**Application of Multiple Machine Learning Method (Hybrid Model) to Detect Phishing Links as Preventive Measure in Minimizing Victims of Data Theft**

Mokhamad Fikri Alfawaid

Sekolah Tinggi Ilmu Kepolisian

Jakarta 12160, Indonesia

E-mail: alfawaidfikri@gmail.com

***Abstract***

*The hegemony resulting from the widespread use of technology and social media cannot be stopped. On the other hand, the negative effects of an interconnected world are always present in the minds of all users. As with technology, crime evolves over time, creating new spaces that were previously uninhabited. Phishing, one of its forms, has become a terrifying spectre in the digital world because it exploits the greatest vulnerability in cyberspace: humans. This journal aims to present the results of the author's research into developing a machine learning (ML) model to assist in detecting and recognizing phishing links, while also proposing the use of ML by the National Police to assist citizens in avoiding phishing crimes. The research was conducted using the ML approach with three (three) methods, namely Support Vector Machine (SVM), Decision Tree, and Random Forest separately and their combination (hybrid model) in four (four) assessment areas, namely the level of accuracy, precision, recall, and scores f1.*

**Keyword: Phishing, Machine Learning, Support Vector Machine (SVM), Decision Tree, Random Forest, Hybrid Model.**

**Pendahuluan**

Phishing merupakan salah satu bentuk serangan siber yang paling sering digunakan oleh para pelaku kejahatan siber. Phishing sudah menjadi sebuah masalah yang berkepanjangan di dalam ruang maya sejak pertama kali serangan tersebut terjadi dan menyerang sebuah web portal America Online (AOL) sebagai pada tahun 1995 (James, 2006, dalam Chiew, Yong & Tan, 2018). Metode yang pertama digunakan adalah dengan berusaha mencuri data password penggunanya dan men-generate nomor kartu kredit secara random menggunakan algoritma tertentu. Kemudian akun tersebut digunakan untuk mengirimkan email phishing secara acak ke beberapa pengguna lain untuk mendapatkan data sensitif milik mereka. Pada akhirnya AOL berhasil menggagalkan serangan tersebut dengan mengaplikasikan langkah-langkah keamanan untuk mencegah metode yang sama terjadi.

Berdasarkan laporan investigasi kebocoran data Verizon, serangan phishing masih menjadi jenis serangan yang paling sering digunakan dan berhasil dengan lebih dari 80% laporan insiden keamanan dan kebocoran data (Sumber: Verizon 2021 Data Breach Investigations Report). Serangan phishing juga merupakan jenis serangan siber yang mengalami peningkatan pada tahun 2020 sebanyak 65% dibandingkan dengan tahun sebelumnya berdasarkan laporan Anti-Phishing Working Group (APWG, 2021). Selain dari jumlah serangan yang terus meningkat, keadaan semakin diperparah dengan fakta bahwa serangan phishing tidak hanya memiliki dampak pada kerugian materiil namun juga non-materiil. Salah satu contoh kerugian materiil yang ditimbulkan oleh serangan phishing diantaranya laporan IBM mengenai Cost of Data Breach tahun 2022 (IBM, 2022) dimana, phishing memiliki proporsi jumlah serangan terbanyak kedua sebesar 16% dengan nominal kerugian $ 4.91 juta sepanjang tahun 2022. Sedangkan dampak non-materiilnya yaitu terjadinya 3.7 juta kebocoran data perbulan sejak Agustus 2020 hingga Juli 2021 di Amerika Serikat dengan 58.8% diantaranya terjadi pada penyedia jasa layanan kesehatan di Amerika Serikat (HIPAA, 2021).

**Pernyataan Masalah**

Selama masa pandemi COVID-19 telah terjadi peningkatan tajam sebanyak 4-5 kali lipat terhadap tindak kejahatan siber melalui serangan phishing hal tersebut berdasarkan pada laporan transparansi Google (Google Transparency Report, 2021, dalam Al Ghifari, Hananto & Wahyono, 2022). Berbagai situs website mulai dari e-commerce, sosial media hingga situs perbankan tidak luput dari serangan phishing. Dalam melakukan serangan phishing, penyerang mencoba untuk memperoleh informasi sensitif dari korban, seperti username, password, nomor kartu kredit, atau informasi keuangan penting lainnya dengan memperdaya korbannya menggunakan link yang menipu. Link phising tersebut pada umumnya dikirim melalui email, pesan instan, atau media sosial.

