SKRIPSI

ANALISIS SERANGAN *PHISHING* PADA *EMAIL* DAN UPAYA MITIGASI MENGGUNAKAN METODE *SUPERVISED MACHINE LEARNING*

Sebagai salah satu syarat untuk menyelesaikan
Program Studi Sarjana Terapan Keamanan Sistem Informasi

Oleh :

SAL SABILLA AZZAHRA
6404221142

JURUSAN TEKNIK INFORMATIKA
POLITEKNIK NEGERI BENGKALIS
TAHUN 2026

HALAMAN PENGESAHAN

SKRIPSI

ANALISIS SERANGAN *PHISHING* PADA *EMAIL* DAN UPAYA MITIGASI MENGGUNAKAN METODE *SUPERVISED MACHINE LEARNING*

Oleh:

SAL SABILLA AZZAHRA

6404221142

Telah diujikan dan dinyatakan lulus ujian skripsi pada tanggal 26 Februari 2026 oleh tim penguji Program Studi Sarjana Terapan Keamanan Sistem Informasi

Bengkalis, 26 Februari 2026

Pembimbing

Kasmawi, M.Kom
NIP. 197706072014041001

Anggota Tim Penguji

Rezki Kurniati, M.Kom
NIP. 198306162018032001

Mansur, S.Kom., M.Kom
NIP. 198209192021211003

Pretti Ristra, S.Pd., M.Ed
NIP. 198710132022032004

Mengetahui
Ketua Jurusan Teknik Informatika

Kasmawi, M.Kom
NIP. 197706072014041001

HALAMAN PERNYATAAN KEASLIAN

Saya menyatakan dengan sesungguhnya bahwa Skripsi ini adalah asli hasil karya saya dan tidak terdapat karya yang pernah dilakukan untuk memperoleh gelar kesarjanaan di perguruan tinggi, dan sepanjang pengatahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau dipublikasikan oleh orang lain, kecuali yang secara tertulis disebutkan sumbernya dalam naskah dan dalam daftar pustaka.

Bengkalis, 26 Februari 2026

Sal Sabilla Azzahra

HALAMAN PERSEMBAHAN

Bismillahirrahmanirrahim
Alhamdulillahirabbil ‘alamin

Segala puji syukur penulis panjatkan ke hadirat Allah SWT atas segala rahmat, karunia, dan kekuatan yang diberikan sehingga penulis dapat menyelesaikan skripsi ini. Shalawat serta salam tidak lupa selalu tercurahkan kepada Nabi Muhammad SAW. Dengan segenap hati dan ketulusan serta rasa syukur, dan bahagia telah sampai pada titik ini, tentunya bukan suatu hal yang mudah, tetapi dengan niat, dukungan dan juga doa dari orang-orang baik disekitar penulis, pada akhirnya tugas akhir penulis terselesaikan dengan baik. Penulis persembahkan skripsi ini untuk :

Cinta pertama dan pintu surgaku, Bapak Husin dan Ibu Nurhayati. Terima kasih atas segala doa dan dukungan yang tidak pernah putus terhadap kakak. Memberikan cinta, kasih sayang, dan pengorbanan yang mengiringi setiap langkah untuk kakak menyelesaikan pendidikan ini. Terima kasih telah mengantarkan kakak sampai mendapatkan gelar sarjana ini walaupun beliau tidak sempat merasakan pendidikan bangku perkuliahan, namun mereka mampu melakukannya untuk anak Perempuan satu-satunya ini, semoga Allah SWT senantiasa menjaga kalian, beribu-ribu rasa cinta, kasih dan sayang untuk cinta pertamaku dan pintu surgaku.

1. Kepada abang saya satu-satunya, Muhammad Faizal terima kasih adik ucapkan, atas segala dukungan dan doa baik selama adik berkuliah, semoga selalu senantiasa menjadi abang yang baik untuk kedua adik nya, sehat dan selalu dalam lindungan Allah SWT beribu-ribu rasa cinta, kasih dan sayang untuk abangku
2. Kepada adik saya satu-satunya, Muhammad baihaqi hakim, terimakasih kakak ucapkan segala doa dan dukungan, setiap hal yang diusahakan untuk kakak Perempuan satu-satunya, dari hal kecil maupun besar adik dahulukan, terima kasih untuk setiap kata iya untuk kakak, terima kasih sudah menjadi

saudara laki-laki yang bertanggung jawab untuk saudara Perempuan nya, selalu menjadi tempat kakak mengadu kesedihan, sehat dan selalu dalam lindungan Allah SWT beribu-ribu rasa cinta, kasih dan sayang untuk adikku.

3. Kepada Kedua sahabat penulis yang sama-sama lagi berjuang mendapatkan gelar sarjana, yaitu Marina dan Putri Aisyah, terima kasih penulis ucapkan, mungkin kata terima kasih saja tidak cukup, karna beliau yang selalu ada disamping penulis baik senang maupun sedih, walaupun kita tidak sedarah tetapi rasanya melebihi itu, terima kasih sudah tulus berteman dengan penulis, penulis tidak akan lupa dengan segala kebaikan kalian berdua selama ini, beribu-ribu rasa cinta, kasih dan sayang penulis ucapkan buat kalian berdua
4. Kepada teman-teman seperjuangan yang tidak bisa penulis sebut satu persatu, tetapi segala kebaikan teman-teman akan selalu penulis ingat sampai kapan pun, terima kasih banyak selalu membantu dan mendukung disegala hal tentang penulis, sampai jumpa dilain waktu dan kesempatannya semoga kalian semua sehat selalu dan segala hal-hal baik dipermudahkan, Aamiin
5. Teruntuk diri sendiri, terima kasih telah bertahan sejauh ini, banyak air mata disetiap proses yang telah dilalui, tetap selalu meyakinkan diri sendiri agar selalu berusaha, kuat dan semangat menjalani proses ini sampai selesai

KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa atas segala rahmat dan anugerah-Nya, sehingga penulis dapat menyelesaikan penyusunan skripsi yang berjudul "Analisis Serangan *Phishing* Pada *Email* Dan Upaya Mitigasi Menggunakan Metode *Supervised Machine Learning*" dengan baik. Skripsi ini disusun sebagai salah satu syarat untuk menyelesaikan Program Studi Sarjana Terapan Keamanan Sistem Informasi, Jurusan Teknik Informatika, Politeknik Negeri Bengkalis.

Selama proses penyusunan skripsi ini, penulis menyadari bahwa tanpa adanya bantuan, bimbingan, arahan, serta dukungan dari berbagai pihak, penyusunan skripsi ini tidak dapat terselesaikan dengan baik. Oleh karena itu, pada kesempatan ini penulis mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Bapak Johny Custer, S.T., M.T selaku Direktur Politeknik Negeri Bengkalis
2. Bapak Kasmawi, S.Kom., M.Kom selaku Ketua Jurusan Teknik Informatika sekaligus pembimbing utama yang telah memberikan arahan, bimbingan, motivasi, serta meluangkan waktu selama proses penyusunan skripsi ini
3. Ibu Nurmi Hidayasari, S.T., M.Kom. selaku Ketua Program Studi Sarjana Terapan Keamanan Sistem Informasi
4. Seluruh dosen dan staf Jurusan Teknik Informatika Politeknik Negeri Bengkalis yang telah memberikan ilmu pengetahuan, pengalaman, serta pelayanan selama masa perkuliahan

ANALISIS SERANGAN *PHISHING* PADA *EMAIL* DAN UPAYA MITIGASI MENGGUNAKAN METODE *SUPERVISED MACHINE LEARNING*

Nama Mahasiswa	:	Sal Sabilla Azzahra
NIM	:	6404221142
Dosen Pembimbing	:	Kasmawi,S.Kom.,M.Kom

ABSTRAK

Perkembangan teknologi informasi menjadikan *email* sebagai media komunikasi utama, namun juga meningkatkan risiko serangan siber, khususnya *phishing*. Serangan *phishing* melalui *email* masih menjadi ancaman serius karena dapat menipu pengguna untuk memperoleh informasi sensitif seperti kredensial akun dan data pribadi, yang berpotensi menimbulkan kerugian finansial dan kebocoran data. Permasalahan utama dalam penelitian ini adalah rendahnya efektivitas metode deteksi *phishing* konvensional yang bersifat statis dan kurang adaptif terhadap perkembangan pola serangan. Penelitian ini bertujuan untuk menganalisis serangan *phishing* pada *email* serta menerapkan metode *supervised machine learning* sebagai upaya mitigasi dengan mengklasifikasikan *email* ke dalam kategori *phishing* dan *non-phishing*. Metode penelitian meliputi pengumpulan dataset *email* berlabel, *preprocessing* teks, ekstraksi fitur, pelatihan model *supervised machine learning*, serta evaluasi kinerja menggunakan metrik akurasi, *presisi*, *recall*, dan F1-*score*. Hasil penelitian menunjukkan bahwa metode *supervised machine learning* mampu mendeteksi dan mengklasifikasikan *email phishing* secara efektif dengan tingkat akurasi yang tinggi. Pendekatan ini dinilai adaptif dan andal sebagai dasar pengembangan sistem keamanan *email* serta berkontribusi dalam meningkatkan upaya mitigasi serangan *phishing*.

Kata Kunci : phishing, email, supervised machine learning

ANALISIS SERANGAN *PHISHING* PADA *EMAIL* DAN UPAYA MITIGASI MENGGUNAKAN METODE *SUPERVISED MACHINE LEARNING*

Student Name	:	Sal Sabilla Azzahra
NIM	:	6404221142
Supervisor	:	Kasmawi,S.Kom.,M.Kom

ABSTRACT

The development of information technology has made email the primary medium of communication, but it has also increased the risk of cyber attacks, particularly phishing. Phishing attacks via email remain a serious threat because they can trick users into providing sensitive information such as account credentials and personal data, which can potentially lead to financial losses and data leaks. The main problem in this study is the low effectiveness of conventional phishing detection methods, which are static and less adaptive to developments in attack patterns. This study aims to analyze phishing attacks on email and apply supervised machine learning methods as a mitigation effort by classifying emails into phishing and non-phishing categories. The research methods include collecting labeled email datasets, text preprocessing, feature extraction, supervised machine learning model training, and performance evaluation using accuracy, precision, recall, and F1-score metrics. The results show that the supervised machine learning method is capable of effectively detecting and classifying phishing emails with a high degree of accuracy. This approach is considered adaptive and reliable as a basis for developing email security systems and contributes to improving phishing attack mitigation efforts.

Keyword: phishing, email, supervised machine learning

DAFTAR ISI

HALAMAN PENGESAHAN i
HALAMAN PERNYATAAN KEASLIAN ii
HALAMAN PERSEMBAHAN iii
KATA PENGANTAR v
ABSTRAK vi
ABSTRACT vii
DAFTAR ISI viii
DAFTAR GAMBAR x
BAB I PENDAHULUAN 1
1.1 Latar Belakang 1
1.2 Permasalahan 3
1.3 Batasan Masalah 3
1.4 Tujuan Penelitian 3
1.5 Manfaat Penelitian 3
BAB II TUJUAN PUSTAKA 5
2.1 Kajian Terdahulu 5
2.2 Teori Penunjang 11
2.2.1 Phishing 11
2.2.2 *Supervised Machine Learning* 12
BAB III DESKRIPSI SISTEM 14
3.1 Data dan Alat Penelitian 14
3.1.1 Data Penelitian 14
3.1.2 Alat Penelitian 14
3.2 Prosedur Penelitian 15
3.3 Analisis Serangan Phishing 17
3.3.1 *Analisis Kebutuhan* phishing 18
3.3.2 *Analisis Dataset* Email 18
3.3.3 Analisis KNN, Decision Tree, dan SVM 20
3.4 Perancangan 23
3.4.1 *Proses Analisis* Phishing 23
3.4.2 *Proses* Supervised Machine Learning 25
3.4.3 *Proses Model* Klasifikasi 26

3.4.4 *Evaluasi* Klasifikasi 27
3.5 Upaya Mitigasi 27
3.5.1 Tahapan Mitigasi 27
3.5.2 Hasil Mitigasi Berdasarkan Algoritma 29
3.5.3 Hasil Akhir Mitigasi 30
BAB IV HASIL DAN PENGUJIAN 31
4.1 Hasil 31
4.1.1 Implementasi Pengumpulan dan Identifikasi Data Email 31
4.1.2 Implementasi Pra-pemrosesan Data (Preprocessing) 32
4.1.3 Visualisasi Karakteristik Teks Email (Word Cloud) 33
4.1.4 Ekstraksi Fitur TF-IDF dan Pembagian Dataset 34
4.1.5 Implementasi dan Hasil K-Nearest Neighbor (KNN) 35
4.1.6 Implementasi dan Hasil Decision Tree 36
4.1.7 Implementasi dan Hasil Support Vector Machine (SVM) 37
4.1.8 *Visualisasi Perbandingan Kinerja Model* 39
4.2 Pengujian 39
4.2.1 Deskripsi Pengujian 40
4.2.2 Prosedur Pengujian 40
4.2.3 Data Hasil Pengujian 41
4.3 Analisis Data/Evaluasi 41
BAB V PENUTUP 44
5.1 Kesimpulan 44
5.2 Saran 45
DAFTAR PUSTAKA 46

DAFTAR GAMBAR

Gambar 3. 1 Prosedur Penelitian 15
Gambar 3. 2 Potongan Dataseet Phishing email 19
Gambar 3. 3 Proses Analisis Phishing 25
Gambar 4. 1 Hasil Pembersihan dan Labeling 33
Gambar 4. 2 gambar output Word Cloud 34
Gambar 4. 3 Hasil Confusion Matrix KNN 36
Gambar 4. 4 Hasil Confusion Matrix Decision Tree 37
Gambar 4. 5 Hasil Confusion Matrix SVM 38
Gambar 4. 6 Hasil gambar Bar Chart 39

BAB I
PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi yang sangat pesat telah membawa perubahan signifikan dalam berbagai aspek kehidupan, khususnya dalam pertukaran informasi digital melalui internet. Salah satu media komunikasi yang masih dominan digunakan hingga saat ini adalah surat elektronik (*email*), baik untuk kepentingan personal, bisnis, maupun institusional. Namun, tingginya ketergantungan terhadap *email* juga membuka peluang terjadinya kejahatan siber, terutama dalam bentuk penipuan berbasis manipulasi sosial atau yang dikenal sebagai *phishing* [1].

