

## Analisis Perbandingan Algoritma Random Forest, SVM, dan Neural Network untuk Klasifikasi Risiko Keamanan E-Learning

### *Comparative Analysis of Random Forest, SVM, and Neural Network Algorithms for E-Learning Security Risk Classification*

Rahmat Taufik Nugraha<sup>1\*</sup>, Deden Hidayat<sup>2</sup>, Fathoni Mahardika<sup>3</sup>

Program Studi Informatika, Universitas Sebelas April, Indonesia<sup>123</sup>

rtaufik@unsap.ac.id<sup>1</sup>, 220660121166@student.unsap.ac.id<sup>2</sup>, fathoni@unsap.ac.id<sup>3</sup>

#### Abstrak

Akselerasi adopsi platform *e-learning* akibat pandemi COVID-19 telah memperluas permukaan serangan siber di sektor pendidikan, menuntut solusi deteksi ancaman yang lebih canggih. *Machine learning (ML)* menawarkan pendekatan proaktif untuk mengatasi tantangan ini, namun belum terdapat konsensus mengenai algoritma yang paling optimal. Studi ini bertujuan melakukan analisis komparatif empiris terhadap tiga algoritma *ML* terkemuka *Random Forest (RF)*, *Support Vector Machine (SVM)*, dan *Neural Network (NN)* dalam mengklasifikasikan risiko keamanan pada lingkungan *e-learning*. Menggunakan dataset sintetik *Classroom Data Security Threats*, penelitian ini menerapkan metodologi kuantitatif yang mencakup pra-pemrosesan data dan evaluasi model menggunakan metrik *accuracy*, *F1-score*, dan *ROC-AUC*. Hasil eksperimen menunjukkan kinerja yang sangat terbatas dari ketiga model. *NN* mencapai akurasi tertinggi, namun hanya sebesar 33.5%, sedikit di atas *SVM* (32.5%) dan *RF* (29.5%). Secara signifikan, skor *ROC-AUC* untuk semua model berada di sekitar 0.5, yang mengindikasikan kemampuan prediktifnya tidak lebih baik dari tebakan acak. Kegagalan serempak ini menyiratkan bahwa tantangan utama bukan terletak pada pemilihan algoritma, melainkan pada kualitas prediktif dataset dan ketiadaan optimisasi *hyperparameter*. Temuan ini menggarisbawahi pentingnya kualitas data dan rigor metodologis sebagai prasyarat fundamental untuk pengembangan sistem keamanan siber berbasis *ML* yang efektif.

Kata kunci: Keamanan Siber; Klasifikasi Ancaman; Machine Learning; Pembelajaran Daring.

#### Abstract

The accelerated adoption of *e-learning* platforms due to the COVID-19 pandemic has expanded the cybersecurity attack surface in the education sector, demanding more sophisticated threat detection solutions. *Machine learning (ML)* offers a proactive approach to address this challenge, yet there is no consensus on the most optimal algorithm. This study aims to conduct an empirical comparative analysis of three prominent *ML* algorithms *Random Forest (RF)*, *Support Vector Machine (SVM)*, and *Neural Network (NN)* in classifying security risks within an *e-learning* environment. Utilizing the synthetic *Classroom Data Security Threats* dataset, this research employed a quantitative methodology involving data preprocessing and model evaluation based on *accuracy*, *F1-score*, and *ROC-AUC* metrics. The experimental results revealed severely limited performance across all three models. The *NN* achieved the highest accuracy, yet only at 33.5%, marginally outperforming *SVM* (32.5%) and *RF* (29.5%). Significantly, the *ROC-AUC* scores for all models hovered around 0.5, indicating that their predictive capability was no better than random guessing. This simultaneous failure implies that the primary challenge lies not in algorithm selection but in the predictive quality of the dataset and the absence of *hyperparameter* optimization. These findings underscore the critical importance of data quality and methodological rigor as fundamental prerequisites for developing effective *ML*-based cybersecurity systems.

Keywords: Cybersecurity; Machine Learning; Online Learning; Threat Classification.