Phishing tidak hanya digunakan sebagai bentuk serangan utama seorang pelaku kejahatan dunia maya dalam melakukan serangan. Tetapi phishing juga dapat dilakukan sebagai serangan pembuka untuk mengirimkan *payload* seperti malware baik itu *ransomware* maupun *spyware*. Selain itu, phishing juga kerap digunakan untuk digunakan sebuah *entry point* untuk memudahkan melakukan serangan pada target yang sebenarnya.

Sudah terdapat berbagai metode untuk menangani permasalahan terkait phishing tersebut, salah satu diantaranya adalah metode yang terbilang masih tradisional untuk mendeteksi link phising yaitu menggunakan metode blacklist dan whitelist untuk mengidentifikasi tautan yang aman atau tidak aman. Namun sayangnya, metode ini memiliki banyak keterbatasan dalam mendeteksi tautan phising yang baru dan seringkali tidak efektif dalam mengenali tautan phising yang tersembunyi di balik URL yang valid. Sehingga, dalam beberapa tahun terakhir, penggunaan ML hadir sebagai solusi untuk mendeteksi tautan phising secara akurat dan otomatis.

Dalam penelitian ini, algoritma ML dilatih untuk mengenali pola dan karakteristik dari tautan phising sehingga dapat mengklasifikasikan tautan yang mencurigakan. Dimana, penulis menerapkan 3 algoritma ML yaitu Support Vector Machine, Decision Tree, dan Random Forest yang kemudian dikolaborasikan untuk menciptakan sebuah algoritma hybrid dalam mendeteksi tautan phishing.

**Maksud & Tujuan Penelitian**

Penelitian ini ditujukan untuk meneliti dan menciptakan sebuah metode baru menggunakan penggabungan hasil dari 3 algoritma ML yang lebih efektif dan efisien dalam mendeteksi tautan palsu yang kerap digunakan dalam serangan phishing. Selain itu, penelitian tentang pengaplikasian ML untuk alat deteksi tautan phising ini diharapkan dapat memberikan kontribusi besar dalam upaya pencegahan kejahatan siber dan perlindungan informasi sensitif baik bagi individu maupun organisasi yang dapat digunakan dan diaplikasikan pada institusi Polri.

**Cakupan & Batasan Penelitian**

Penelitian ini dilakukan menggunakan bahasa pemrograman Python untuk membuat algoritma ML dengan menggunakan 1.677 *dataset* yang berisi 827 tautan *phishing* dan 850 tautan *non-phishing* untuk melatih model ML yang dibuat dengan rasio 80:20 untuk data *training* dan *test*. Dataset tersebut penulis dapatkan dari database milik Phishtank (Phishtank, 2021). Sebelum melatih model ML, penulis terlebih dahulu melakukan ekstraksi fitur dari masing-masing dataset dalam sebuah ekosistem *sandbox* pada OS Linux dengan menggunakan jaringan TOR. Selanjutnya, sebuah pilot software yang berbasis ekstensi Google Chrome dapat dibuat untuk menguji pengaplikasian model ML yang dibuat dalam kasus nyata.

Pada awal penelitian, penulis menggunakan 10.000 dataset phishing dan non-phishing URL. Namun, dalam proses ekstraksi fitur pertama kali terjadi beberapa kendala diantaranya lama waktu yang diperlukan untuk memproses ekstraksi fitur yang tidak kunjung berakhir, dimana hal tersebut disebabkan oleh beberapa faktor seperti halaman web dari tautan yang sudah tidak aktif dan sistem keamanan yang cukup sensitif. Sehingga, hasil ekstraksi fitur pada percobaan pertama tidak begitu kuat untuk mengidentifikasi link phising dan non-phishing. Oleh sebab itu, penulis menetapkan *time-limit* pada percobaan yang kedua dan berhasil menghasilkan 1.677 dataset.

