jumlah alamat *email* dalam teks.
7. *num_spelling_errors* : Jumlah kata yang salah eja.
8. *num_urgent_keywords* : Jumlah kata-kata mendesak (cth: "urgent", "verify").
9. *label*: Variabel target (0 = *Email* aman, 1 = *Email phishing*).

2.2. Pengolahan Data

Proses pengolahan data mengikuti langkah-langkah yang diuraikan dalam rancangan penelitian.

1. Data Loading and Exploration

Data dimuat menggunakan pustaka Pandas pada *Python*. Eksplorasi awal dilakukan untuk memahami struktur *dataset*, tipe data, dan jumlah data.

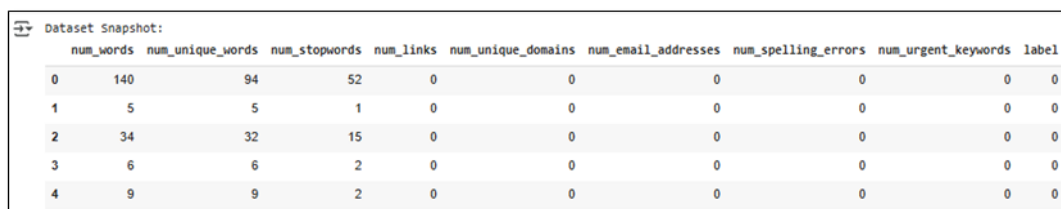


```

[8] # Load the dataset
data_path = '/content/email_phishing_data.csv'
df = pd.read_csv(data_path, encoding='ascii', delimiter=',')

```

Gambar 4. Memuat Dataset.



	num_words	num_unique_words	num_stopwords	num_links	num_unique_domains	num_email_addresses	num_spelling_errors	num_urgent_keywords	label
0	140	94	52	0	0	0	0	0	0
1	5	5	1	0	0	0	0	0	0
2	34	32	15	0	0	0	0	0	0
3	6	6	2	0	0	0	0	0	0
4	9	9	2	0	0	0	0	0	0

Gambar 5. Menampilkan Lima Baris Pertama Dari Dataset.

2. Data Cleaning and Preprocessing

Tahap ini krusial untuk memastikan kualitas data. Dilakukan pengecekan nilai hilang (*missing values*) dan data duplikat.

```
# Check for missing values
print('Missing values per column:')
display(df.isnull().sum())
```

Gambar 6. Pengecekan Nilai Yang Hilang.

```
# Check for duplicate rows
num_duplicates = df.duplicated().sum()
print(f'Number of duplicate rows: {num_duplicates}')
```

Gambar 7. Pengecekan Baris Yang Duplikat.

```
# If needed, drop duplicates
if num_duplicates > 0:
    df = df.drop_duplicates()
```

Gambar 8. Pengkondisian Jika Terdapat Duplikat.

```
Missing values per column:
0
num_words      0
num_unique_words 0
num_stopwords  0
num_links      0
num_unique_domains 0
num_email_addresses 0
num_spelling_errors 0
num_urgent_keywords 0
label          0

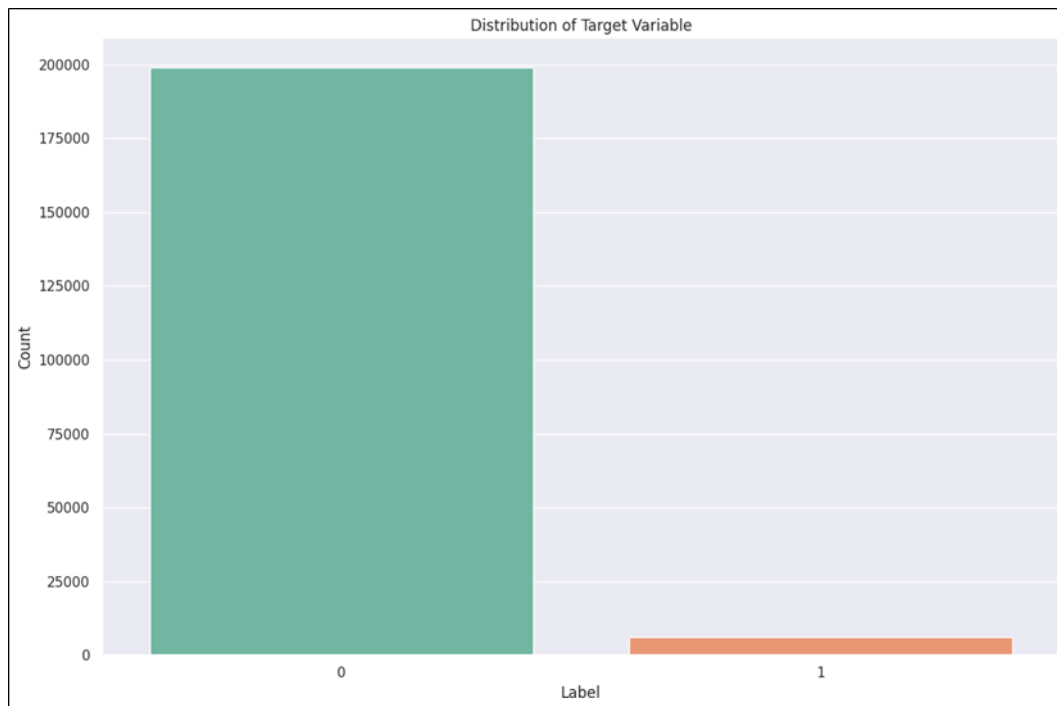
dtype: int64
Number of duplicate rows: 319794
Shape after cleaning: (205052, 9)
```