Naskah diterima 26 Juli 2025; direvisi 26 Februari 2026; dipublikasi 7 Maret 2026.

JATI is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.



## 1. Pendahuluan

Perkembangan teknologi digital telah lama mendorong transformasi bertahap dalam sektor pendidikan, namun pandemi global COVID-19 berfungsi sebagai katalisator yang memicu akselerasi eksponensial dalam adopsi platform pembelajaran daring. Institusi pendidikan di seluruh dunia secara masif dan dalam waktu singkat terdorong untuk mengintegrasikan sistem perangkat lunak *e-learning*, termasuk *learning management systems (LMS)*, sebagai infrastruktur vital untuk keberlangsungan edukasi [1]. Transisi yang bersifat reaktif dan tergesa-gesa ini, meskipun berhasil merevolusi metode penyampaian pendidikan, secara tidak terhindarkan juga membuka berbagai celah kerentanan keamanan yang sebelumnya tidak menjadi perhatian utama. Dalam urgensi untuk memastikan aksesibilitas dan fungsionalitas, banyak institusi mengesampingkan implementasi protokol keamanan siber yang komprehensif, menciptakan sebuah ekosistem digital yang kaya akan data sensitif namun minim proteksi. Platform-platform ini menjadi

repositori masif bagi informasi krusial, mulai dari data identitas pribadi mahasiswa dan staf pengajar, materi akademik, hingga catatan evaluasi, yang menjadikannya target bernilai tinggi bagi para pelaku kejahatan siber. Konsekuensinya, lanskap ancaman bagi sektor pendidikan meluas secara dramatis, mencakup berbagai vektor serangan seperti pelanggaran data (*data breaches*), serangan *phishing*, infeksi *malware* dan *ransomware*, ancaman dari dalam (*insider threats*), serta kerentanan yang timbul dari konfigurasi keamanan sistem yang tidak memadai [1]. Situasi ini secara kolektif memperluas permukaan serangan (*attack surface*) dan secara signifikan meningkatkan postur risiko keamanan siber bagi institusi pendidikan di seluruh dunia.

Menghadapi ancaman siber yang semakin canggih dan dinamis, mekanisme pertahanan tradisional yang mengandalkan deteksi berbasis tanda tangan (*signature-based detection*) terbukti tidak lagi memadai. Solusi konvensional sering kali gagal mengidentifikasi serangan modern yang kompleks dan mampu mengubah bentuknya, seperti *zero-day exploits* dan *polymorphic malware* [2]. Sebagai respons terhadap keterbatasan ini, paradigma *Machine Learning* (ML) telah muncul sebagai pendekatan alternatif yang sangat menjanjikan, menawarkan kapabilitas deteksi yang lebih proaktif, adaptif, dan cerdas. ML telah menjadi instrumen yang kuat dalam memperkokoh benteng keamanan siber, dengan kemampuannya untuk menganalisis data dalam volume masif, mengidentifikasi pola-pola tersembunyi, dan pada akhirnya meningkatkan efektivitas mekanisme deteksi dan pertahanan terhadap ancaman [2]. Keunggulan fundamental dari pendekatan ML terletak pada kemampuannya untuk mengekstraksi pola atau wawasan terkait insiden keamanan dari data mentah dan membangun model berbasis data yang sesuai untuk prediksi dan klasifikasi [3]. Dengan melatih model pada data historis, sistem berbasis ML dapat mempelajari seperti apa perilaku normal dalam sebuah jaringan atau sistem, dan kemudian secara otomatis mengidentifikasi anomali atau deviasi yang mencurigakan. Kemampuan ini memungkinkan deteksi ancaman secara *real-time* dengan tingkat akurasi yang lebih tinggi dan tingkat positif palsu (*false positive*) yang lebih rendah dibandingkan dengan sistem berbasis aturan statis, menjadikannya komponen krusial dalam arsitektur keamanan siber modern.