Serangan *phishing* terus mengalami peningkatan seiring dengan berkembangnya teknik rekayasa sosial yang semakin kompleks dan sulit dikenali oleh pengguna awam. *Phishing* tidak hanya menyebabkan kerugian finansial, tetapi juga berdampak pada kebocoran data sensitif seperti kredensial akun, informasi pribadi, dan data perbankan. Studi menunjukkan bahwa *email* masih menjadi vektor utama serangan *phishing* karena sifatnya yang masif, murah, dan mampu menjangkau banyak target dalam waktu singkat [1].

Metode konvensional dalam mendeteksi *phishing*, seperti pemfilteran berbasis aturan (*rule-based*) dan *blacklist*, dinilai tidak lagi efektif dalam menghadapi pola serangan *phishing* yang dinamis. Teknik tersebut cenderung bergantung pada pola statis sehingga sulit beradaptasi dengan variasi konten *email phishing* yang terus berkembang. Kondisi ini menuntut adanya pendekatan yang lebih adaptif dan mampu belajar dari data historis serangan [2].

Machine Learning (ML) sebagai bagian dari kecerdasan buatan menawarkan solusi yang lebih cerdas dalam mendeteksi *email phishing*. Dengan pendekatan *supervised learning*, sistem dapat mempelajari karakteristik *email phishing* dan *non-phishing* berdasarkan data berlabel. Pendekatan ini memungkinkan sistem untuk melakukan klasifikasi secara otomatis dan meningkatkan akurasi deteksi seiring bertambahnya data pelatihan [3].

Metode *supervised machine learning* banyak digunakan dalam klasifikasi

teks karena mampu menghasilkan performa yang baik ketika dataset memiliki label yang jelas. Dalam konteks *email phishing*, *supervised learning* sangat relevan karena data dapat dikategorikan ke dalam kelas *phishing* dan *non-phishing*. Beberapa algoritma populer yang sering digunakan antara lain *Naïve Bayes*, *Decision Tree*, *Random Forest*, dan *Support Vector Machine* (SVM) [1].

Support Vector Machine (SVM) dikenal sebagai algoritma *supervised learning* yang unggul dalam menangani data berdimensi tinggi, seperti data teks hasil ekstraksi fitur TF-IDF. SVM bekerja dengan mencari *hyperplane* optimal yang mampu memisahkan dua kelas data dengan margin maksimum. Penelitian sebelumnya menunjukkan bahwa SVM memiliki tingkat akurasi dan stabilitas yang lebih baik dibandingkan beberapa algoritma klasifikasi lainnya dalam mendeteksi *email phishing* [3].

Penelitian tentang penggunaan SVM dalam prediksi *email phishing* menunjukkan bahwa SVM efektif dalam mengenali pola serangan *phishing* pada *body email*, terutama pada dataset berbahasa alami [1]. Kombinasi metode TF-IDF dan algoritma *supervised machine learning* memberikan hasil yang signifikan dalam meningkatkan akurasi deteksi *phishing* dan mampu mengurangi *noise* pada data teks sehingga meningkatkan kualitas fitur yang digunakan oleh model klasifikasi [2]. Metode SVM dengan *kernel linear* menunjukkan hasil akurasi yang sangat tinggi, bahkan mencapai 100% pada dataset tertentu. Hal ini mengindikasikan bahwa SVM sangat andal dalam memisahkan *email phishing* dan *non-phishing*. Namun, penelitian tersebut masih terbatas pada dataset statis dan belum menguji ketahanan model terhadap variasi serangan *phishing* terbaru.

Tujuan utama dari penelitian ini adalah menganalisis serangan *phishing* pada pesan *email* serta mitigasi berbasis *supervised machine learning* yang mampu meningkatkan akurasi deteksi. Secara teoretis, penelitian ini memperkaya kajian keilmuan di bidang keamanan siber dan *text mining*. Secara praktis, hasil penelitian diharapkan dapat menjadi referensi bagi institusi, perusahaan, dan pengembang sistem keamanan dalam meningkatkan perlindungan terhadap ancaman *phishing* melalui *email*

1.2 Permasalahan

Berdasarkan latar belakang yang telah diuraikan, maka rumusan masalah dalam penelitian ini adalah bagaimana kinerja metode supervised machine learning dalam menganalisis dan mengklasifikasikan email phishing dan non-phishing menggunakan algoritma K-Nearest Neighbor (KNN), Decision Tree, dan Support Vector Machine (SVM)

1.3 Batasan Masalah

1. Penelitian ini hanya berfokus pada serangan *phishing* yang disampaikan melalui media *email*.
2. Metode yang digunakan dalam penelitian ini dibatasi pada pendekatan *supervised machine learning*.
3. Dataset yang digunakan merupakan dataset publik berjudul *Phishing Email* yang bersumber dari repositori *Kaggle*.

1.4 Tujuan Penelitian

Tujuan penelitian ini adalah menganalisis serangan *phishing* pada *email* menggunakan metode *supervised machine learning.* Pengujian dilakukan dengan 3 algoritma klasifikasi, yaitu *K-Nearest Neighbor* (KNN), *Decision Tree*, dan *Support Vector Machine* (SVM). Studi kasus pada *email phishing* dan *non-phishing*.

1.5 Manfaat Penelitian

1. Memberikan kontribusi keilmuan dalam bidang keamanan siber, khususnya terkait penerapan metode *supervised machine learning* dalam mendeteksi dan mengklasifikasikan pesan *email phishing*.
2. Menjadi referensi akademik bagi peneliti selanjutnya yang ingin mengembangkan atau membandingkan metode *machine learning* dalam mitigasi serangan *phishing* pada *email*.
3. Membantu memahami karakteristik dan pola pesan *email phishing* berdasarkan hasil klasifikasi menggunakan pendekatan *supervised machine*

learning.

4. Menghasilkan model deteksi *email phishing* yang dapat digunakan sebagai dasar dalam pengembangan sistem keamanan *email* yang lebih adaptif dan akurat.
5. Mendukung upaya mitigasi serangan *phishing* pada *email* dengan meningkatkan kemampuan sistem dalam membedakan *email phishing* dan *non-phishing* secara otomatis.
6. Mengurangi risiko kebocoran data sensitif pengguna akibat serangan *phishing* melalui peningkatan akurasi deteksi *email phishing*.
7. Memberikan manfaat praktis bagi pengguna, institusi, dan pengembang sistem keamanan dalam meningkatkan perlindungan terhadap ancaman *phishing* pada layanan *email.*

BAB II
TUJUAN PUSTAKA

2.1 Kajian Terdahulu

Dalam penyusunan skripsi ini, terdapat beberapa penelitian sebelumnya yang berkaitan dengan latar belakang dan masalah pada skripsi ini. Berikut ini adalah penelitian terdahulu yang berhubungan dengan skripsi ini antara lain :

Penelitian yang dilakukan oleh Fiqri Alwan, Ramzi Al Pasha, Eka Prasetya Kaspandiri, dan Muhammad Dzaky Al Fariz (2026) bertujuan untuk menganalisis karakteristik teknis dan pola penyamaran serangan *phishing* yang semakin kompleks serta mengevaluasi efektivitas upaya pencegahannya. Penelitian ini menggunakan metode deskriptif kualitatif dengan pendekatan analisis fitur URL, di mana sebanyak 14 sampel URL *phishing* aktif periode 2024–2025 dianalisis berdasarkan struktur domain, penggunaan protokol keamanan, dan teknik manipulasi visual seperti *typosquatting* serta penyalahgunaan *Top-Level* Domain (TLD) murah. Hasil penelitian menunjukkan bahwa sekitar 80% URL *phishing* telah menggunakan protokol HTTPS untuk memanipulasi kepercayaan pengguna, serta didominasi oleh teknik *typosquatting* dan penggunaan domain murah yang sulit terdeteksi oleh metode keamanan berbasis *blacklist*. Penelitian ini menyimpulkan bahwa pendekatan keamanan tradisional tidak lagi memadai dan menekankan pentingnya integrasi sistem deteksi berbasis *Machine Learning* dengan edukasi kesadaran keamanan pengguna sebagai strategi mitigasi yang lebih adaptif terhadap serangan *phishing* modern [1].

Penelitian terdahulu dilakukan oleh Bagas Adiansyah Souhoka dkk. pada tahun 2025 yang bertujuan untuk menganalisis modus operasi *phishing* di media sosial Facebook, dampaknya terhadap pengguna, serta strategi pencegahan yang efektif. Penelitian ini menggunakan metode deskriptif kualitatif dengan teknik pengumpulan data berupa wawancara terhadap korban *phishing*, observasi aktivitas pengguna, dan studi kasus serangan *phishing* di Facebook. Hasil

penelitian menunjukkan bahwa *phishing* di Facebook dilakukan melalui berbagai modus seperti pembuatan akun palsu, penyebaran tautan *phishing*, dan pemanfaatan iklan Facebook, dengan dampak yang signifikan berupa pencurian identitas, kerugian finansial, dan kecemasan psikologis pengguna. Penelitian ini juga menyimpulkan bahwa rendahnya kesadaran pengguna menjadi faktor utama kerentanan terhadap *phishing*, sehingga diperlukan upaya mitigasi berupa peningkatan edukasi pengguna, kewaspadaan terhadap tautan mencurigakan, perlindungan data pribadi, serta penerapan autentikasi dua faktor untuk meningkatkan keamanan akun [2].

Penelitian terdahulu dilakukan oleh Alfin Nur Fadli dan Jagad Satria Dewa Arta (2026) yang bertujuan untuk membangun model deteksi indikasi *phishing* pada pesan *email* menggunakan algoritma *Support Vector Machine* (SVM) guna meningkatkan keamanan komunikasi digital. Penelitian ini menggunakan metode *supervised machine learning* dengan tahapan *preprocessing* teks meliputi *cleaning*, *stopword removal*, dan *stemming*, serta ekstraksi fitur menggunakan metode *Term Frequency–Inverse Document Frequency* (TF-IDF). Dataset yang digunakan adalah *Phishing Validation Emails* sebanyak 2.000 data *email* yang terdiri dari *email* aman (ham) dan *phishing*, dengan pembagian data latih dan data uji sebesar 80:20. Evaluasi kinerja model dilakukan menggunakan *confusion matrix* dengan parameter akurasi, *presisi*, *recall*, dan F1-*score*. Hasil penelitian menunjukkan bahwa model SVM dengan *kernel linear* mampu mengklasifikasikan *email phishing* dan *email* aman secara sangat efektif, dengan nilai akurasi, *presisi*, *recall*, dan F1-*score* masing-masing mencapai 100%, sehingga menunjukkan bahwa kombinasi metode SVM dan TF-IDF sangat andal dalam mendeteksi pola serangan *phishing* pada pesan *email* [4].

Penelitian terdahulu dilakukan oleh Ilham Ramadhan dan Budi Triandi (2025) yang bertujuan untuk menerapkan algoritma *Random Forest* dalam menganalisis dan memprediksi *email phishing* sebagai upaya meningkatkan sistem keamanan komunikasi digital. Penelitian ini menggunakan metode *machine learning* dengan tahapan pengumpulan dataset *email* dari *Kaggle* yang menggabungkan *email phishing* dan *email* aman, dilanjutkan dengan proses *data*

cleaning, *preprocessing*, *exploratory data analysis*, pembagian data latih dan data uji dengan rasio 70:30, serta pembangunan model klasifikasi menggunakan algoritma *Random Forest*. Evaluasi kinerja model dilakukan menggunakan metrik akurasi, *presisi*, *recall*, F1-*score*, dan ROC-AUC. Hasil penelitian menunjukkan bahwa model *Random Forest* mampu mencapai akurasi sebesar 97,12% dengan kemampuan yang sangat baik dalam mengklasifikasikan *email* aman, serta nilai AUC sebesar 0,81 yang menandakan kemampuan diskriminatif model yang baik, meskipun masih terdapat keterbatasan dalam mendeteksi *email phishing* akibat ketidakseimbangan data, sehingga diperlukan pengembangan lanjutan untuk meningkatkan performa pada kelas minorit [5].

Berdasarkan penelitian yang dilakukan oleh M. Azmi Al Fadillah, Mochammad Guntur Ramadhan, dan Mohammad Erza Ariefiandi (2024), penelitian ini bertujuan untuk menganalisis ancaman *phishing* terhadap penggunaan *e-commerce* di Indonesia serta meningkatkan kesadaran pengguna terhadap bahaya *phishing*. Metode yang digunakan adalah pendekatan deskriptif kualitatif dengan teknik pengumpulan data melalui studi literatur dan penyebaran kuesioner kepada 46 responden pengguna *e-commerce*. Penelitian ini berfokus pada identifikasi tingkat pemahaman pengguna terhadap indikator *phishing*, jenis-jenis serangan *phishing*, serta upaya pencegahan yang dapat dilakukan. Hasil penelitian menunjukkan bahwa masih banyak pengguna *e-commerce* yang belum memahami indikator keamanan dan ciri-ciri *phishing* sehingga rentan menjadi korban, namun pengguna yang telah mendapatkan pelatihan menunjukkan kemampuan yang lebih baik dalam mengenali serangan *phishing*, yaitu sebesar 53%. Penelitian ini menyimpulkan bahwa edukasi berkelanjutan, peningkatan kesadaran pengguna, serta penerapan teknologi keamanan seperti filter *email* dan *One Time Password* (OTP) sangat diperlukan untuk meminimalkan ancaman *phishing* pada sektor *e-commerce* di Indonesia [4].