Gambar 9. Menampilkan Hasil Setelah Pembersihan.

Dari proses ini, ditemukan 319.794 baris duplikat. Setelah penghapusan, *dataset* bersih terdiri dari 205.052 baris data unik.

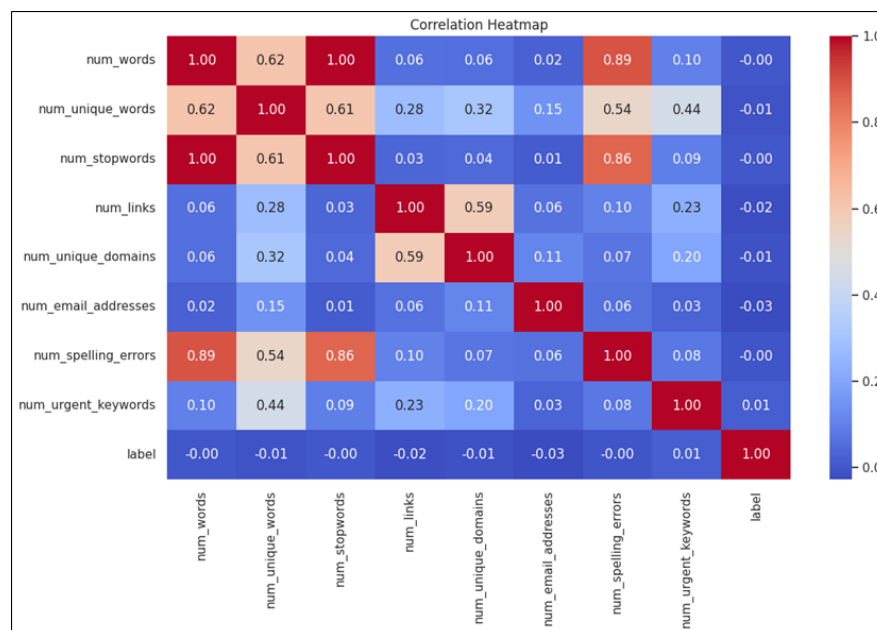
3. *Exploratory Data Analysis (EDA)*

EDA dilakukan untuk membantu memahami karakteristik dari pola data, distribusi, dan korelasi antar fitur [13]. Visualisasi distribusi variabel target (Gambar 10) menunjukkan adanya ketidakseimbangan kelas (*class imbalance*) yang signifikan, di mana jumlah *email* aman (Label 0) jauh lebih banyak daripada *email phishing* (Label 1).



Gambar 10. Grafik Distribusi Variabel Target 'Label'.

Selain itu, *heatmap* korelasi (Gambar 11) menunjukkan bahwa tidak ada fitur tunggal yang memiliki korelasi linear kuat dengan label, yang mengindikasikan perlunya model non-linear seperti *Random Forest* untuk menangkap pola yang kompleks.



Gambar 11. Coleration Heatmap.