Kajian literatur akademik secara konsisten menunjukkan potensi dan efektivitas penerapan berbagai algoritma *Machine Learning* dalam domain keamanan siber. Sebagai contoh, algoritma Random Forest (RF), yang merupakan salah satu teknik *ensemble learning* terkemuka, telah menunjukkan kinerja yang sangat impresif dalam tugas klasifikasi ancaman. Sebuah studi yang mengimplementasikan RF untuk memprediksi kategori serangan siber melaporkan pencapaian akurasi yang sangat tinggi, yaitu sebesar 99.84%, yang mengindikasikan keandalannya sebagai model prediktif [4]. Di sisi lain, Support Vector Machine (SVM) dikenal luas sebagai algoritma klasifikasi yang tangguh dan efektif, terutama dalam menangani data dengan ruang fitur berdimensi tinggi (*high-dimensional feature spaces*) yang merupakan karakteristik umum dari data log keamanan. Sebuah penelitian komparatif bahkan menyimpulkan bahwa SVM menunjukkan efektivitas yang lebih superior dibandingkan dengan Artificial Neural Network (ANN) dalam konteks deteksi intrusi, dengan mencatatkan akurasi pelatihan sebesar 99,87% dan akurasi pengujian sebesar 99,81% [5]. Temuan ini menempatkan SVM sebagai kandidat kuat untuk analisis data keamanan yang kompleks dan bervariasi. Selain itu, arsitektur Jaringan Saraf Tiruan (*Neural Network* - NN), termasuk varian *deep learning* seperti *Multilayer Perceptron* (MLP) dan *Recurrent Neural Network* (RNN), telah menunjukkan fleksibilitas dan performa yang luar biasa di berbagai skenario. Sebuah implementasi MLP untuk tugas klasifikasi berhasil mencapai tingkat akurasi 91% [6], sementara pendekatan yang lebih canggih menggunakan kombinasi XGBoost dan *Long Short-Term Memory* (LSTM), sebuah varian dari RNN, mampu mencapai akurasi pengujian sebesar 88.13% pada dataset keamanan jaringan yang menantang [7]. Serangkaian hasil penelitian ini secara kolektif menegaskan bahwa keluarga algoritma RF, SVM, dan NN memiliki potensi yang sangat besar untuk diaplikasikan secara efektif dalam upaya mengamankan platform *e-learning*.

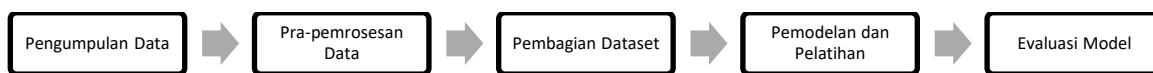
Meskipun berbagai penelitian telah memvalidasi keunggulan masing-masing algoritma secara terpisah, masih terdapat celah penelitian yang signifikan dan mendesak untuk diisi. Kinerja yang dilaporkan dalam literatur sering kali sangat bervariasi dan sangat bergantung pada konteks spesifik penelitian, seperti jenis ancaman yang dianalisis, karakteristik unik dari dataset yang digunakan, serta metrik evaluasi yang menjadi prioritas. Akibatnya, hingga saat ini belum terbentuk sebuah konsensus yang jelas mengenai algoritma mana yang secara definitif paling unggul untuk diterapkan dalam domain keamanan *e-learning* secara holistik. Secara khusus, belum ada studi komprehensif yang melakukan analisis perbandingan secara langsung (*head-to-head*) antara tiga algoritma yang sangat kuat ini Random Forest, SVM, dan Neural Network dengan menggunakan satu dataset terpadu yang secara spesifik merepresentasikan spektrum luas risiko keamanan dalam ekosistem *e-learning*, seperti *phishing*, *malware*, anomali upaya akses, dan pelanggaran data. Ketiadaan tolok ukur (*benchmark*) yang kontekstual ini menciptakan ketidakpastian praktis bagi para pengembang sistem, administrator, dan pembuat kebijakan di institusi pendidikan yang ingin mengimplementasikan solusi keamanan berbasis ML yang paling efektif dan efisien. Oleh karena itu, kebaruan (*novelty*) dari penelitian ini terletak pada analisis komparatif empiris secara langsung antara Random Forest, SVM, dan Neural Network untuk tugas klasifikasi risiko keamanan *e-learning* yang