Penelitian terdahulu dilakukan oleh Devi Puspitasari dan Tata Sutabri (2023) yang bertujuan untuk menganalisis kejahatan *phishing* pada sektor *e-commerce* dengan studi kasus *marketplace Shopee* serta mengidentifikasi faktor penyebab dan upaya pencegahan serangan *phishing*. Penelitian ini menggunakan

metode kualitatif dengan pendekatan *literature review*, yaitu mengkaji berbagai sumber pustaka berupa jurnal, buku, dan laporan kasus *phishing*, serta memanfaatkan data pengaduan konsumen untuk memperkuat analisis. Hasil penelitian menunjukkan bahwa serangan *phishing* pada *e-commerce* umumnya dilakukan melalui *email*, tautan palsu, dan manipulasi sosial dengan menyamar sebagai pihak resmi, dengan faktor utama penyebabnya adalah minimnya pengetahuan pengguna, kebocoran data pribadi, serta ketertarikan pengguna terhadap promosi atau hadiah palsu. Penelitian ini menyimpulkan bahwa mitigasi *phishing* dapat dilakukan melalui edukasi pengguna, peningkatan kewaspadaan terhadap *email* dan URL mencurigakan, penggunaan perangkat lunak anti-*phishing*, serta penerapan sistem keamanan tambahan seperti *One Time Password* (OTP) untuk melindungi akun dan informasi sensitif pengguna [6].

Penelitian yang dilakukan oleh Purnama Ramadani Silalahi dkk. (2022) bertujuan untuk menganalisis keamanan transaksi *e-commerce* dalam mencegah terjadinya penipuan online, khususnya yang berkaitan dengan rendahnya tingkat kepercayaan dan kesadaran konsumen terhadap risiko kejahatan siber seperti *phishing*. Penelitian ini menggunakan metode kualitatif dengan pendekatan fenomenologi, yaitu dengan mengkaji berbagai fenomena penipuan transaksi online melalui studi literatur dan observasi terhadap kasus-kasus yang terjadi di masyarakat. Hasil penelitian menunjukkan bahwa penipuan dalam transaksi *e-commerce* dipengaruhi oleh beberapa faktor utama, antara lain minimnya pengetahuan pengguna, kebocoran data pribadi, ketertarikan terhadap hadiah palsu, kondisi ekonomi, serta lemahnya kebijakan keamanan. Selain itu, penelitian ini mengidentifikasi berbagai bentuk penipuan yang sering terjadi, seperti *phishing*, *pharming*, *pretexting*, dan *quid pro quo*, serta menekankan pentingnya peningkatan keamanan sistem dan edukasi pengguna sebagai upaya mitigasi penipuan online [7].

Penelitian terdahulu dilakukan oleh Aseh Ginanjar, Nur Widiyasono, dan Rohmat Gunawan (2018) yang bertujuan untuk menganalisis serangan *web phishing* pada layanan *e-commerce* serta mengungkap jejak serangan dan identitas pelaku *phishing* melalui analisis jaringan. Penelitian ini menggunakan metode

Network Forensic Process, yang meliputi tahapan akuisisi dan pengintaian data, analisis, serta *recovery*, dengan memanfaatkan data hasil *capture* jaringan berupa *file.pcap* dan alat bantu seperti *Wireshark*, *HashCalc*, dan layanan analisis domain. Hasil penelitian menunjukkan bahwa metode *Network Forensic Process* berhasil mengungkap berbagai informasi penting terkait serangan *phishing*, seperti alamat *email* pengirim dan penerima, waktu pengiriman *email phishing*, domain palsu (*phishing host*), *IP Address*, DNS, serta protokol jaringan yang terlibat seperti DNS, FTP, IMAP, SMTP, dan HTTP. Penelitian ini membuktikan bahwa pendekatan forensik jaringan efektif dalam merekonstruksi aktivitas serangan *phishing* pada *e-commerce* dan dapat digunakan sebagai dasar analisis keamanan siber untuk mengidentifikasi serta membuktikan terjadinya kejahatan *phishing* [8].

Penelitian terdahulu dilakukan oleh Made Ody Gita Permana (2021) yang bertujuan untuk menganalisis tingkat keamanan *platform e-commerce* di Indonesia terhadap risiko serangan *password harvesting phishing* serta mengidentifikasi mekanisme autentikasi yang efektif dalam mencegah serangan tersebut. Penelitian ini menggunakan metode *Action Research* yang terdiri dari lima tahapan, yaitu diagnosis, *planning*, *action*, *evaluation*, dan *learning*, dengan pengujian keamanan dilakukan menggunakan *Social Engineering Toolkit* (SET) pada 31 *platform e-commerce* yang dipilih melalui teknik *purposive* sampling. Hasil penelitian menunjukkan bahwa sebanyak 22 *platform* memiliki tingkat keamanan rendah karena masih menggunakan metode *Single-Page* Login atau URL khusus yang mudah direplikasi untuk *phishing*, sedangkan 9 *platform* lainnya memiliki tingkat keamanan tinggi karena menerapkan *Step-Wise Authentication* dan *Embedded Login*. Selain itu, efektivitas aplikasi SET dalam menguji keamanan tercatat sebesar 70,9% yang termasuk kategori tinggi, sehingga penelitian ini menyimpulkan bahwa penerapan metode autentikasi berlapis sangat efektif dalam memitigasi serangan *phishing* pada *platform e-commerce* [9].

Penelitian ini dilakukan oleh Putri Nur Izzati dan Kasmawi (2025) yang bertujuan untuk mengidentifikasi serta memperbaiki kerentanan keamanan pada

aplikasi mobile XYZ yang digunakan sebagai *platform* pelaporan kekerasan dalam rumah tangga dan kekerasan seksual terhadap anak. Penelitian ini menggunakan metode analisis statik dengan memanfaatkan *Mobile Security Framework* (MOBSF) untuk mendeteksi kerentanan keamanan pada aplikasi berbasis Android yang dikembangkan menggunakan Flutter. Proses penelitian meliputi analisis awal kerentanan, identifikasi tingkat risiko, perbaikan konfigurasi keamanan, serta analisis ulang untuk mengevaluasi efektivitas perbaikan yang dilakukan. Hasil penelitian menunjukkan bahwa aplikasi XYZ awalnya memiliki tingkat risiko tinggi dengan skor keamanan 37/100 dan ditemukan beberapa kerentanan kritis seperti konfigurasi debug aktif, penggunaan sertifikat debug, serta dukungan terhadap versi Android yang rentan. Setelah dilakukan perbaikan terhadap kerentanan utama tersebut, skor keamanan aplikasi meningkat menjadi 61/100 dengan tingkat risiko rendah (*low risk*), sehingga penelitian ini menyimpulkan bahwa penggunaan MOBSF efektif tidak hanya dalam mendeteksi tetapi juga sebagai acuan teknis dalam meningkatkan keamanan aplikasi mobile sebelum dirilis ke publik[10].

Penelitian terdahulu dilakukan oleh Dylan Kaisar Hasya, Destania Safitri, Dewangkoro Ramadhan Putra, Farhan Bilawa Gita Maulana, dan Nur Aini Rakhmawati (2024) yang bertujuan untuk menganalisis implikasi etika dalam profil dan strategi penipuan online, termasuk *phishing*, pada transaksi *e-commerce* di ranah *cybercrime* serta dampaknya terhadap kepercayaan konsumen. Penelitian ini menggunakan metode penelitian kualitatif dengan pendekatan survei, di mana data primer dikumpulkan melalui kuesioner daring kepada dua kelompok responden, yaitu mahasiswa Sistem Informasi ITS dan mahasiswa non Sistem Informasi ITS. Hasil penelitian menunjukkan bahwa penipuan online dalam transaksi *e-commerce* menimbulkan dampak emosional yang signifikan pada korban seperti panik, marah, dan sedih, serta menyebabkan kerugian finansial dan risiko kebocoran data pribadi yang pada akhirnya menurunkan tingkat kepercayaan konsumen terhadap *platform e-commerce*. Penelitian ini juga menyimpulkan bahwa rendahnya kesadaran etika pelaku dan minimnya edukasi keamanan digital pengguna menjadi faktor utama tingginya kasus penipuan

online, sehingga diperlukan peningkatan literasi keamanan siber, transparansi penjual, perbaikan mekanisme pelaporan, dan penerapan prinsip etika bisnis untuk menciptakan ekosistem *e-commerce* yang lebih aman dan berkelanjutan [11].

2.2 Teori Penunjang

2.2.1 Phishing

Phishing merupakan salah satu bentuk kejahatan siber yang dilakukan dengan cara menipu pengguna untuk memperoleh informasi sensitif secara tidak sah. Informasi yang menjadi target *phishing* biasanya meliputi data pribadi, akun pengguna, kata sandi, hingga informasi keuangan. Serangan ini dilakukan dengan berbagai metode, seperti pengiriman *email* palsu yang menyerupai pihak resmi, pembuatan situs web tiruan yang tampak meyakinkan, serta manipulasi tautan yang mengarahkan pengguna ke halaman berbahaya. Melalui teknik tersebut, pelaku berupaya memanfaatkan kelengahan dan kurangnya kewaspadaan pengguna agar secara sukarela memberikan informasi penting tanpa disadari [4].

1. Cara Kerja *Phishing*

   Serangan *phishing* pada *email* bekerja dengan memanfaatkan kepercayaan pengguna terhadap pesan yang terlihat resmi dan meyakinkan. Pelaku terlebih dahulu membuat *email* palsu yang menyerupai komunikasi dari pihak terpercaya, seperti perusahaan, layanan perbankan, atau *platform* digital. *Email* tersebut biasanya berisi pesan yang bersifat mendesak, seperti permintaan verifikasi akun, pembaruan data, atau peringatan keamanan. Di dalam *email phishing* terdapat tautan atau lampiran berbahaya yang mengarahkan pengguna ke situs web palsu. Ketika pengguna mengklik tautan dan memasukkan informasi sensitif, data tersebut akan dikirimkan kepada pelaku. Dalam konteks penelitian ini, karakteristik *email phishing* tersebut dianalisis dan dimanfaatkan sebagai fitur dalam metode *supervised machine learning* untuk membedakan *email phishing* dan *email non-phishing.*

2. Jenis Jenis Phishing

Phishing memiliki beberapa jenis berdasarkan teknik dan media yang digunakan. *Email phishing* merupakan jenis yang paling umum, di mana pelaku mengirimkan *email* palsu secara massal kepada banyak pengguna. *Spear phishing* menargetkan individu atau organisasi tertentu dengan pesan yang lebih personal dan spesifik. *Whaling* merupakan bentuk *phishing* yang ditujukan kepada pihak dengan jabatan tinggi, seperti pimpinan atau manajer. Selain itu, terdapat link *phishing* yang memanipulasi tautan dalam *email*, serta *website phishing* yang meniru tampilan situs resmi. Jenis-jenis *phishing* ini menjadi dasar dalam analisis pola serangan dan perancangan sistem mitigasi menggunakan metode *supervised machine learning*.

2.2.2 *Supervised Machine Learning*

Supervised machine learning merupakan suatu bidang keilmuan yang bersifat multidisipliner karena melibatkan berbagai disiplin ilmu, seperti ilmu komputer, statistika, ilmu kognitif, teknik, teori optimasi, serta beragam cabang matematika dan sains lainnya. Pendekatan ini mengintegrasikan konsep-konsep dari berbagai bidang tersebut untuk membangun model yang mampu mempelajari pola data dan menghasilkan prediksi secara sistematis. *Supervised machine learning* merupakan metode pembelajaran mesin yang digunakan untuk mempelajari hubungan antara data masukan dan label yang telah diketahui, sehingga dapat menghasilkan prediksi berupa estimasi nilai atau pengelompokan data ke dalam kelas tertentu [12].

Dalam *supervised machine learning* untuk klasifikasi, terdapat beberapa algoritma yang sering digunakan untuk mengelompokkan data ke dalam kategori tertentu.

1. *K-Nearest Neighbor*

   K-Nearest Neighbor (KNN) adalah salah satu metode klasifikasi dalam *supervised machine learning* yang bersifat sederhana dan tidak bergantung pada asumsi distribusi data tertentu. Metode ini mengklasifikasikan data baru dengan melihat kelas yang paling banyak muncul dari sejumlah *K* data latih terdekat. Penentuan kedekatan dilakukan

melalui perhitungan jarak antara data uji dan data pelatihan menggunakan metrik jarak tertentu. Berdasarkan tingkat kemiripan tersebut, KNN dapat menentukan kelas objek dengan membandingkannya terhadap data pelatihan yang sudah tersedia [5].

2. *Decision Tree*

Decision Tree merupakan model prediksi yang menggunakan struktur pohon atau hierarki untuk membantu proses pengambilan keputusan. Setiap cabang pada pohon merepresentasikan suatu kondisi atau aturan yang harus dipenuhi untuk melanjutkan ke cabang berikutnya, hingga akhirnya mencapai simpul daun yang menunjukkan hasil prediksi. Model ini mampu mengubah data yang awalnya berbentuk tabel menjadi struktur pohon keputusan, sehingga aturan-aturan yang digunakan dapat divisualisasikan secara terstruktur dan lebih mudah dipahami. Dengan proses tersebut, aturan yang kompleks dapat disederhanakan ke dalam bentuk model keputusan yang sistematis [5].