4. Split Data

Dataset akan dibagi menjadi data latih (70%) dan data uji (30%) menggunakan *train_test_split* dari *Scikit-learn* alasan mengapa ini dilakukan agar dataset tidak mengalami *overfitting* [14]. Parameter *stratify=target* digunakan untuk memastikan proporsi kelas yang tidak seimbang tetap terjaga di kedua set data.

```
# Define feature columns and target
features = df.drop('label', axis=1)
target = df['label']
```

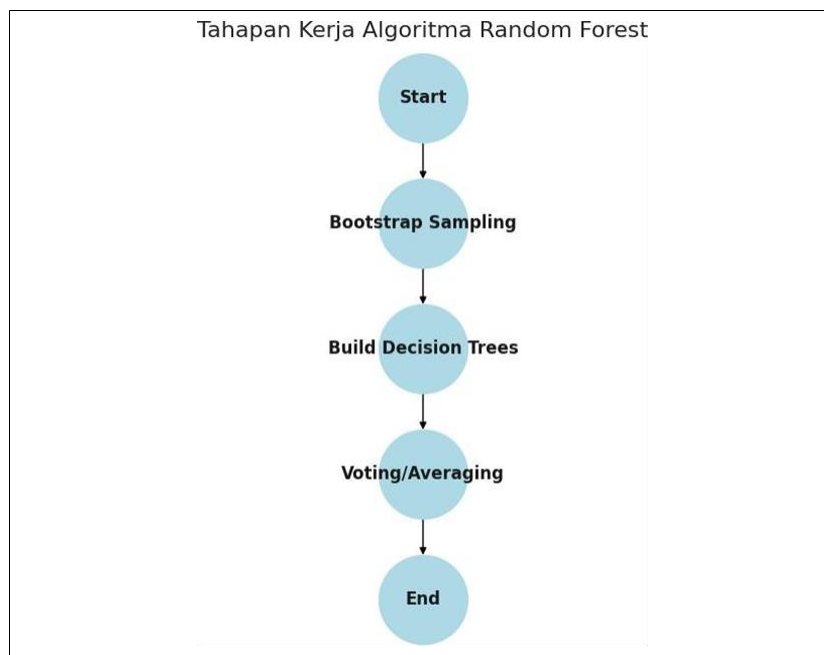
Gambar 12. Mendefenisikan Kolom Fitur.

```
# Split into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(features, target, test_size=0.3, random_state=RANDOM_STATE, stratify=target)
```

Gambar 13. Pembagian Data ke dalam Set Latih dan Uji.

5. Prediction Modeling

Model *Random Forest Classifier* diinisialisasi dengan 100 pohon ($n_estimators=100$) dan dilatih menggunakan data latih.



Gambar 14. Tahapan Kerja Dari Algoritma Random Forest.

6. Evaluation Model

Merupakan langkah penting dalam memahami kinerja model yang telah dibangun *Random Forest* digunakan untuk memprediksi dua kelas (*Class 0* dan *Class 1*) Proses evaluasi dilakukan melalui beberapa tahapan untuk mengukur seberapa baik dari kinerja model dalam melakukan prediksi yang akurat. Model dievaluasi menggunakan data uji dengan metrik standar untuk klasifikasi, digunakan *voting* mayoritas. Prediksi akhir adalah kelas yang paling sering diprediksi oleh semua pohon keputusan yaitu metrik evaluasi yang digunakan adalah Akurasi, Presisi, dan *Recall* [15].

$$\text{Akurasi} = \frac{\text{Jumlah Prediksi Benar}}{\text{Jumlah Total Data}}$$