komprehensif. Penelitian ini secara strategis memposisikan diri untuk mengisi celah pengetahuan tersebut dengan menyediakan bukti kuantitatif yang solid mengenai performa relatif ketiga model ini dalam konteks yang sangat spesifik dan relevan. Dengan demikian, kontribusi utama dari penelitian ini adalah penyediaan sebuah *template* sistem klasifikasi ancaman yang berbasis bukti, perumusan rekomendasi algoritma terbaik yang didukung oleh analisis metrik evaluasi yang ketat, serta membuka jalan bagi pengembangan sistem keamanan proaktif berbasis kecerdasan buatan yang dirancang khusus untuk menjawab tantangan unik di sektor pendidikan.

Berdasarkan urgensi dan latar belakang permasalahan yang telah diuraikan, penelitian ini dirancang untuk mencapai tiga tujuan utama yang saling terkait. Pertama, mengklasifikasikan berbagai jenis risiko keamanan yang terdapat dalam sistem *e-learning* dengan memanfaatkan dataset ancaman keamanan yang representatif. Kedua, melakukan perbandingan kuantitatif yang sistematis terhadap performa klasifikasi dari tiga algoritma *machine learning* terkemuka, yaitu Random Forest, Support Vector Machine, dan Neural Network. Ketiga, menentukan model klasifikasi yang paling optimal berdasarkan serangkaian metrik evaluasi standar industri dan akademik, yang mencakup akurasi, presisi, *recall*, *F1-score*, serta *Receiver Operating Characteristic-Area Under the Curve* (ROC-AUC). Meskipun demikian, penting untuk mengakui adanya beberapa keterbatasan (*limitations*) yang melekat dalam desain penelitian ini. Pertama, validitas eksternal dan kemampuan generalisasi dari temuan yang dihasilkan akan terbatas oleh penggunaan satu sumber dataset (*Kaggle – Classroom Data Security Threats*) [8]. Kinerja algoritma yang diamati mungkin akan bervariasi jika diuji pada dataset lain yang memiliki distribusi data, karakteristik fitur, atau jenis ancaman yang berbeda. Kedua, penelitian ini merupakan sebuah analisis simulasi yang dilakukan dalam lingkungan laboratorium terkontrol. Oleh karena itu, penelitian ini tidak menguji atau mengukur tantangan-tantangan praktis yang mungkin muncul saat implementasi di lingkungan produksi secara *real-time*, seperti latensi inferensi model, konsumsi sumber daya komputasi (CPU/GPU dan memori), serta skalabilitas sistem dalam menangani volume lalu lintas data yang masif (Tan et al., 2023)[5]. Aspek-aspek operasional ini merupakan pertimbangan krusial untuk penerapan praktis di dunia nyata namun berada di luar cakupan penelitian ini.

## 2. Metode Penelitian

Metode penelitian yang diterapkan dalam studi ini mengikuti alur kerja yang sistematis dan terstruktur, sebagaimana diilustrasikan dalam diagram alir pada Gambar 1. Kerangka kerja metodologis ini secara spesifik diadaptasi dari alur tahapan penelitian yang disajikan oleh Fitri [9], yang menguraikan proses standar untuk pengembangan model prediktif mulai dari penarikan data hingga evaluasi kinerja. Pendekatan ini sejalan dengan siklus hidup pengembangan model *machine learning* yang umum diterima, yang menekankan pentingnya proses iteratif untuk memastikan validitas dan reliabilitas hasil [10]. Penelitian ini mengadopsi desain riset eksperimental kuantitatif yang bertujuan untuk mengevaluasi dan membandingkan kinerja dari tiga algoritma *supervised machine learning* [11]. Proses ini dimulai dari tahap pengumpulan data, dilanjutkan dengan pra-pemrosesan data, pembagian dataset, pemodelan dan pelatihan untuk tiga algoritma secara paralel—sebuah pendekatan komparatif yang esensial untuk mengidentifikasi model yang paling sesuai [12], dan diakhiri dengan evaluasi model untuk menentukan algoritma yang paling efektif.