**Metodologi Penelitian**

Penelitian ini terbagi atas beberapa bagian dalam prosesnya, diantaranya: Pertama, melakukan pendalaman latar belakang termasuk di dalamnya studi literatur dengan membaca dan mempelajari beberapa penelitian terdahulu pada area penelitian yang sama serta menentukan teknologi yang akan digunakan. Kedua, proses pengumpulan dataset berupa link phising dan non-phishing. Ketiga, melakukan data pre-processing dengan metode ekstraksi fitur. Keempat, menyiapkan dan melatih 3 model ML (SVM, Decision Tree, dan Random Forest), Kelima, menyiapkan algoritma penggabungan 3 model ML untuk menciptakan model hybrid. Keenam, pengujian masing-masing model baik model yang berdiri sendiri maupun model hybrid berdasarkan 4 area penilaian (akurasi, presisi, recall dan f1-score). Ketujuh, analisis dan evaluasi hasil penelitian terhadap keempat model ML tersebut.



**Survei Latar Belakang dan Studi Literatur**

**Machine Learning Untuk Deteksi Phishing URL**

Machine Learning adalah cabang ilmu komputer yang mengeksplorasi algoritma dan strategi untuk mengotomatiskan solusi dari masalah rumit yang sulit diprogram menggunakan teknik pemrograman standar (Rebala, Ravi and Churiwala, 2019). Beberapa literatur tentang penelitian terdahulu pada pemanfaatan ML dalam mendeteksi phishing telah diteliti dan dipelajari, diantaranya:

* Subasi, Molah, Almkallawi and Chaudhery (2017) meneliti performa beberapa model ML diantaranya ANN, KNN, SVM, Random Forest, dan C4.5 dalam mendeteksi phishing. Mereka berhasil memproduksi angka akurasi yang cukup tinggi dalam mendeteksi tautan phishing; model Random Forest adalah model yang paling akurat dengan tingkat akurasi 97.26%. Dataset yang digunakan dalam penelitiannya didapatkan dari University of California's UCI repository dengan 30 fitur. Penelitian ini menguji performa masing-masing model ML berdasarkan akurasi, f-measure and Receiver Operating Characteristics (ROC).
* Pandey, Gill, Sai Prasad Nadendla and Thaseen (2019) dalam penelitiannya menggabungkan model SVM dan Random Forest untuk menciptakan sebuah model hybrid. Model tersebut kemudian dilatih dan diuji pada 1.353 dataset URL dari University of California, Irvine's ML Repository. Model Random Forest merupakan model yang paling akurat diantara kedua model tersebut (RF and SVM). Namun, jika dibandingkan antara model hybrid yang diciptakan dengan masing-masing model secara berdiri sendiri, model hybrid menunjukkan signifikansi dalam hal keakuratan. Selain akurasi tes yang dilakukan juga meliputi f1, precision, correlation coefficient, true positive rate and true negative rate.

Kemudian, dengan mempertimbangkan beberapa penelitian tersebut, penulis menetapkan untuk menggunakan 3 model ML dalam penelitian ini. Berikut adalah 3 model yang dipilih beserta alasan penggunaannya:

|  |  |  |
| --- | --- | --- |
|  | **ML Model** | **Alasan** |
| 1 | Support Vector Machine (SVM) | Merupakan sebuag model *supervised learning* yang umum digunakan untuk kategorisasi, terkenal karena efisiensinya dalam mengkategorikan bidang dimensi yang beragam. |
| 2 | Decision Tree | Sebuah model ML yang banyak digunakan secara luas, model ini menggunakan struktur seperti pohon untuk mendeskripsikan peristiwa dan potensi konsekuensinya dalam berbagai konteks. |
| 3 | Random Forest | Model *ensemble learning* untuk klasifikasi, regresi dan masalah lainnya, dengan *output* berupa serangkaian pilihan acak untuk mewakili kelas mayoritas. |