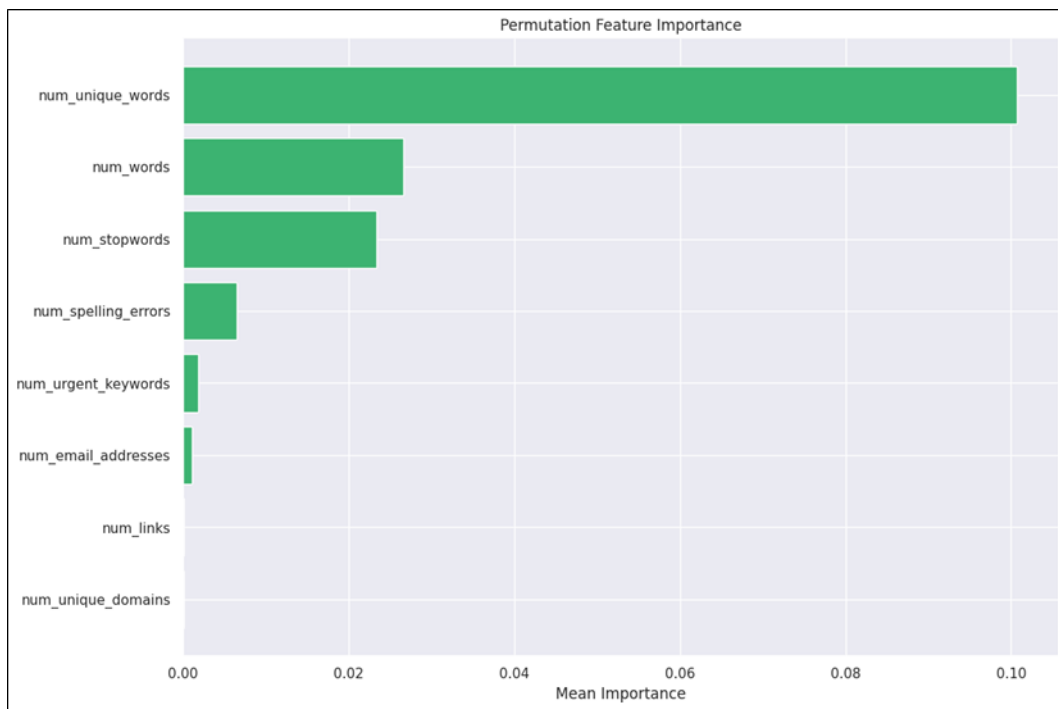
$$\text{Presisi} = \frac{\text{True Positive (TP)}}{\text{True Positive (TP)} + \text{False Positive (FP)}}$$

$$\text{Recall} = \frac{\text{True Positive (TP)}}{\text{True Positive (TP)} + \text{False Negative (FN)}}$$

Dimana TP = *True Positive*, FP = *False Positive*, dan FN = *False Negative*.

3. HASIL DAN PEMBAHASAN

Setelah model dilatih, kami menganalisis fitur mana yang paling berkontribusi terhadap keputusan model menggunakan *Permutation Feature Importance*. Hasilnya disajikan pada Gambar 15.



Gambar 15. *Permutation Feature Importance*.

Berdasarkan grafik, fitur *num_unique_words* (jumlah kata unik) adalah prediktor paling penting, dengan nilai *mean importance* jauh di atas 0.10. Ini menunjukkan bahwa variasi kosakata dalam *email* adalah indikator kuat untuk membedakan *phishing* dari *email* aman. Fitur *num_words* (jumlah total kata) dan *num_stopwords* (jumlah kata umum) juga memiliki kontribusi signifikan. Menariknya, fitur seperti *num_links* dan *num_unique_domains* memiliki pengaruh yang sangat kecil dalam model ini, meskipun secara intuitif sering dianggap sebagai indikator *phishing*.

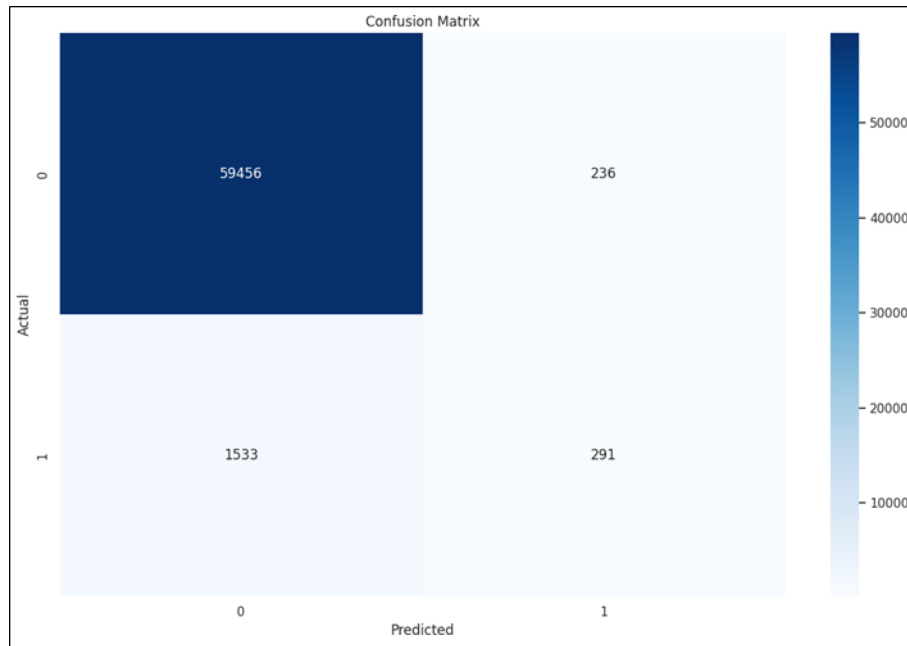
3.1. Evaluasi Kinerja Model

Kinerja model dievaluasi secara kuantitatif menggunakan data uji.