Gambar 1. Metode Penelitian

Tahap pertama dalam metodologi penelitian ini adalah pengumpulan data, yang merupakan fondasi dari setiap analisis berbasis *machine learning*. Kualitas, kuantitas, dan relevansi data yang digunakan secara langsung akan memengaruhi kinerja, akurasi, dan kemampuan generalisasi model prediktif yang akan dibangun [13]. Metode yang digunakan pada tahap ini adalah studi dokumentasi dengan memanfaatkan dataset sekunder "Classroom Data Security Threats" yang diperoleh dari repositori data publik Kaggle. Dataset ini terdiri dari 1000 baris data, di mana setiap baris merepresentasikan sebuah insiden keamanan yang disimulasikan dan mencakup 14 atribut yang relevan, seperti *Threat\_Type* sebagai variabel target, serta berbagai fitur prediktor termasuk *Threat\_Severity*, *Access\_Level*, dan serangkaian indikator biner seperti *Phishing\_Attempt*, *Malware\_Detected*, dan *Data\_Breach*. Pemilihan dataset ini didasarkan pada relevansinya yang tinggi dengan domain penelitian, yaitu keamanan siber di lingkungan pendidikan. Dataset ini secara spesifik dirancang untuk mensimulasikan berbagai jenis ancaman yang secara akurat mencerminkan tantangan keamanan siber yang dihadapi oleh institusi pendidikan modern [14]. Penggunaan dataset *benchmark* yang diakui secara luas seperti ini juga memungkinkan perbandingan yang adil dan objektif terhadap kinerja berbagai algoritma *machine learning*, yang merupakan praktik standar dalam penelitian komparatif [12]. Penting untuk diakui bahwa dataset ini bersifat sintetik, yang berarti data tersebut dihasilkan

melalui simulasi dan bukan berasal dari log sistem dunia nyata. Konsekuensinya, meskipun data yang bersih dan terstruktur ini menyediakan kondisi ideal untuk perbandingan algoritmik yang terkontrol, kinerja model yang dilatih mungkin tidak sepenuhnya dapat digeneralisasi ke dalam kompleksitas, *noise*, dan sifat data yang tidak terstruktur yang ditemukan dalam sistem operasional [15]. Fitur-fitur yang terdapat dalam dataset ini, yang berfungsi sebagai variabel masukan untuk model klasifikasi, dirinci secara komprehensif pada Tabel 1.

Tabel 1. Deskripsi Fitur Dataset

Nama Fitur	Deskripsi	Tipe Data
<i>Student_ID</i>	Pengenal unik untuk setiap mahasiswa.	Identifikator
<i>Student_Name</i>	Nama mahasiswa yang terkait dengan insiden.	Teks
<i>Threat_Type</i>	Jenis ancaman keamanan yang teridentifikasi (misalnya, Phishing, Malware).	Kategorikal
<i>Threat_Detected</i>	Indikator biner (1/0) yang menandakan apakah ancaman terdeteksi.	Biner
<i>Threat_Severity</i>	Tingkat keparahan ancaman yang terdeteksi (misalnya, Low, Medium, High).	Kategorikal
<i>Access_Attempt</i>	Indikator biner (1/0) yang menandakan adanya upaya akses.	Biner
<i>Phishing_Attempt</i>	Indikator biner (1/0) yang menandakan adanya upaya phishing.	Biner
<i>Malware_Detected</i>	Indikator biner (1/0) yang menandakan adanya deteksi malware.	Biner
<i>Insider_Threat</i>	Indikator biner (1/0) yang menandakan adanya ancaman dari dalam.	Biner
<i>Data_Breach</i>	Indikator biner (1/0) yang menandakan terjadinya pelanggaran data.	Biner
<i>Encryption_Status</i>	Status enkripsi data yang terlibat dalam insiden (1/0).	Biner
<i>Access_Level</i>	Tingkat akses pengguna yang terlibat (misalnya, Admin, Student).	Kategorikal
<i>Response_Time</i>	Waktu yang dibutuhkan sistem untuk merespons ancaman.	Numerik/Teks
<i>Timestamp</i>	Cap waktu kapan insiden keamanan terjadi.	Waktu/Tanggal