**Dataset dan Data Pre-Processing**

Penulis memperoleh himpunan data URL phishing dan non-phising dari repositori dari Phishtank1 dan University of New Brunswick2. Penulis menggunakan 19 fitur untuk ekstraksi fitur yang telah dibuat oleh Lakshmi dan Vijaya (2021) dan Mohammad, Thabtah dan McCluskey (2012). Berikut adalah diantaranya:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Fitur Pada Address Bar** | **Indikator** | **Kesimpulan** |
| 1 | Menggunakan IP sebagai alamat URL | Jika ditemukan | Phishing |
| 2 | Menggunakan symbol ‘@’ pada alamat URL | Jika ditemukan | Phishing |
| 3 | Panjang URL | Lebih dari 54 karakter | Phishing |
| 4 | Kedalaman URL | Hitung simbol ‘/’ | - |
| 5 | URL redirection | Bila ditemukan posisi simbol “//” diluar dari urutan ke 6 dan 7 | Phishing |
| 6 | HTTPS palsu | Jika ditemukan | Phishing |
| 7 | Menggunakan URL shortening | Jika ditemukan | Phishing |
| 8 | Terdapat prefiks maupun sufiks symbol ‘-‘ | Jika ditemukan | Phishing |
| 9 | Jumlah subdomain | > 2  | Phishing |
| 10 | Jumlah parameter yang digunakan | Hitung jumlah query dalam URL | - |
| 11 | Jumlah period pada URL | Hitung jumlah simbol ‘.’ | - |
|  |  |  |  |
|  | **Fitur pada Domain** |  |  |
| 12 | Lakukan pengecekan DNS | Jika tidak ditemukan | Phishing |
| 13 | Cek web traffic (Alexa Rating) | Ranking > 100.000 | Phishing |
| 14 | Cek sertifikat SSL | Jika tidak ditemukan | Phishing |
| 15 | Cek umur domain | < 6 bulan | Phishing |
|  |  |  |  |
|  | **Fitur berbasis pada HTML dan Javascript** |  |  |
| 16 | Pengalihan Iframe | Jika ditemukan | Phishing |
| 17 | Kostumisasi bar status | Jika ditemukan | Phishing |
| 18 | Right-click dinonaktifkan | Jika ditemukan | Phishing |
| 19 | Website forwarding | Jika > 2 | Phishing |

Selanjutnya, untuk mengklasifikasikannya maka data yang sudah diekstrak akan diberikan kode 1 untuk fitur yang dicurigai merupakan sebuah phishing dan 0 untuk fitur non-phishing. Kemudian data yang sudah diekstrak penulis gunakan untuk melatih model ML.

**Metode Pengujian Model ML**

Pengujian model ML dilakukan dengan menguji performa berdasarkan 4 kriteria penilaian yaitu: akurasi, presisi, recall, dan f1-score. Dalam pengujian performa yang dilakukan penulis menitik beratkan pada didapatkannya performa model ML yang seimbang, oleh karena itu pengujian f1-score sangat dibutuhkan dalam pengujian ini (M.D. Hossain et al., 2020). Tabel dibawah menjelaskan masing-masing pengujian performa yang dilakukan beserta confusion matrix.

|  |  |  |
| --- | --- | --- |
|  | **Nilai diklasifikasikan positif** | **Nilai diklasifikasikan negatif** |
| Actual Positive | True Positive | False Negative |
| Actual Negative | False Positive | True Negative |

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Pengujian** | **Formula** | **Deskripsi** |
| 1 | Akurasi | *True Positive* + True Negative*nsamples* | Akurasi adalah jumlah TP dibandingkan dengan banyaknya data |
| 2 | Presisi | *True Positive**True Positive + False Positive* | Presisi dapat dinyatakan baik apabila FP memiliki pengaruh yang signnifikan |
| 3 | Recall | True PositiveTrue Positive + False Negative | Recall dikatakan baik apabila FN memiliki pengaruh yang signifikan |
| 4 | F1-Score | Presisi x Recall2 xPresisi + Recall | F1-Score diperlukan apabila model yang membutuhkan keseimbangan antara Presisi dan Recall |

**Teknologi yang Digunakan**

Figur di bawah menjelaskan mengenai perangkat lunak baik teknologi maupun environment yang penulis gunakan dalam penelitian ini.