Classification Report:				
	precision	recall	f1-score	support
Class 0	0.97	1.00	0.99	59692
Class 1	0.55	0.16	0.25	1824
accuracy			0.97	61516
macro avg	0.76	0.58	0.62	61516
weighted avg	0.96	0.97	0.96	61516

Gambar 16. Laporan Klasifikasi (*Classification Report*)

Akurasi Keseluruhan Model *Random Forest* mencapai akurasi prediksi keseluruhan yang sangat tinggi Akurasi Prediksi: 0.9712 (97.12%). Sedangkan untuk *confusion matrix* akan di perlihatkan gambar 17 dibawah ini.



Gambar 17. Confusion Matrix.

Dari *confusion matrix* di atas di temukan data sebagai berikut:

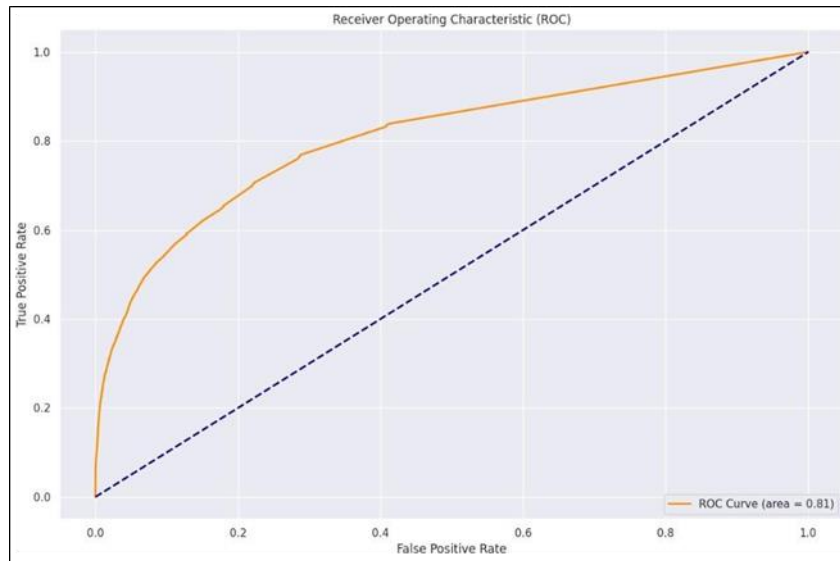
1. *True Negative (TN)*: 59.456 (*Email aman diprediksi aman*).
2. *False Positive (FP)*: 236 (*Email aman diprediksi phishing*).
3. *False Negative (FN)*: 1.533 (*Email phishing diprediksi aman*).
4. *True Positive (TP)*: 291 (*Email phishing diprediksi phishing*).

Dari laporan ini, terlihat sebuah temuan penting:

1. Kelas 0 (*Email Aman*): Model memiliki kinerja nyaris sempurna dengan *precision* 0.97, *recall* 1.00, dan *f1-score* 0.99. Ini berarti model sangat andal dalam mengidentifikasi *email* yang aman dan hampir tidak pernah salah mengklasifikasikannya sebagai *phishing* (ditunjukkan oleh nilai FP yang rendah, 236, dibandingkan TN 59.456).
2. Kelas 1 (*Email Phishing*): Kinerja model untuk kelas ini jauh lebih rendah. *Precision* 0.55 menunjukkan bahwa ketika model memprediksi *phishing*, prediksinya benar sekitar 55% kasus. Namun, *recall* sangat rendah (0.16), yang berarti model hanya berhasil mendeteksi 16% dari semua *email phishing* yang sebenarnya (ditunjukkan oleh nilai FN yang tinggi, 1.533).