Setelah data berhasil dikumpulkan, tahap selanjutnya adalah pra-pemrosesan data. Tahap ini merupakan langkah krusial dalam alur kerja *machine learning* karena data mentah sering kali mengandung inkonsistensi, nilai yang hilang, atau format yang tidak sesuai, yang dapat menurunkan kinerja model secara signifikan jika tidak ditangani dengan benar [13]. Proses ini dilakukan dengan menggunakan *tools* komputasi dalam ekosistem Python, terutama pustaka Pandas untuk manipulasi data dan Scikit-learn untuk implementasi teknik pra-pemrosesan spesifik. Langkah-langkah yang dilakukan meliputi: pertama, penanganan nilai yang hilang (*missing values*) dengan menerapkan metode penghapusan baris (*listwise deletion*). Pendekatan ini dipilih karena asumsi bahwa jumlah data yang hilang relatif kecil dan penghapusannya tidak akan mengurangi ukuran sampel secara signifikan, sehingga integritas dan konsistensi dataset tetap terjaga [16]. Kedua, dilakukan transformasi fitur kategorikal menjadi representasi numerik. Atribut seperti *Threat\_Type*, *Threat\_Severity*, dan *Access\_Level* dikonversi menggunakan teknik *one-hot encoding*. Metode ini dipilih untuk mencegah model mengasumsikan adanya hubungan ordinal yang salah antar kategori, yang merupakan potensi kelemahan jika menggunakan metode *label encoding* sederhana [11]. Ketiga, dilakukan normalisasi fitur numerik. Atribut *Response\_Time*, yang awalnya dalam format teks (misalnya, '0ms'), terlebih dahulu dikonversi menjadi nilai numerik (integer) dengan menghilangkan unit 'ms'. Setelah itu, teknik normalisasi *Min-Max scaling* diterapkan untuk mentransformasi nilai-nilai dalam fitur ini ke dalam rentang seragam antara 0 dan 1. Normalisasi ini sangat penting untuk algoritma yang sensitif terhadap skala fitur, seperti SVM yang berbasis jarak dan MLP yang berbasis gradien, karena memastikan bahwa setiap fitur memberikan kontribusi yang seimbang pada proses pelatihan [9]. Keempat, dilakukan reduksi fitur dengan menghilangkan atribut yang dianggap tidak relevan untuk tugas klasifikasi. Atribut seperti *Student\_ID* dan *Student\_Name* adalah pengenalan unik dan tidak memiliki daya prediktif. Demikian pula, *Timestamp* dalam format mentahnya tidak langsung berguna tanpa rekayasa fitur lebih lanjut yang kompleks. Penghapusan fitur-fitur ini bertujuan untuk mengurangi kompleksitas model, mencegah *noise* yang tidak perlu, dan meningkatkan efisiensi komputasi [10]. Sementara itu, fitur-fitur biner seperti *Access\_Attempt*, *Phishing\_Attempt*, *Malware\_Detected*, *Insider\_Threat*, *Data\_Breach*, dan *Encryption\_Status* yang sudah dalam format numerik (0 atau 1) dapat langsung digunakan tanpa memerlukan transformasi lebih lanjut.