3. *Support Vector Machine*

Support Vector Machine (SVM) merupakan algoritma klasifikasi yang menggunakan pemetaan nonlinier untuk mentransformasikan data ke dalam dimensi yang lebih tinggi. Pendekatan ini memungkinkan SVM menangani data berdimensi tinggi dengan baik sehingga dapat meningkatkan tingkat akurasi prediksi. Prinsip utama SVM dalam proses klasifikasi adalah menentukan *hyperplane* terbaik yang mampu memisahkan dua kelas yang telah ditentukan secara optimal [5].

BAB III
DESKRIPSI SISTEM

3.1 Data dan Alat Penelitian

Penelitian ini menggunakan data dan alat penelitian yang mendukung proses analisis serangan *phishing* pada *email* serta penerapan metode *supervised machine learning* sebagai upaya mitigasi. Data penelitian berperan sebagai sumber utama dalam proses pembelajaran model, sedangkan alat penelitian digunakan untuk mengolah data, membangun model, dan mengevaluasi hasil penelitian.

3.1.1 Data Penelitian

Data yang digunakan dalam penelitian ini berupa dataset *email* yang terdiri dari *email phishing* dan *email non-phishing*. Data tersebut digunakan untuk menganalisis karakteristik serangan *phishing* pada *email* serta sebagai bahan pembelajaran pada metode *supervised machine learning*. Dataset mencakup konten *email*, seperti subjek, isi pesan, dan informasi pendukung lainnya yang relevan untuk proses klasifikasi. Data yang telah dikumpulkan selanjutnya melalui tahap praproses, pelabelan, dan pembagian data menjadi data latih dan data uji. Data latih digunakan untuk membangun model *supervised machine learning*, sedangkan data uji digunakan untuk mengevaluasi kinerja model dalam mendeteksi *email phishing*.

3.1.2 Alat Penelitian

Alat penelitian yang digunakan terdiri dari perangkat keras dan perangkat lunak yang mendukung proses perancangan, pengembangan, dan pengujian sistem. Adapun alat penelitian yang digunakan adalah sebagai berikut:

1. Kebutuhan Hardware

   Perangkat keras digunakan untuk menjalankan proses pengolahan data dan pelatihan model *machine learning*. Spesifikasi minimum perangkat keras yang digunakan dalam penelitian ini adalah sebagai berikut:

   a. Laptop

b. Prosesor Intel Core i5
c. RAM minimal 8 GB
d. Storage

B. Kebutuhan Software

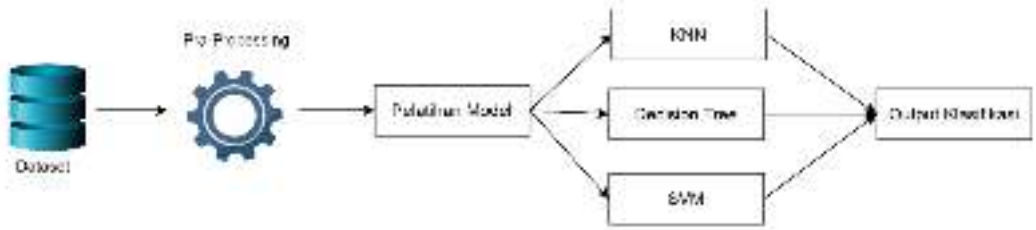
Perangkat lunak digunakan untuk mendukung proses pengolahan data *email*, penerapan algoritma *supervised machine learning*, serta evaluasi hasil penelitian. Perangkat lunak yang digunakan antara lain:

a. Windows
b. Python
c. Library Python
d. Jupyter Notebook

3.2 Prosedur Penelitian

Prosedur penelitian ini disusun untuk memberikan gambaran tahapan yang dilakukan dalam menganalisis serangan *phishing* pada *email* menggunakan *supervised machine learning*. Adapun tahapan-tahapan dalam *supervised machine learning* sebagai berikut:

A. Pengumpulan *Dataset*
B. *Pre-Processing*
C. Pelatihan Model
D. *Output* Klasifikasi



Gambar 3. 1 Prosedur Penelitian

Tahap pertama adalah pengumpulan dataset *email*. Dataset yang digunakan merupakan kumpulan *email* berlabel yang terdiri atas dua kategori, yaitu *phishing* dan *non-phishing*. Data dipilih karena merepresentasikan kondisi nyata komunikasi digital, di mana *email phishing* mengandung unsur manipulasi sosial, tautan mencurigakan, atau permintaan informasi sensitif, sedangkan *email non-phishing*

tidak memiliki karakteristik tersebut. Dataset selanjutnya dibagi menjadi data latih dan data uji dengan proporsi tertentu. Data latih digunakan untuk membangun model pembelajaran, sedangkan data uji digunakan untuk mengevaluasi kemampuan generalisasi model terhadap data yang belum pernah dipelajari sebelumnya. Tahap kedua adalah *pra-processing* data. Data *email* merupakan teks tidak terstruktur yang mengandung berbagai elemen *noise* seperti simbol, karakter khusus, angka, *hyperlink*, serta tag HTML. Oleh karena itu dilakukan proses pembersihan data untuk menghilangkan elemen-elemen yang tidak memiliki makna semantik dalam klasifikasi. Selanjutnya dilakukan normalisasi huruf menjadi huruf kecil, tokenisasi untuk memecah kalimat menjadi unit kata, serta penghapusan stopword guna mengurangi kata umum yang tidak memiliki kontribusi signifikan terhadap proses klasifikasi. Tahapan ini bertujuan meningkatkan kualitas data sehingga informasi penting lebih dominan dalam proses pembelajaran.

Setelah teks dibersihkan, dilakukan transformasi data ke dalam bentuk numerik menggunakan metode *Term Frequency–Inverse Document Frequency* (TF-IDF). Metode ini memberikan bobot pada setiap kata berdasarkan frekuensi kemunculannya dalam dokumen dan tingkat kelangkaannya dalam keseluruhan dataset. Dengan demikian, kata yang memiliki nilai diskriminatif tinggi terhadap kelas *phishing* atau *non-phishing* akan memperoleh bobot yang lebih besar. Representasi numerik ini menjadi dasar dalam proses pelatihan model *machine learning*. Tahap berikutnya adalah pelatihan model klasifikasi. Penelitian ini menggunakan tiga algoritma *supervised learning*, yaitu *K-Nearest Neighbor* (KNN), *Decision Tree*, dan *Support Vector Machine* (SVM). Ketiga algoritma tersebut dipilih untuk dibandingkan kinerjanya dalam mendeteksi *email phishing*. Proses pelatihan dilakukan menggunakan data latih yang telah direpresentasikan dalam bentuk vektor TF-IDF. Model mempelajari pola hubungan antara fitur teks dan label kelas sehingga mampu membentuk fungsi klasifikasi.

Tahap akhir adalah proses klasifikasi dan evaluasi *output*. Data uji yang telah melalui tahapan *preprocessing* dan ekstraksi fitur dimasukkan ke dalam masing-masing model untuk dilakukan prediksi. *Output* sistem berupa label kelas, yaitu *phishing* atau *non-phishing*. Hasil prediksi dibandingkan dengan label

sebenarnya untuk menghitung nilai akurasi, *precision*, *recall*, dan F1-*score*. Nilai-nilai tersebut digunakan sebagai dasar dalam menentukan efektivitas model sebagai upaya mitigasi serangan *phishing* pada *email*.

3.3 Analisis Serangan Phishing

Analisis serangan *phishing* dilakukan untuk mengidentifikasi karakteristik dan pola yang terdapat pada *email phishing* serta membandingkannya dengan *email non-phishing*. Analisis dilakukan berdasarkan isi pesan email yang terdapat dalam dataset, kemudian karakteristik tersebut digunakan sebagai dasar dalam proses klasifikasi menggunakan metode *supervised machine learning*.

Email phishing umumnya memiliki karakteristik berupa penggunaan kata-kata persuasif, pemberian ancaman atau tekanan kepada penerima, permintaan informasi pribadi atau kredensial, serta penyertaan tautan yang mencurigakan. Sementara itu, *email non-phishing* cenderung berisi komunikasi normal dan tidak memiliki unsur manipulasi yang bertujuan memperoleh informasi sensitif.

Berdasarkan karakteristik tersebut, analisis dilakukan melalui beberapa tahapan, yaitu:

A. Identifikasi karakteristik email phishing Mengidentifikasi pola kata, kalimat, permintaan informasi, serta tautan atau unsur mencurigakan yang terdapat pada email.
B. Identifikasi karakteristik email non-phishing Mengidentifikasi email yang memiliki pola komunikasi normal dan tidak menunjukkan indikasi manipulasi.
C. Preprocessing data Membersihkan data email dari karakter khusus, hyperlink, simbol, serta melakukan normalisasi huruf, tokenisasi, dan stopword removal.
D. Ekstraksi fitur menggunakan TF-IDF Mengubah data teks menjadi representasi numerik berdasarkan tingkat kepentingan setiap kata sehingga dapat diproses oleh algoritma machine learning.
E. Klasifikasi email Data yang telah diproses digunakan untuk melatih tiga

algoritma, yaitu KNN, Decision Tree, dan SVM.

F. Analisis hasil klasifikasi Hasil klasifikasi dibandingkan berdasarkan Accuracy, Precision, Recall, F1-Score, dan Confusion Matrix.

3.3.1 *Analisis Kebutuhan* phishing

Analisis kebutuhan dilakukan untuk menentukan komponen yang diperlukan dalam membangun sistem deteksi *phishing* berbasis *machine learning*. Sistem memerlukan data *email* berlabel sebagai sumber pembelajaran model. Data tersebut harus mencakup dua kategori utama, yaitu *email phishing* dan *email non-phishing*, agar model dapat mempelajari perbedaan karakteristik keduanya.

Dari sisi perangkat lunak, sistem membutuhkan lingkungan pemrograman yang mendukung pengolahan data dan *machine learning*, seperti bahasa pemrograman Python beserta pustaka pengolahan data, text mining, dan klasifikasi. Sistem juga membutuhkan modul *preprocessing* untuk membersihkan teks, modul ekstraksi fitur untuk mengubah teks menjadi numerik, serta modul klasifikasi untuk melakukan pelatihan dan prediksi.

Dari sisi fungsional, sistem harus mampu menerima data *email*, melakukan pembersihan teks, mengekstraksi fitur, melatih model, serta menghasilkan *output* berupa kategori *email*. Dari sisi nonfungsional, sistem diharapkan memiliki tingkat akurasi tinggi, mampu bekerja otomatis, serta dapat digunakan sebagai alat bantu mitigasi keamanan *email*.

3.3.2 *Analisis Dataset* Email

Data yang digunakan dalam penelitian ini merupakan dataset sekunder publik berjudul Phishing Email yang bersumber dari platform Kaggle (dipublikasikan oleh Subhajournal). Dataset ini berisikan 18.650 baris data teks email yang telah diklasifikasikan ke dalam dua label kategori utama, yaitu *Phishing Email* (email berbahaya) dan *Safe Email* (email aman). Setiap data merepresentasikan komunikasi digital nyata, di mana *phishing email* umumnya mengandung kata-kata persuasif, ancaman, permintaan kredensial, atau tautan

mencurigakan, sedangkan *safe email* berisi komunikasi normal tanpa unsur manipulasi sosial. Mengingat distribusi data di dunia nyata tidak seimbang, dataset ini juga mencerminkan kondisi tersebut di mana porsi *safe email* lebih dominan dibandingkan *phishing email*.
https://docs.google.com/spreadsheets/d/1EM6cszdB3G7k9pxNdTShWh5J6SharAdlJ8BCTcI4LBg/edit?usp=sharing

Gambar 3. 2 Potongan Dataseet Phishing email

Sebelum data dimasukkan ke dalam algoritma *machine learning*, data teks yang tidak terstruktur tersebut harus melalui beberapa tahapan prapemrosesan (*preprocessing*) agar siap diolah secara komputasi. Tahapan prapemrosesan teks yang diterapkan dalam penelitian ini adalah sebagai berikut:

A. Data Cleaning (Pembersihan Data): Tahap awal dilakukan untuk membersihkan anomali pada *dataframe*, yaitu dengan menghapus kolom *index* yang tidak bernama ("Unnamed: 0"), membuang baris data yang kosong/hilang (*missing values/dropna*), serta menghapus baris data yang duplikat (*drop duplicates*)
B. Penghapusan Hyperlink dan Tanda Baca: Sistem menggunakan *regular expression* (RegEx) untuk mendeteksi dan menghapus seluruh tautan URL (hyperlink) serta membuang seluruh karakter tanda baca dan simbol khusus (*non-alphanumeric*) dari isi pesan email
C. Case Folding (Normalisasi Huruf): Seluruh karakter teks yang tersisa diubah menjadi format huruf kecil (*lowercase*). Tujuannya adalah agar

komputer menganggap kata yang sama dengan kapitalisasi berbeda (misalnya "Bank" dan "bank") sebagai satu entitas yang identic

D. Penghapusan Spasi Berlebih: Spasi ganda atau spasi kosong di awal dan akhir kalimat dihapus agar format teks menjadi seragam dan padat
E. Label Encoding: Label kategori "*Phishing Email*" dan "*Safe Email*" yang berformat teks (string) diubah menjadi nilai numerik biner menggunakan fungsi LabelEncoder(), di mana label 0 merepresentasikan *Phishing Email* dan label 1 merepresentasikan *Safe Email*.

Setelah teks bersih, data mentah tersebut ditransformasikan ke dalam representasi vektor numerik menggunakan metode ekstraksi fitur *Term Frequency–Inverse Document Frequency* (TF-IDF). Metode ini menghitung tingkat kepentingan (bobot) sebuah kata berdasarkan frekuensi kemunculannya di dalam satu dokumen email dibandingkan dengan kelangkaannya di seluruh dataset. Representasi matriks numerik inilah yang kemudian dibagi menjadi *data latih* dan *data uji* untuk menjadi dasar proses pelatihan model klasifikasi.