Pembahasan dari hasil ini adalah bahwa akurasi 97% yang tinggi sebagian besar didorong oleh kemampuan model memprediksi kelas mayoritas (*Email Aman*) dengan benar, yang disebabkan oleh ketidakseimbangan data yang ekstrem. Model ini sangat baik dalam tidak menandai *email* aman sebagai *phishing*, tetapi buruk dalam menemukan *email phishing* disebabkan ketidakseimbangan data ekstrem namun kemampuan model ini masih sangat baik dengan mempertimbangkan data *email* aman dengan ini model memiliki pondasi yang bagus di karenakan terkenal dalam menangani *dataset* kompleks dan tetap stabil meskipun masih terdapat *noise* atau *outlier*.

Untuk *ROC Curve (Receiver Operating Characteristic)* evaluasi terakhir menggunakan kurva *ROC* untuk mengukur kemampuan model dalam membedakan kedua kelas.



Gambar 18. ROC Curve (Receiver Operating Characteristic).

Model ini menghasilkan *Area Under the Curve (AUC)* sebesar 0.81. Nilai *AUC* di atas 0.8 menunjukkan kemampuan diskriminatif yang "baik" hingga "sangat baik". Ini menegaskan bahwa meskipun *recall* untuk kelas 1 rendah pada *threshold* standar, model *Random Forest* secara fundamental memiliki pemahaman yang kuat tentang fitur-fitur yang membedakan kedua kelas.

4. KESIMPULAN

Penelitian ini menunjukkan bahwa model *Random Forest* memiliki performa yang sangat baik dalam mendeteksi *email* aman (*Class 0*), dengan akurasi keseluruhan mencapai 97.12%. Hal ini tercermin dari *precision* dan *recall* yang sangat tinggi untuk *Class 0*, menghasilkan *f1-score* sebesar 0.99.

Analisis *Receiver Operating Characteristic (ROC)* memperkuat evaluasi ini, dengan nilai *Area Under the Curve (AUC)* sebesar 0.81. Nilai *AUC* ini menunjukkan bahwa model memiliki kemampuan prediktif yang kuat dalam membedakan *email phishing* dari *email* aman, bahkan di bawah kondisi ketidakseimbangan kelas yang signifikan. Fitur *num_unique_words* (jumlah kata unik) teridentifikasi sebagai prediktor paling penting.

Namun, model masih menunjukkan kelemahan signifikan dalam mendeteksi *email phishing* (*Class 1*), dengan *f1-score* hanya 0.25. Performa ini terhambat oleh *recall* yang rendah (0.16), yang terutama disebabkan oleh distribusi data yang sangat tidak seimbang, di mana *email phishing* hanya mencakup sebagian kecil dari dataset.

Secara keseluruhan, penelitian ini berhasil mengidentifikasi kekuatan model *Random Forest* dalam mengklasifikasikan *email* aman dengan sangat baik, namun pengembangan lebih lanjut diperlukan untuk meningkatkan kemampuan model dalam menangani deteksi kelas minoritas (*phishing*).

5. SARAN

Untuk meningkatkan performa model dalam mendeteksi *email phishing* (*Class 1*), beberapa langkah strategis dapat diambil:

1. Mengatasi Ketidakseimbangan Data: Menerapkan teknik resampling seperti oversampling pada kelas minoritas (cth: *Synthetic Minority Oversampling Technique / SMOTE*) atau undersampling pada kelas mayoritas.
2. Optimasi *Threshold*: Melakukan penyesuaian pada *threshold* klasifikasi untuk meningkatkan *recall* pada *email phishing*, meskipun ini mungkin sedikit mengurangi *precision*.
3. Rekayasa Fitur (*Feature Engineering*): Menambahkan fitur yang lebih relevan, seperti analisis semantik yang lebih mendalam pada isi teks, *metadata header email*, atau analisis pola tautan yang lebih canggih.

4. Penggunaan *Algoritma* Lain: Mengeksplorasi *algoritma* yang dirancang khusus untuk menangani ketidakseimbangan data, seperti *Balanced Random Forest* atau *metode boosting* (cth: *XGBoost*) dengan pembobotan kelas.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih yang sebesar-besarnya kepada Dr. Lili Tanti, M. Kom, selaku dosen pembimbing, atas arahan, bimbingan, dan masukan yang konstruktif selama proses penyusunan dan penyelesaian penelitian ini. Ucapan terima kasih juga ditujukan kepada seluruh jajaran dosen di Program Studi Informatika, Fakultas Teknik dan Ilmu Komputer, Universitas Potensi Utama, atas ilmu dan dukungan akademis yang telah diberikan.