**Hasil Pengujian dan Diskusi**

**Alur ML Deployment**

Proses penyusunan model ML dimulai dari pengumpulan dataset URL phishing dan non-phishing. Selanjutnya, penulis melakukan proses ekstraksi fitur terhadap dataset yang ada berdasarkan 19 fitur ada pada bagian data pre-processing. Data yang sudah diekstraksi kemudian dibersihkan untuk menghapus data dengan nilai null atau eror, setelah itu data dibagi menjadi 2 set untuk training dan testing dari model ML yang dibuat. Setelah data dibagi menjadi 2 set penulis melatih model ML yang dibuat menggunakan dataset untuk training serta men-deploy model ML tyang sudah dilatih. Figur di bawah mengilustrasikan flowchart dari proses pelatihan model ML, testing hingga deployment. Dalam proses ekstraksi fitur penulis menyiapkan sebuah program berbasis bahasa python untuk mempercepat prosesnya menggunakan fungsi looping yang terotomatisasi. Proses ekstraksi fitur merupakan bagian terpenting untuk menghasilkan kualitas model ML yang baik. Dikarenakan proses ekstraksi fitur memerlukan request dan perintah whois pada phishing URL maka proses tersebut penulis lakukan pada sebuah environment terisolasi di dalam sebuah sandbox. Penulis menggunakan Operating System (OS) Kali Linux dalam sebuah virtual box dengan menggunakan jaringan TOR dan Whonix gateway yang dikombinasikan dengan VPN pada OS utamanya.



**Performa Model ML**

Dalam proses pengujian performa model ML yang dibuat, pada proses data cleaning penulis menghapus beberapa fitur dari dataset untuk menghindari bias dari model ML untuk membedakan URL Phishing dan non-Phishing. Beberapa fitur yang dihapus dan alasan dihapusnya fitur rersebut diantaranya adalah sebagai berikut:

|  |  |  |
| --- | --- | --- |
|  | **Fitur yang Dihapus** | **Alasan** |
|  | Jumlah tag tersembunyi | Hampir separuh data mengalami nilai timeout |
|  | Jumlah hyperlink | Hampir separuh data mengalami nilai timeout |
|  | Jumlah iframe | Hampir separuh data mengalami nilai timeout |
|  | Script untuk menon-aktifkan *right-click* | Hampir tidak ada perbedaan antara URL phishing dan non-phishing |

Selanjutnya, dataset dibagi menjadi 2 dengan rasio 80:20 untuk training dan testing terhadap 3 model ML yang sebelumnya sudah disampaikan pada bab sebelumnya. Kemudian masing-masing model yang telah dilatih dievaluasi tingkat akurasi, presisi, recall dan skor f1 nya dengan hasil sebagai berikut:



Berdasarkan hasil pengujian tersebut, dapat dilihat bahwa model RF memiliki performa yang lebih baik dibandingkan dengan kedua model yang lain. Ketiga model ML tersebut selanjutnya dikombinasikan untuk menciptakan sebuah model ML baru, dan dilakukan pengujian yang sama terhadap model tersebut dengan hasil sebagai berikut:



Simulasi training model ML dan pengujian performa yang dilakukan dalam penelitian ini sudah diunggah secara online pada platform IDE Google Colab yang dapat dilihat dengan melakukan scan pada kode qr berikut atau dengan mengakses link berikut: <https://tinyurl.com/y5ywr299>.



**Kesimpulan dan Rekomendasi**

Secara umum, model ML yang diciptakan dalam penelitian ini menunjukan hasil yang baik dengan performa yang mumpuni. Model hybrid yang merupakan gabungan dari ketiga model ML yaitu Decision Tree, Random Forest dan Support Vector Machine (SVM) mampu menunjukan performa yang lebih unggul dibandingkan dengan masing-masing model berdiri sendiri. Skor yang didapatkan dari tes performa dari model hybrid yang dibuat diantaranya akurasi 94%, presisi 97% dan recall 96%. Model hybrid yang diciptakan juga mampu menunjukan skor f1 sebesar 94% yang mengindikasikan bahwa model tersebut memiliki keseimbangan antara presisi dan recallnya.

Berkaca pada penelitian yang telah dilakukan oleh penulis, maka penulis menyarankan para peneliti yang berniat untuk melakukan penelitian yang serupa untuk memperkaya jumlah dataset yang digunakan agar hasil klasifikasi dan performa ML yang dihasilkan lebbih baik. Selain itu dalam data pre-processing peneliti juga dapat mengimbuhkan beberapa teknik ekstraksi fitur yang lain untuk meningkatkan kualitas dari ekstraksi fitur pada masing-masing URL seperti penggunaan citra dari tampilan suatu situs web agar diketahui ciri suatu situs web phishing tidak hanya dari URL saja.