3.3.3Analisis KNN, Decision Tree, dan SVM

Metode yang digunakan dalam penelitian ini adalah *supervised machine learning* dengan tiga algoritma klasifikasi, yaitu K-Nearest Neighbor (KNN), Decision Tree, dan Support Vector Machine (SVM). Pemilihan beberapa algoritma bertujuan untuk melakukan perbandingan performa sehingga dapat diketahui metode paling efektif dalam mendeteksi *phishing*. Algoritma ini diproses menggunakan bahasa pemrograman Python dan pustaka *scikit-learn* (sklearn).

A. Langkah-langkah Proses dan Implementasi K-Nearest Neighbor (KNN)

KNN bekerja berdasarkan kedekatan jarak antar data. Dalam sistem ini, langkah-langkah teori dan komputasinya berjalan secara terintegrasi sebagai berikut:

1. Inisialisasi Model dan Penentuan Parameter: Sistem memulai proses dengan mendefinisikan nilai ketetanggaan (K). Pada penelitian ini,

inisialisasi dilakukan menggunakan modul KNeighborsClassifier dengan menetapkan jumlah tetangga terdekat n_neighbors=5.

2. Proses Pelatihan (Fitting) Data Latih: Sistem menyiapkan data latih yang telah melalui proses ekstraksi fitur TF-IDF. Model kemudian menyimpan representasi matriks ruang vektor dari data latih beserta labelnya melalui eksekusi fungsi fit().
3. Kalkulasi Jarak dan Prediksi (Predicting): Saat data uji dimasukkan melalui pemanggilan fungsi predict(), sistem secara matematis menghitung jarak antara vektor email uji dengan seluruh vektor email latih.
4. Penentuan Mayoritas Kelas (Output): Setelah jarak dihitung, sistem mengurutkannya dari yang terkecil, mengambil 5 data (K=5) dengan jarak paling dekat, dan melakukan proses *voting* (perhitungan jumlah kelas terbanyak pada tetangga tersebut) untuk menentukan label akhir klasifikasi. kode program dari tahapan algoritma KNN di atas pada sistem adalah sebagai berikut:

```
from sklearn.neighbors import KNeighborsClassifier

# 1. Inisialisasi model KNN dengan K=5
knn = KNeighborsClassifier(n_neighbors=5)

# 2. Proses pelatihan model (Fitting) dengan data latih
knn.fit(x_train, y_train)

# 3 & 4. Kalkulasi jarak, penentuan kelas, dan prediksi pada data uji
pred_knn = knn.predict(x_test)
```

B. Langkah-langkah Proses dan Implementasi Decision Tree

Decision Tree membangun struktur pohon keputusan berdasarkan fitur kata (TF-IDF) yang paling informatif. Langkah proses pemecahan datanya adalah sebagai berikut:

1. Inisialisasi Kriteria Pemisahan Pohon: Inisialisasi dilakukan menggunakan kelas `DecisionTreeClassifier` dengan menetapkan parameter `random_state=0` agar hasil pohon stabil

dan konsisten.

2. Perhitungan Node dan Pembentukan Cabang (Fitting): Data fitur TF-IDF dilatih ke dalam model menggunakan perintah `fit()`. Di belakang layar, sistem secara rekursif menghitung nilai informasi setiap atribut teks, memilih atribut dengan nilai tertinggi sebagai akar (*root node*), dan terus membagi data ke cabang-cabang (*branches*) hingga seluruh data berujung di simpul daun (*leaf node*).
3. Penelusuran Aturan Pohon dan Prediksi: Data email uji ditransformasikan menjadi vektor numerik, lalu dilewatkan ke dalam struktur pohon keputusan melalui fungsi `predict()`.

Keputusan Akhir (Output): Sistem mencocokkan nilai fitur data uji dengan aturan-aturan logis pada setiap node cabang yang telah dibentuk, hingga berujung pada *leaf node* yang menghasilkan keputusan mutlak klasifikasi email tersebut. kode program dari tahapan algoritma Decision Tree di atas pada sistem adalah sebagai berikut:

```
from sklearn.tree import DecisionTreeClassifier

# 1. Inisialisasi model Decision Tree
dt = DecisionTreeClassifier(random_state=0)

# 2. Proses pembentukan node dan cabang pohon (Fitting)
dt.fit(x_train, y_train)

# 3 & 4. Penelusuran pohon bersyarat dan prediksi klasifikasi
pred_dt = dt.predict(x_test)
```

C. Langkah-langkah Proses dan Implementasi Support Vector Machine (SVM)

SVM bekerja dengan memetakan data ke dimensi tinggi dan membentuk garis pemisah (hyperplane) optimal. Langkah-langkah pemrosesannya dalam sistem ini adalah:

1. Inisialisasi Pemetaan Hyperplane: Karena vektor teks TF-IDF memiliki dimensi yang sangat tinggi, sistem mendefinisikan pembentukan batas pemisah berjenis linier menggunakan kelas `SVC` dengan parameter `kernel='linear'`.
2. Penentuan Support Vector dan Margin (Fitting): Matriks data latih dimasukkan ke dalam model menggunakan eksekusi `fit()`. Melalui

fungsi ini, perhitungan aljabar linier dijalankan untuk mencari *hyperplane* optimal yang memberikan jarak *margin* terlebar (maksimum) antara kumpulan vektor email kelas *phishing* dan *non-phishing*.

3. Pemetaan Data Uji dan Prediksi: Untuk menguji model, data email baru (data uji) diproses menggunakan `predict()`. Sistem akan memproyeksikan fitur teks email uji tersebut ke dalam ruang dimensi tinggi.

Hasil Klasifikasi Geometris (Output): Sistem mengkalkulasi posisi titik koordinat data uji terhadap letak *hyperplane*. Apabila nilainya jatuh di satu sisi batas *margin*, maka diklasifikasikan sebagai *phishing*, dan jika di sisi sebaliknya, diklasifikasikan sebagai email aman (*non-phishing*). kode program dari tahapan algoritma SVM di atas pada sistem adalah sebagai berikut:

```
from sklearn.svm import SVC

# 1. Inisialisasi model SVM dengan kernel pemisah linear
svm = SVC(kernel='linear')

# 2. Proses perhitungan margin dan hyperplane (Fitting)
svm.fit(x_train, y_train)

# 3 & 4. Pemetaan posisi data uji dan prediksi klasifikasi akhir
pred_svm = svm.predict(x_test)
```

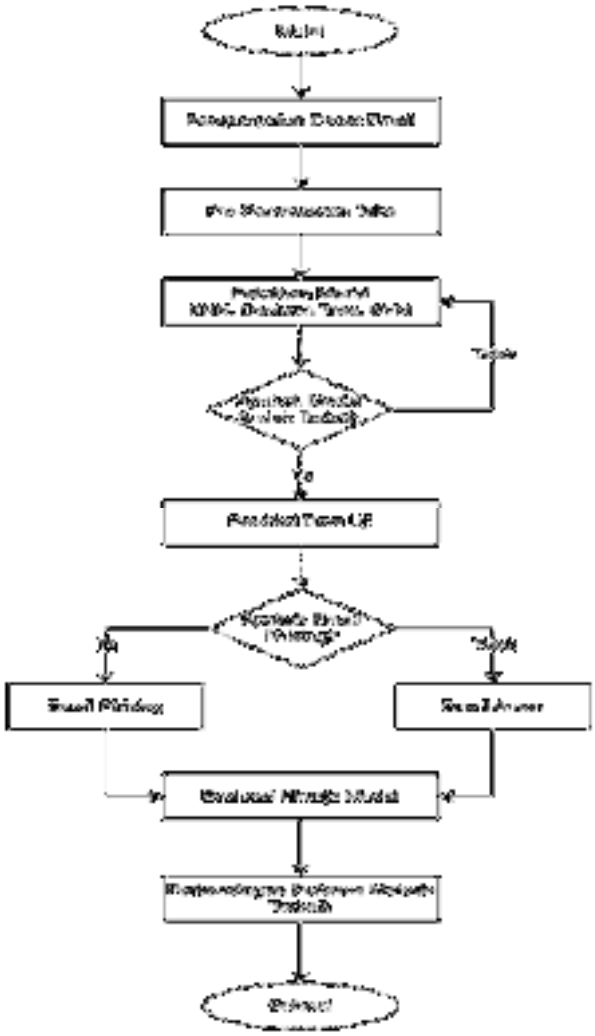
3.4 Perancangan

Perancangan ini disusun untuk membangun kerangka kerja penelitian dalam menganalisis serangan *phishing* pada *email* menggunakan *pendekatan supervised machine learning*. Tahap ini dirancang secara sistematis agar setiap proses penelitian saling terintegrasi dan mendukung pencapaian tujuan deteksi *phishing* yang akurat dan terukur. Perancangan dalam penelitian ini meliputi empat komponen utama, yaitu proses analisis *phishing*, proses *supervised machine learning*, proses model klasifikasi, dan evaluasi klasifikasi.

3.4.1 *Proses Analisis* Phishing

Proses analisis *phishing* merupakan tahapan awal dalam penelitian yang bertujuan untuk mengidentifikasi karakteristik, pola, dan indikator yang

membedakan *email phishing* dari *email non-phishing*. Tahap ini menjadi dasar dalam menentukan fitur yang akan digunakan pada proses klasifikasi berbasis *supervised machine learning*. Proses analisis *phishing* diawali dengan pengumpulan dataset *email*, yang terdiri dari *email phishing* dan *email non-phishing*. Dataset diperoleh dari sumber yang relevan dan telah memiliki label kelas untuk memastikan kejelasan kategori dalam proses pembelajaran model. Selanjutnya dilakukan pra-pemrosesan teks (*preprocessing*) untuk membersihkan dan menyiapkan data agar dapat dianalisis secara optimal. Tahapan ini meliputi pembersihan karakter khusus, penghapusan tag HTML, *case folding*, tokenisasi, serta penghapusan kata-kata yang tidak memiliki makna signifikan (*stopwords*). Proses ini bertujuan untuk mengurangi *noise* dan meningkatkan kualitas data. Adapun tahapan dalam Analisis Serangan *Phishing* pada *Email* dan Upaya Mitigasi Menggunakan Metode *Supervised Machine Learning* Sebagai berikut:



Gambar 3. 3 Proses Analisis Phishing

3.4.2 *Proses* Supervised Machine Learning

Proses *supervised machine learning* dirancang sebagai pipeline terstruktur yang terdiri dari beberapa tahapan berikut:

1. Data Email

   Dataset yang digunakan terdiri dari *email* berlabel *phishing* dan *email non-phishing*. Pembagian data dilakukan menggunakan skema training dan testing, misalnya dengan rasio 80:20 untuk memastikan model dapat diuji pada data yang tidak dilatih sebelumnya.

2. Proses *Preprocessing*

   Preprocessing dilakukan untuk membersihkan dan menormalisasi data sebelum diproses oleh algoritma klasifikasi. Tahap ini bertujuan

meningkatkan kualitas data dan mengurangi *noise*.

3. Proses Ekstraksi Fitur

   Ekstraksi fitur dilakukan untuk mengubah data teks menjadi representasi numerik agar dapat diproses oleh algoritma *machine learning*.

4. Proses Training Model

   Pada tahap ini, dataset training digunakan untuk melatih algoritma klasifikasi agar mampu mengenali pola *phishing* dan *non-phishing*. Model mempelajari hubungan antara fitur dan label kelas sehingga dapat membangun fungsi prediksi.

5. Proses Evaluasi

   Setelah proses pelatihan selesai, model diuji menggunakan data testing. Pengujian ini bertujuan untuk mengukur kemampuan model dalam mengklasifikasikan data baru yang belum pernah dilihat sebelumnya.

3.4.3 *Proses Model* Klasifikasi

Penelitian ini menggunakan tiga algoritma *supervised machine learning* untuk membandingkan performa klasifikasi, yaitu:

1. K-Nearest Neighbor *(KNN)*

   KNN merupakan algoritma berbasis jarak yang mengklasifikasikan data baru berdasarkan kedekatannya dengan sejumlah tetangga terdekat (K). Klasifikasi ditentukan berdasarkan mayoritas kelas dari tersebut.

2. Decision Tree Decision Tree

   Merupakan algoritma berbasis struktur pohon yang membagi data berdasarkan atribut tertentu untuk membentuk aturan keputusan. Model ini mudah diinterpretasikan karena menghasilkan aturan berbentuk cabang keputusan.

3. Support Vector Machine *(SVM)*

   SVM bekerja dengan mencari hyperplane optimal *yang memisahkan dua kelas dengan margin maksimum. Algoritma ini efektif dalam menangani data berdimensi tinggi seperti data teks hasil ekstraksi fitur TF-IDF.*

3.4.4 *Evaluasi* Klasifikasi

Evaluasi dilakukan untuk mengukur kinerja model klasifikasi dalam mendeteksi *email phishing*. Metrik evaluasi yang digunakan meliputi:

1. Accuracy adalah mengukur persentase keseluruhan prediksi yang benar terhadap total data.
2. Precision adalah mengukur tingkat ketepatan model dalam memprediksi email sebagai phishing.
3. Recall adalah mengukur kemampuan model dalam mendeteksi seluruh email phishing yang sebenarnya.
4. F1-Score adalah merupakan rata-rata harmonik antara precision dan recall yang memberikan keseimbangan antara keduanya.