DAFTAR PUSTAKA

- [1] F. Nthurima, A. Mutua, and W. Stephen Titus, "Detecting Phishing Emails Using Random Forest and AdaBoost Classifier Model," *Open Journal for Information Technology*, vol. 6, no. 2, pp. 123–136, Nov. 2023, doi: 10.32591/coas.ojit.0602.03123n.
- [2] APWG(Anti-Phishing Working Group), "PHISHING ACTIVITY TRENDS REPORT," 2024. Accessed: Oct. 24, 2025. [Online]. Available: https://docs.apwg.org/reports/apwg_trends_report_q4_2024.pdf
- [3] A. H. Salem, S. M. Azzam, O. E. Emam, and A. A. Abohany, "Advancing cybersecurity: a comprehensive review of AI-driven detection techniques," *J Big Data*, vol. 11, no. 1, p. 105, Aug. 2024, doi: 10.1186/s40537-024-00957-y.
- [4] S. Kavya and D. Sumathi, "Staying ahead of phishers: a review of recent advances and emerging methodologies in phishing detection," *Artif Intell Rev*, vol. 58, no. 2, p. 50, Dec. 2024, doi: 10.1007/s10462-024-11055-z.
- [5] M. A. Tamal, M. K. Islam, T. Bhuiyan, A. Sattar, and N. U. Prince, "Unveiling suspicious phishing attacks: enhancing detection with an optimal feature vectorization algorithm and supervised machine learning," *Front Comput Sci*, vol. 6, Jul. 2024, doi: 10.3389/fcomp.2024.1428013.
- [6] K. Owa and O. Adewole, "Benchmarking Machine Learning Techniques for Phishing Detection and Secure URL Classification," *International Journal of Computer Science and Mobile Computing*, vol. 14, no. 1, pp. 20–37, Jan. 2025, doi: 10.47760/ijcsmc.2025.v14i01.003.
- [7] L. Shalini, S. S. Manvi, N. C. Gowda, and K. N. Manasa, "Detection of Phishing Emails using Machine Learning and Deep Learning," in *2022 7th International Conference on Communication and Electronics Systems (ICCES)*, IEEE, Jun. 2022, pp. 1237–1243. doi: 10.1109/ICCES54183.2022.9835846.
- [8] G. Ramesh, R. B. Lokitha, R. R. Monisha, and N. S. Neha, "Phishing Detection System using Random Forest Algorithm," 2023. [Online]. Available: www.ijrti.org
- [9] C. Umam, L. B. Handoko, and F. O. Isinkaye, "Performance Analysis of Support Vector Classification and Random Forest in Phishing Email Classification," *Scientific Journal of Informatics*, vol. 11, no. 2, pp. 367–374, May 2024, doi: 10.15294/sji.v11i2.3301.
- [10] M. F. Khan, R. K. Tiwari, S. K. Saroj, and T. Tripathi, "A Comparative Study of Machine Learning Techniques for Phishing Website Detection," 2023, pp. 97–109. doi: 10.1007/978-3-031-15542-0_6.
- [11] Md. F. Rabbi, A. I. Champa, and M. F. Zibran, "Phishy? Detecting Phishing Emails Using Machine Learning and Natural Language Processing," 2024, pp. 119–137. doi: 10.1007/978-3-031-55174-1_9.
- [12] Sanskritu., "Phishing Detection on Enron Dataset," Kaggle. Accessed: Oct. 27, 2025. [Online]. Available: <https://www.kaggle.com/code/sanskritu/phishing-detection-on-enron-dataset>
- [13] S. Dhummad, "The Imperative of Exploratory Data Analysis in Machine Learning," *Scholars Journal of Engineering and Technology*, vol. 13, no. 01, pp. 30–44, Jan. 2025, doi: 10.36347/sjet.2025.v13i01.005.
- [14] V. R. Joseph and A. Vakayil, "SPlit: An Optimal Method for Data Splitting," *Technometrics*, vol. 64, no. 2, pp. 166–176, Apr. 2022, doi: 10.1080/00401706.2021.1921037.

-
- [15] W. N. Ismail, M. M. Hassan, H. A. Alsalamah, and G. Fortino, "CNN-Based Health Model for Regular Health Factors Analysis in Internet-of-Medical Things Environment," *IEEE Access*, vol. 8, pp. 52541–52549, 2020, doi: 10.1109/ACCESS.2020.2980938.