3.5 Upaya Mitigasi

Upaya mitigasi pada penelitian ini dilakukan dengan memanfaatkan metode *supervised machine learning* untuk mendeteksi dan mengklasifikasikan *email phishing* dan *non-phishing* secara otomatis. Model dibangun menggunakan data email yang telah memiliki label sehingga dapat mempelajari karakteristik masing-masing kelas dan memberikan prediksi terhadap *email* baru. Pendekatan ini digunakan sebagai alat bantu untuk memberikan deteksi dini terhadap *email* yang berpotensi mengandung serangan *phishing*.

3.5.1 Tahapan Mitigasi

1. *Identifikasi* Email

   Tahap pertama adalah menerima atau mengidentifikasi data *email* yang akan dianalisis. *Email* yang digunakan dalam penelitian terdiri atas dua kelas, yaitu *email phishing* dan *email non-phishing*.

   Email phishing memiliki karakteristik seperti penggunaan kata persuasif, ancaman atau tekanan kepada penerima, permintaan informasi pribadi, serta tautan yang mencurigakan. Karakteristik tersebut menjadi dasar dalam proses pembelajaran model untuk membedakan *email phishing* dari *email* yang normal.

2. Preprocessing

Email yang diperoleh tidak langsung dimasukkan ke dalam algoritma. Data terlebih dahulu melalui tahap *preprocessing* untuk mengurangi *noise* dan menghasilkan data yang lebih sesuai untuk proses klasifikasi.

Tahapan *preprocessing* meliputi:

1. pembersihan karakter khusus.
2. penghapusan tag HTML.
3. *case folding*.
4. tokenisasi.
5. penghapusan stopword.

Tahapan tersebut digunakan untuk membersihkan teks dan mempertahankan informasi yang relevan dari isi *email*.

3. Ekstraksi Fitur TF-IDF

Setelah preprocessing, teks *email* diubah menjadi bentuk numerik menggunakan *Term Frequency–Inverse Document Frequency* (TF-IDF).

Tahap ini penting karena algoritma *machine learning* tidak memproses teks secara langsung. TF-IDF memberikan nilai berdasarkan tingkat kepentingan kata dalam suatu *email* sehingga karakteristik teks dapat digunakan sebagai fitur untuk proses klasifikasi.

4. Pembentukan Model Klasifikasi

Data yang telah diproses kemudian digunakan untuk membangun tiga model klasifikasi:

1. *K-Nearest Neighbor* (KNN)
2. *Decision Tree*
3. *Support Vector Machine* (SVM)

Ketiga model tersebut dilatih menggunakan data training untuk mempelajari hubungan antara fitur *email* dan label *phishing/non-phishing*

5. Deteksi *Email Phishing*

Setelah model dilatih, data *email* yang belum pernah dilihat model digunakan sebagai data testing.

Model kemudian memberikan hasil klasifikasi berupa:

Phishing = email terindikasi sebagai serangan *phishing*.

Non-Phishing = email tidak terindikasi sebagai *phishing*.

Dengan demikian, model dapat digunakan sebagai alat bantu deteksi dini untuk mengidentifikasi *email* yang berpotensi membahayakan pengguna.

Evaluasi Model

Setiap model kemudian dievaluasi menggunakan:

1. *Accuracy*
2. *Precision*
3. *Recall*
4. F1-*Score*
5. *Confusion Matrix*

Evaluasi dilakukan untuk mengetahui kemampuan model dalam membedakan *email phishing* dan *non-phishing* serta melihat kesalahan klasifikasi pada masing-masing kelas.

3.5.2 Hasil Mitigasi Berdasarkan Algoritma

1. K-Nearest Neighbor (*KNN*)

KNN memperoleh akurasi 65,71% dan F1-score 61,75%. Meskipun memiliki *recall phishing* yang tinggi, KNN menghasilkan cukup banyak false positive, yaitu *email non-phishing* yang salah dikategorikan sebagai *phishing*.

Kondisi tersebut menunjukkan bahwa KNN belum optimal untuk dijadikan model utama mitigasi pada penelitian ini.

Kesimpulan mitigasi KNN:

KNN dapat digunakan sebagai model pembanding, tetapi belum direkomendasikan sebagai model utama.

2. Decision Tree

Decision Tree menghasilkan akurasi 92,96% dan F1-score 94,24%. Model ini mampu mengklasifikasikan *email phishing* dan *non-phishing* dengan lebih seimbang dan memiliki tingkat kesalahan yang relatif kecil.

Decision Tree dapat menjadi alternatif model mitigasi yang cukup andal, meskipun masih memiliki potensi *overfitting* apabila parameter model tidak diatur secara optimal. Kesimpulan mitigasi *Decision Tree*: *Decision Tree* dapat digunakan sebagai alternatif dalam mendeteksi *email phishing*.

3. Support Vector Machine (SVM)

Hasil terbaik diperoleh oleh *Support Vector Machine* dengan kernel linear.

SVM memperoleh:

1. *Accuracy* = 98,35%
2. *F1-Score* = 98,66%

Hasil *confusion matrix* juga menunjukkan jumlah kesalahan klasifikasi yang sangat kecil dan distribusi kesalahan yang relatif seimbang. Hal ini menunjukkan bahwa SVM mampu meminimalkan kesalahan baik *false positive* maupun *false negative*.

Karena memberikan hasil terbaik dibandingkan KNN dan *Decision Tree*, SVM dipilih sebagai model utama dalam upaya mitigasi serangan *phishing.*

3.5.3 Hasil Akhir Mitigasi

Mitigasi yang dihasilkan dalam penelitian ini berupa model deteksi dini *email phishing* berbasis supervised machine learning. Model dibangun melalui *preprocessing* dan ekstraksi fitur TF-IDF, kemudian dilakukan klasifikasi menggunakan KNN, *Decision Tree*, dan SVM. Berdasarkan hasil pengujian, SVM dengan *kernel linear* memberikan performa terbaik dengan akurasi 98,35% dan F1-*Score* 98,66%. Oleh karena itu, SVM direkomendasikan sebagai model utama untuk mendeteksi *email* yang terindikasi *phishing*. Hasil deteksi tersebut dapat digunakan sebagai peringatan awal bagi pengguna agar tidak membuka tautan mencurigakan, memberikan informasi pribadi, atau mengikuti instruksi yang terdapat dalam *email* yang terindikasi *phishing*.

BAB IV
HASIL DAN PENGUJIAN

4.1 Hasil

Hasil penelitian ini diperoleh untuk mencapai tujuan penelitian, yaitu menganalisis serangan *phishing* pada *email* serta mengevaluasi efektivitas metode *supervised machine learning* sebagai upaya mitigasi. Analisis dilakukan dengan memanfaatkan dataset *email* berlabel yang terdiri dari *email phishing* dan *email non-phishing*, sehingga model dapat mempelajari pola karakteristik serangan *phishing* berdasarkan konten dan struktur pesan *email*. Hasil yang diperoleh memberikan gambaran mengenai kemampuan sistem dalam membedekan *email phishing* dan *non-phishing* secara otomatis.

Objek pada penelitian ini adalah kumpulan data *email* yang digunakan sebagai data pelatihan dan pengujian model. Setiap *email* diproses melalui tahapan *preprocessing teks* untuk menghilangkan *noise*, kemudian dilakukan ekstraksi fitur menggunakan metode TF-IDF agar informasi teks dapat direpresentasikan dalam bentuk numerik. Selanjutnya, model *supervised machine learning* dilatih untuk mengenali pola serangan *phishing* berdasarkan hubungan antara fitur dan label kelas *email.*

Pelaksanaan pengujian dilakukan setelah model selesai dilatih, dengan menggunakan data uji untuk mengukur kinerja klasifikasi. Hasil pengujian menunjukkan bahwa model mampu mengklasifikasikan *email phishing* dan *email non-phishing* dengan tingkat akurasi yang tinggi serta kesalahan klasifikasi yang minimal. Temuan ini menunjukkan bahwa penerapan metode *supervised machine learning* efektif sebagai Upaya mitigasi serangan *phishing* pada *email*, karena mampu memberikan deteksi dini dan meningkatkan keamanan komunikasi digital.

4.1.1 Implementasi Pengumpulan dan Identifikasi Data Email

Langkah pertama dalam sistem adalah memuat dataset mentah yang diperoleh dari Kaggle menggunakan pustaka pandas. Sistem membaca file Phishing_Email.csv dan menampilkan cuplikan datanya untuk mengidentifikasi struktur teks sebelum diproses.

Kode Implementasi:

```
import pandas as pd
# Memuat dataset email phishing ke dalam DataFrame Pandas
Df =
pd.read_csv("/root/.cache/kagglehub/datasets/.../Phishing_Email.csv")
df.head()
```

Output Sistem :

Unnamed: 0	Email Text	Email Type
0	re : 6 . 1100 , disc : uniformitarianism , re ...	Safe Email
1	the other side of * galicismos * * galicismo *...	Safe Email
2	re : equistar deal tickets are you still avail...	Safe Email
3	\nHello I am your hot lil horny toy.\n I am...	Phishing Email
4	software at incredibly low prices (86 % lower...	Phishing Email

Hasil di atas memperlihatkan bahwa data mentah masih memiliki kolom indeks yang tidak diperlukan (Unnamed: 0) dan isi teks masih mengandung karakter *noise* (seperti tanda baca dan enter `\n`).

4.1.2 Implementasi Pra-pemrosesan Data (Preprocessing)

Tahap ini diimplementasikan untuk membersihkan *noise* pada data dan mengubah label kategori teks menjadi numerik biner (0 untuk Phishing, 1 untuk Aman) menggunakan `LabelEncoder`:

Kode Implementasi Pembersihan dan Labeling:

```
import re
from sklearn.preprocessing import LabelEncoder

# Menghapus kolom tidak relevan, nilai kosong, dan duplikat
df.drop(["Unnamed: 0"], axis=1, inplace=True)
df.dropna(inplace=True, axis=0)
df.drop_duplicates(inplace=True)
print("Dimension of the row data:", df.shape)

# Merubah label kategorikal menjadi numerik
le = LabelEncoder()
df["Email Type"] = le.fit_transform(df["Email Type"])
```

```
# Fungsi Preprocessing Teks
def preprocess_text(text):
    text = re.sub(r'http\S+', '', text) # Hapus hyperlink
    text = re.sub(r'[^\w\s]', '', text) # Hapus tanda baca
    text = text.lower()                 # Ubah ke huruf kecil
    text = re.sub(r'\s+', ' ', text).strip() # Hapus spasi
berlebih
    return text

# Penerapan fungsi pada DataFrame
df["Email Text"] = df["Email Text"].apply(preprocess_text)
df.head()
```

Output Sistem:

Dimension of the row data: (17538, 2)

	Email Text	Email Type
0	re : 6 . 1100 , disc : uniformitarianism , re ...	Safe Email
1	the other side of * galicismos * * galicismo *...	Safe Email
2	re : equistar deal tickets are you still avail...	Safe Email
3	\nHello I am your hot lil horny toy.\n I am...	Phishing Email
4	software at incredibly low prices (86 % lower...	Phishing Email

Gambar 4. 1 Hasil Pembersihan dan Labeling

Keluaran di atas membuktikan bahwa dari 18.650 baris data mentah, tersisa 17.538 data bersih. Teks telah seragam (huruf kecil tanpa tanda baca) dan siap diproses secara matematis.

4.1.3 Visualisasi Karakteristik Teks Email (Word Cloud)

Sistem memvisualisasikan kata-kata yang memiliki frekuensi tinggi dalam dataset secara keseluruhan menggunakan pustaka `wordcloud`.

Kode Implementasi:

```
from wordcloud import WordCloud
import matplotlib.pyplot as plt
all_mails = " ".join(df['Email Text'])
word_cloud    =    WordCloud(stopwords="english",    width=800,
height=400, background_color='white').generate(all_mails)

plt.figure(figsize=(10,6))
plt.imshow(word_cloud, interpolation='bilinear')
plt.axis("off")
plt.show()
```

Output Sistem:

Gambar 4. 2 gambar output Word Cloud

Visualisasi ini menunjukkan kata-kata dominan yang nantinya akan dibobotkan pada tahap ekstraksi fitur.

4.1.4 Ekstraksi Fitur TF-IDF dan Pembagian Dataset

Data teks diubah menjadi vektor numerik (`TfidfVectorizer`) dengan batas maksimal 10.000 fitur, lalu dibagi menjadi data latih (80%) dan data uji (20%).

Kode Implementasi:

```
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.model_selection import train_test_split
```

```
tf = TfidfVectorizer(stop_words="english", max_features=10000)
feature_x = tf.fit_transform(df["Email Text"]).toarray()
y_tf = np.array(df['Email Type'])

x_train, x_test, y_train, y_test = train_test_split(feature_x,
y_tf, train_size=0.8, random_state=0)
```

Proses ini menghasilkan matriks fitur `x_train` dan `x_test` yang menjadi *input* wajib bagi algoritma klasifikasi.

4.1.5 Implementasi dan Hasil K-Nearest Neighbor (KNN)

Algoritma KNN mencari kedekatan jarak fitur dengan menetapkan nilai tetangga terdekat `n_neighbors=5`.

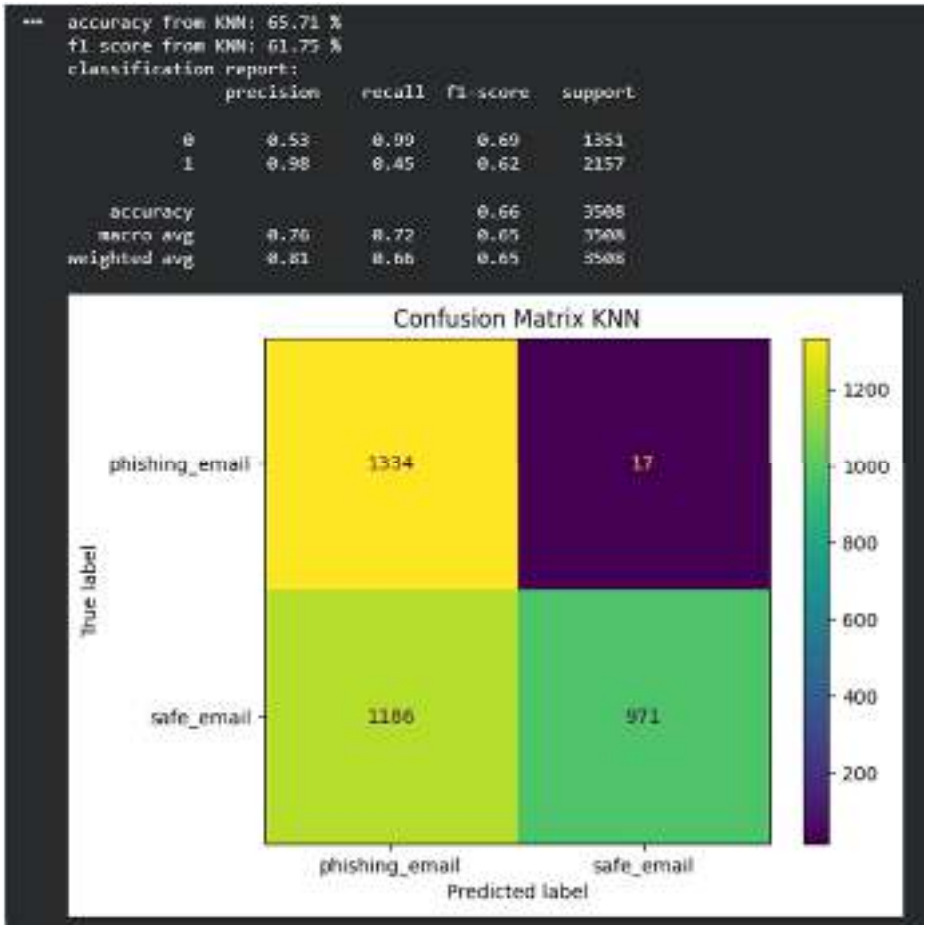
Kode Implementasi:

```
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score, f1_score,
classification_report

knn = KNeighborsClassifier(n_neighbors=5)
knn.fit(x_train, y_train)
pred_knn = knn.predict(x_test)

print(f"accuracy from KNN: {accuracy_score(y_test,
pred_knn)*100:.2f} %")
print("classification report:\n",
classification_report(y_test, pred_knn))
```

Output Sistem:



Gambar 4. 3 Hasil Confusion Matrix KNN

Hasil eksekusi program memperlihatkan KNN mampu mendeteksi phishing dengan recall (0.99), tetapi memiliki precision rendah (0.53) sehingga menghasilkan banyak false positive pada data safe email.

4.1.6 Implementasi dan Hasil Decision Tree

Algoritma Decision Tree diimplementasikan dengan random_state=0 agar pembentukan pohon keputusan tetap konsisten.

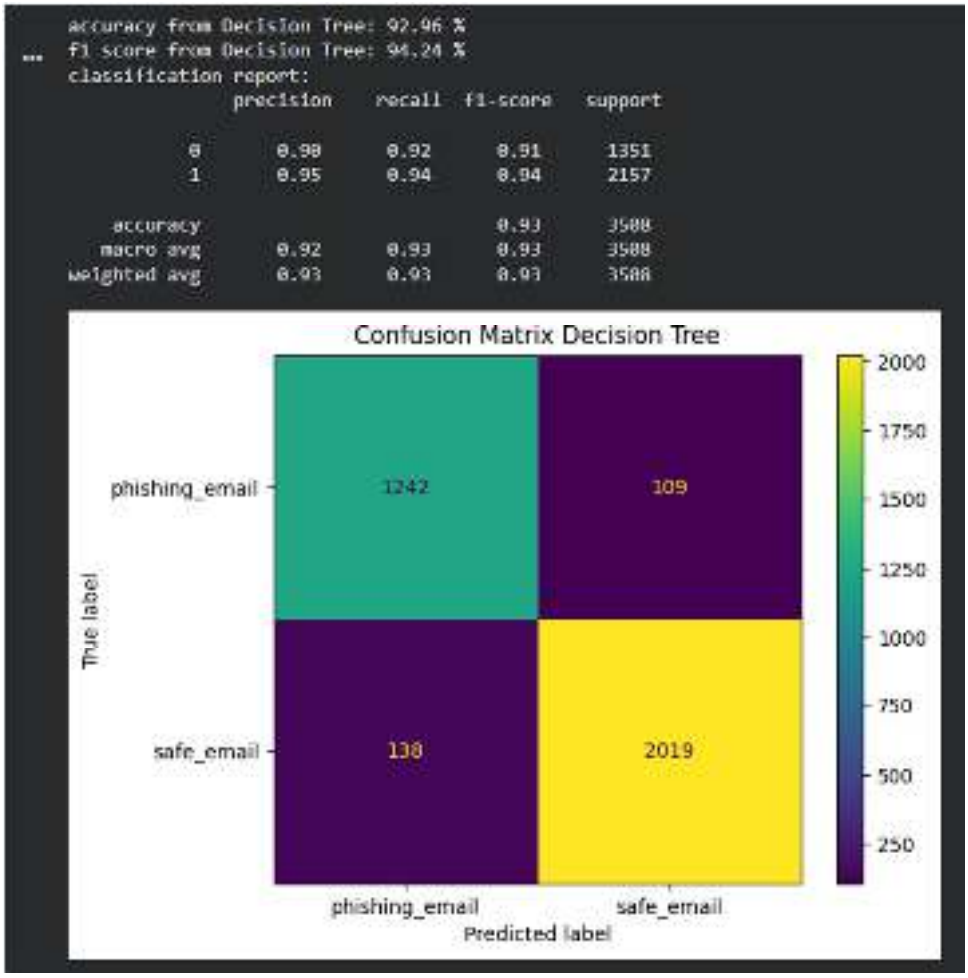
Kode Implementasi:

```
from sklearn.tree import DecisionTreeClassifier

dt = DecisionTreeClassifier(random_state=0)
dt.fit(x_train, y_train)
pred_dt = dt.predict(x_test)
```

```
print(f"accuracy from Decision Tree: {accuracy_score(y_test, pred_dt)*100:.2f} %")
print("classification report:\n", classification_report(y_test, pred_dt))
```

Output Sistem:



Gambar 4. 4 Hasil Confusion Matrix Decision Tree

Aturan hierarki pada Decision Tree dieksekusi secara sistematis dan menghasilkan metrik pemisahan kelas yang stabil (akurasi 92.96%), jauh lebih seimbang antara presisi dan recall dibandingkan KNN.

4.1.7 Implementasi dan Hasil Support Vector Machine (SVM)

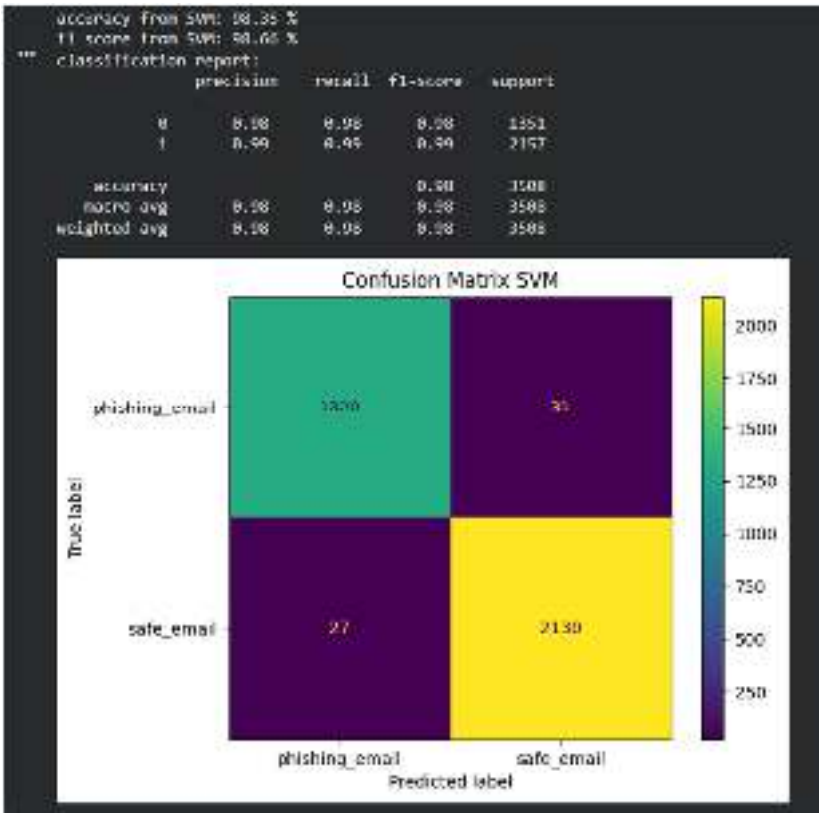
Untuk memisahkan 10.000 fitur teks yang berdimensi tinggi, algoritma SVM disetel dengan garis pemisah lurus (kernel='linear').

Kode Implementasi:

```
from sklearn.svm import SVC

svm = SVC(kernel='linear')
svm.fit(x_train, y_train)
pred_svm = svm.predict(x_test)
print(f"accuracy from SVM: {accuracy_score(y_test,
pred_svm)*100:.2f} %")
print("classification report:\n",
classification_report(y_test, pred_svm))
```

Output Sistem:



Gambar 4. 5 Hasil Confusion Matrix SVM

Perhitungan aljabar pada SVM memetakan hyperplane secara ekstrem presisi, menghasilkan akurasi tertinggi (98.35%) dengan tingkat kesalahan (false positive / negative) yang nyaris menyentuh batas minimal.

4.1.8 *Visualisasi Perbandingan Kinerja Model*

Sebagai tahap evaluasi akhir, sistem membandingkan kinerja seluruh model yang diuji ke dalam bentuk grafik batang interaktif menggunakan plotly.

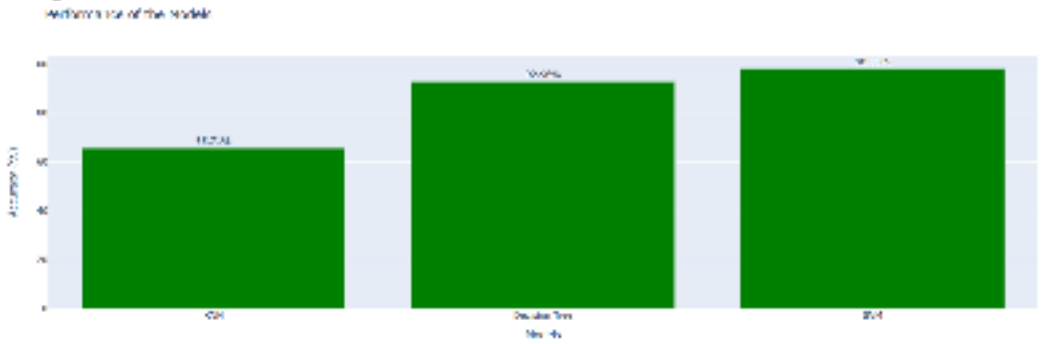
Kode Implementasi:

```
import plotly.express as px

models = ['KNN', 'Decision Tree', 'SVM']
accuracies = [65.71, 92.96, 98.35]

fig = px.bar(x=models, y=accuracies, labels={'x': 'Models',
'y': 'Accuracy (%)'},
             title="Performance of the Models",
text=accuracies)
fig.update_traces(texttemplate='%{text:.2f}%',
textposition='outside', marker_color='green')
fig.show()
```

Output Sistem:



Gambar 4. 6 Hasil gambar Bar Chart

Grafik di atas menegaskan bahwa SVM adalah metode mitigasi phishing yang paling adaptif dan optimal diterapkan pada layanan email dalam penelitian ini.

4.2 Pengujian

Pengujian pada penelitian ini dilakukan untuk mengetahui tingkat kinerja dan keandalan sistem dalam mendeteksi serangan *phishing* pada *email* menggunakan metode *supervised machine learning*. Tahap pengujian bertujuan untuk mengevaluasi sejauh mana model yang dibangun mampu mengklasifikasikan *email* ke dalam kategori *phishing* dan *email non-phishing* secara akurat berdasarkan

data uji yang belum pernah dilihat sebelumnya oleh model. Hasil pengujian ini menjadi dasar dalam menilai efektivitas masing-masing algoritma dalam mendukung upaya mitigasi serangan *phishing* pada *email*.

4.2.1 Deskripsi Pengujian

Pengujian dilakukan terhadap beberapa model *supervised machine learning*, yaitu *K-Nearest Neighbor* (KNN), *Decision Tree*, dan *Support Vector Machine* (SVM). Setiap model diuji menggunakan data uji yang berasal dari dataset *email* berlabel *phishing* dan *email non-phishing* yang telah melalui tahap *preprocessing* dan ekstraksi fitur menggunakan metode TF-IDF. Data uji digunakan untuk mengukur kemampuan generalisasi model dalam mengenali pola serangan *phishing* secara otomatis. Parameter evaluasi yang digunakan dalam pengujian meliputi *accuracy*, *precision*, *recall*, dan F1-*score*. Penggunaan metrik tersebut bertujuan untuk memberikan gambaran kinerja model secara lebih komprehensif, terutama karena dataset memiliki distribusi kelas yang tidak seimbang antara *email non-phishing* dan *email phishing*.

4.2.2 Prosedur Pengujian

Prosedur pengujian dimulai dengan membagi dataset menjadi data latih dan data uji. Data latih digunakan untuk membangun dan melatih model *supervised machine learning*, sedangkan data uji digunakan untuk mengevaluasi performa model. Seluruh data telah melalui proses *preprocessing*, seperti pembersihan teks, penghapusan *noise*, serta transformasi data teks menjadi bentuk numerik menggunakan TF-IDF agar dapat diproses oleh algoritma *machine learning*. Setelah proses pelatihan selesai, setiap model digunakan untuk melakukan prediksi terhadap data uji. Hasil prediksi kemudian dibandingkan dengan label asli untuk menghitung nilai *accuracy*, *precision*, *recall*, dan F1-*score*. Selanjutnya, *confusion matrix* dianalisis untuk melihat distribusi prediksi benar dan salah pada masing-masing kelas *email*. Proses ini dilakukan secara konsisten pada seluruh model agar hasil pengujian dapat dibandingkan secara objektif. Tahap akhir dari prosedur pengujian adalah melakukan analisis komparatif terhadap hasil evaluasi masing-masing model. Analisis ini bertujuan untuk menentukan model terbaik yang memiliki kinerja

paling optimal dalam mendeteksi *email phishing*.

4.2.3 Data Hasil Pengujian

Tahap pengujian dilakukan untuk mengevaluasi kinerja algoritma *supervised machine learning* dalam mengklasifikasikan *email phishing* dan *email non-phishing*. Pada penelitian ini, pengujian difokuskan pada tiga algoritma, yaitu *K-Nearest Neighbor* (KNN), *Decision Tree*, dan *Support Vector Machine* (SVM). Ketiga algoritma tersebut dipilih karena memiliki karakteristik dan pendekatan yang berbeda dalam proses klasifikasi, sehingga dapat memberikan gambaran komparatif mengenai efektivitas masing-masing metode dalam mitigasi serangan *phishing* berbasis *email*.

Dataset *email* yang digunakan sebelumnya telah melalui tahap prapemrosesan, termasuk pembersihan data, transformasi teks, serta ekstraksi fitur menggunakan metode *Term Frequency–Inverse Document Frequency* (TF-IDF). Selanjutnya, data dibagi menjadi data latih dan data uji untuk memastikan bahwa proses evaluasi dilakukan secara objektif terhadap kemampuan generalisasi model. Model dilatih menggunakan data latih dan kemudian diuji menggunakan data uji yang tidak terlibat dalam proses pembelajaran.

Pengujian performa model dilakukan dengan menggunakan beberapa metrik evaluasi, yaitu akurasi**,** *precision***,** *recall*, dan F1-*score*, serta didukung dengan analisis *confusion matrix*. Akurasi digunakan untuk mengukur tingkat ketepatan model secara keseluruhan, sedangkan *precision* dan *recall* digunakan untuk mengevaluasi kemampuan model dalam mengklasifikasikan masing-masing kelas, khususnya dalam mendeteksi *email phishing*. F1-*score* digunakan sebagai metrik gabungan untuk menilai keseimbangan antara *precision* dan *recall*. *Confusion matrix* digunakan untuk menganalisis kesalahan klasifikasi secara lebih rinci pada setiap kelas.

4.3 Analisis Data/Evaluasi

Hasil pengujian menunjukkan bahwa setiap algoritma memiliki tingkat performa yang berbeda dalam mengklasifikasikan *email phishing* dan *email non-phishing*. Pada algoritma *K-Nearest Neighbor* (KNN), diperoleh nilai akurasi

sebesar 65,71% dan F1-*score* sebesar 61,75%. Nilai ini menunjukkan bahwa kinerja KNN masih tergolong sedang dan belum optimal. Meskipun model mampu mendeteksi *email phishing* dengan tingkat *recall* yang sangat tinggi, yaitu 0,99, nilai *precision* yang rendah pada kelas *phishing* menunjukkan banyaknya *email non-phishing* yang salah diklasifikasikan sebagai *phishing*. Hal ini menyebabkan tingginya jumlah *false positive* dan berdampak pada rendahnya kemampuan model dalam mengenali *email non-phishing* secara konsisten. Kondisi ini diperkuat oleh *confusion matrix* yang menunjukkan bahwa sebagian besar kesalahan klasifikasi terjadi pada kelas *email non-phishing*. Dengan demikian, KNN dinilai kurang efektif sebagai metode utama dalam sistem mitigasi *phishing* berbasis *email* pada penelitian ini.

Berbeda dengan KNN, algoritma *Decision Tree* menunjukkan performa yang lebih baik dan stabil. Hasil pengujian menghasilkan nilai akurasi sebesar 92,96% dan F1-*score* sebesar 94,24%. Model mampu mengklasifikasikan kedua kelas dengan cukup seimbang, ditunjukkan oleh nilai *precision* dan *recall* yang tinggi pada kelas *phishing* dan *email non-phishing*. *Confusion matrix* menunjukkan bahwa kesalahan klasifikasi relatif kecil dan tidak didominasi oleh salah satu kelas tertentu. Hal ini mengindikasikan bahwa *Decision Tree* mampu mempelajari pola karakteristik *email phishing* dan *email non-phishing* secara efektif melalui aturan keputusan yang terbentuk dari fitur TF-IDF. Meskipun demikian, *Decision Tree* tetap memiliki potensi *overfitting* apabila tidak dilakukan pengaturan parameter lanjutan.

Hasil terbaik diperoleh dari algoritma *Support Vector Machine* (SVM) dengan kernel linear. Model ini menghasilkan nilai akurasi sebesar 98,35% dan F1-*score* sebesar 98,66%, yang menunjukkan tingkat kinerja yang sangat tinggi. Nilai *precision* dan *recall* yang hampir sempurna pada kedua kelas menunjukkan bahwa SVM mampu meminimalkan kesalahan klasifikasi, baik dalam bentuk *false positive* maupun *false negative*. *Confusion matrix* memperlihatkan jumlah kesalahan yang sangat kecil dan distribusi yang seimbang antar kelas, menandakan stabilitas dan konsistensi model. Kemampuan SVM dalam membentuk *hyperplane* pemisah yang optimal pada ruang fitur berdimensi tinggi menjadikannya sangat sesuai untuk data

teks berbasis TF-IDF.

Secara keseluruhan, hasil pengujian menunjukkan bahwa *Support Vector Machine* merupakan algoritma paling optimal dalam mitigasi serangan *phishing* pada *email* dibandingkan dengan KNN dan *Decision Tree*. *Decision Tree* dapat dijadikan alternatif yang cukup andal, sedangkan KNN lebih sesuai digunakan sebagai model pembanding. Dengan demikian, SVM direkomendasikan sebagai model utama dalam sistem deteksi dan mitigasi serangan *phishing email* pada penelitian ini.

BAB V
PENUTUP

5.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan mengenai analisis serangan *phishing* pada *email* serta upaya mitigasinya menggunakan metode *supervised machine learning*, dapat disimpulkan bahwa pendekatan pembelajaran terawasi mampu memberikan solusi yang efektif dalam mendeteksi dan membedakan *email phishing* dan *email non-phishing* secara otomatis. Proses prapemrosesan data dan ekstraksi fitur menggunakan metode TF-IDF terbukti mampu merepresentasikan karakteristik teks *email* ke dalam bentuk numerik yang dapat dipelajari dengan baik oleh model klasifikasi. Hasil pengujian menunjukkan bahwa setiap algoritma *supervised machine learning* yang digunakan memiliki tingkat performa yang berbeda. Algoritma *K-Nearest Neighbor* (KNN) menghasilkan tingkat akurasi dan F1-*score* yang relatif rendah dibandingkan algoritma lainnya. Meskipun KNN mampu mendeteksi *email phishing* dengan tingkat *recall* yang sangat tinggi, model ini cenderung menghasilkan kesalahan klasifikasi yang besar pada *email non-phishing*, sehingga meningkatkan jumlah *false positive*. Hal tersebut menunjukkan bahwa KNN kurang optimal untuk diterapkan pada data teks berdimensi tinggi seperti *email* berbasis TF-IDF.

Algoritma *Decision Tree* menunjukkan performa yang lebih baik dan stabil dengan tingkat akurasi dan F1-*score* yang tinggi. Model ini mampu mengklasifikasikan *email phishing* dan *email non-phishing* secara seimbang, serta memiliki tingkat kesalahan klasifikasi yang relatif kecil. *Decision Tree* mampu membentuk aturan keputusan yang jelas berdasarkan fitur-fitur penting, sehingga efektif dalam mengenali pola serangan *phishing*. Namun demikian, model ini tetap memiliki potensi *overfitting* apabila tidak dilakukan pengaturan parameter secara optimal. Algoritma *Support Vector Machine* (SVM) dengan *kernel linear* memberikan hasil terbaik dibandingkan algoritma lainnya. Model ini mencapai tingkat akurasi dan F1-*score* yang sangat tinggi serta mampu meminimalkan kesalahan klasifikasi pada kedua kelas. Kemampuan SVM dalam membentuk

hyperplane pemisah yang optimal pada ruang fitur berdimensi tinggi menjadikannya sangat sesuai untuk klasifikasi *email* berbasis teks. Dengan performa yang konsisten dan stabil, SVM dapat direkomendasikan sebagai model utama dalam sistem deteksi dan mitigasi serangan *phishing* pada *email*. Penelitian ini membuktikan bahwa penerapan metode *supervised machine learning*, khususnya *Support Vector Machine*, efektif digunakan sebagai upaya mitigasi serangan *phishing* pada *email* dan mampu smeningkatkan keamanan komunikasi digital melalui deteksi dini terhadap *email phishing*.

5.2 Saran

Berdasarkan hasil penelitian yang telah dilakukan, terdapat beberapa saran yang dapat dijadikan acuan untuk pengembangan penelitian selanjutnya. Penelitian berikutnya disarankan untuk menggunakan dataset yang lebih besar dan lebih beragam agar model yang dibangun memiliki kemampuan generalisasi yang lebih baik terhadap berbagai jenis serangan *phishing* yang terus berkembang. Selain itu, penelitian lanjutan dapat mempertimbangkan penggunaan teknik ekstraksi fitur lain, seperti *word embedding* atau *deep learning-based feature representation*, untuk membandingkan efektivitasnya dengan metode TF-IDF dalam klasifikasi *email phishing*. Dari sisi implementasi sistem, hasil penelitian ini dapat dikembangkan lebih lanjut ke dalam bentuk aplikasi atau sistem deteksi *phishing* berbasis *email* yang dapat diintegrasikan langsung dengan layanan *email*. Dengan demikian, sistem tidak hanya bersifat akademis, tetapi juga dapat memberikan manfaat praktis dalam meningkatkan keamanan pengguna terhadap ancaman serangan *phishing*

DAFTAR PUSTAKA

[1] F. Alwan, R. Al Pasha, E. P. Kaspandiri, M. Dzaky, and A. Fariz, "Analisis Serangan Phising Dan Upaya Pencegahannya Pada Sistem Informasi," vol. 2, no. 2, pp. 161–165, 2026.

[2] B. A. Souhoka, R. A. Fadillah, and M. Fathan, "Analisis Strategi Pencegahan Phising Studi Kasus Pada Media Sosial Facebook," vol. 3, no. 1, pp. 10–22, 2025.

[3] A. N. Fadli, J. Satria, and D. Arta, "Analisis Indikasi Phishing Pada Pesan Email Menggunakan Metode SVM," vol. 5, pp. 834–838, 2026.

[4] M. A. Al Fadillah, M. G. Ramadhan, and M. E. Ariefiandi, "Analisis Ancaman Phishing Terhadap Penggunaan E-Commerce di Indonesia," *J. Inform. Inf. Secur.*, vol. 5, no. 2, pp. 85–96, 2024, doi: 10.31599/aes6yw36.

[5] I. Of *et al.*, "Penerapan Algoritma Random Forest Untuk Analisis Dan Prediksi Phishing E-Mail Menggunakan Machine Learning," vol. 2, no. 1, pp. 463–475, 2025.

[6] D. Puspitasari and T. Sutabri, "Analisis kejahatan phishing pada sektor e-commerce di marketplace Shopee," *J. Digit. Teknol. Inf.*, vol. 6, no. 2, pp. 76–81, 2023, doi: 10.32502/digital.v6i2.5653.

[7] Purnama Ramadani Silalahi, Aisy Salwa Daulay, Tanta Sudiro Siregar, and Aldy Ridwan, "Analisis Keamanan Transaksi E-Commerce Dalam Mencegah Penipuan Online," *Profit J. Manajemen, Bisnis dan Akunt.*, vol. 1, no. 4, pp. 224–235, 2022, doi: 10.58192/profit.v1i4.481.

[8] A. Ginanjar, N. Widiyasono, and R. Gunawan, "Analisis Serangan Web Phishing pada Layanan E-commerce dengan Metode Network Forensic Process," no. 2, pp. 147–157, 2018, doi: 10.21460/jutei.2018.22.103.

[9] M. Aplikasi, S. Engineering, T. Set, and P. T. Informatika, "Analisis keamanan platform".

[10] P. N. Izzati, "Static Analysis-Based Security Enhancement for Mobile Applications Using Mobile Security Framework (MOBSF)," vol. 9, no. 4, pp. 1272–1278, 2025.

[11] Dylan Kaisar Hasya, Destania Safitri, Dewangkoro Ramadhan Putra, Farhan Bilawa Gita Maulana, and Nur Aini Rakhmawati, "Implikasi Etika dalam Profil dan Strategi Penipuan Online dalam Transaksi e-Commerce di Ranah Cybercrime," *J. Manaj. Ris. Inov.*, vol. 2, no. 1, pp. 236–247, 2023, doi: 10.55606/mri.v2i1.2219.

[12] R. S. Nurhalizah and R. Ardianto, "Analisis Supervised dan Unsupervised Learning pada Machine Learning : Systematic Literature Review," vol. 4, no.1,pp.61–72,2024