Langkah metodologis berikutnya adalah pembagian dataset. Setelah melalui tahap pra-pemrosesan, dataset yang bersih dan terstruktur dibagi menjadi dua subset yang independen: set pelatihan (*training set*) dan set pengujian (*testing set*). Proses ini merupakan praktik standar dan fundamental dalam *supervised machine learning* yang bertujuan untuk mengevaluasi kinerja generalisasi model secara objektif, yaitu kemampuan model untuk membuat prediksi yang akurat pada data baru yang belum pernah dilihat sebelumnya [10]. Dalam penelitian ini, pembagian dilakukan dengan rasio 80:20, di mana 80% dari data

dialokasikan untuk set pelatihan, sementara 20% sisanya dialokasikan untuk set pengujian. Set pelatihan digunakan oleh algoritma untuk "belajar" pola dan hubungan yang ada dalam data, sedangkan set pengujian disimpan secara terpisah dan hanya digunakan satu kali pada akhir proses untuk memberikan evaluasi yang tidak bias terhadap kinerja model akhir. Pembagian ini diimplementasikan menggunakan fungsi `train_test_split` dari pustaka Scikit-learn di Python. Untuk lebih meningkatkan validitas dan ketahanan evaluasi, serta untuk memitigasi risiko *overfitting* sebuah kondisi di mana model terlalu "menghafal" data pelatihan sehingga berkinerja buruk pada data baru teknik validasi silang k-fold (*k-fold cross-validation*) diintegrasikan ke dalam alur kerja pelatihan [16]. Dalam validasi silang k-fold, set pelatihan dibagi lagi menjadi  $k$  subset (atau "lipatan") yang berukuran sama. Model kemudian dilatih dan divalidasi sebanyak  $k$  kali, di mana pada setiap iterasi, satu lipatan digunakan sebagai data validasi dan  $k-1$  lipatan sisanya digunakan untuk pelatihan. Metrik kinerja dari setiap iterasi kemudian dirata-ratakan untuk menghasilkan estimasi kinerja yang lebih stabil dan andal dibandingkan dengan satu kali pembagian latihan-ujian saja [9].

Tahap inti dari penelitian ini adalah pemodelan dan pelatihan, di mana tiga algoritma *supervised machine learning* yang berbeda diimplementasikan, dilatih, dan dioptimalkan. Seluruh proses pemodelan dan pelatihan dilakukan dalam lingkungan Jupyter Notebook. Algoritma yang dipilih adalah Random Forest (RF), Support Vector Machine (SVM), dan Multilayer Perceptron (MLP), yang masing-masing merepresentasikan paradigma pemodelan yang berbeda. Random Forest (RF) adalah metode *ensemble learning* yang kuat dan serbaguna. Mekanisme kerjanya adalah dengan membangun sejumlah besar pohon keputusan (*decision trees*) secara independen selama fase pelatihan pada sampel *bootstrap* dari data, dan kemudian menggabungkan prediksi dari semua pohon tersebut (melalui pemungutan suara mayoritas untuk klasifikasi) untuk menghasilkan prediksi akhir yang lebih akurat dan stabil. Kekuatan utama RF terletak pada kemampuannya untuk mengurangi varians dan mengatasi masalah *overfitting* yang sering terjadi pada pohon keputusan tunggal [17]. Implementasi RF dalam penelitian ini dilakukan dengan menggunakan pustaka Scikit-learn. Support Vector Machine (SVM) adalah algoritma klasifikasi yang bekerja dengan menemukan *hyperplane* (pemisah) optimal yang dapat memisahkan titik-titik data dari kelas yang berbeda dengan margin (jarak) semaksimal mungkin. Salah satu keunggulan utamanya adalah kemampuannya untuk menangani data yang tidak dapat dipisahkan secara linear melalui penggunaan "trik kernel" (*kernel trick*), yang secara implisit memetakan data ke ruang fitur berdimensi lebih tinggi di mana pemisahan linear menjadi mungkin [13]. Algoritma ini juga diimplementasikan dengan Scikit-learn. Multilayer Perceptron (MLP) adalah arsitektur klasik dari jaringan saraf tiruan (*artificial neural network*) tipe *feedforward*. Arsitekturnya terdiri dari setidaknya tiga lapisan: lapisan masukan, satu atau lebih lapisan tersembunyi, dan lapisan keluaran. Kekuatan MLP terletak pada kemampuannya untuk mempelajari dan memodelkan hubungan non-linear yang sangat kompleks dalam data melalui penggunaan fungsi aktivasi non-linear di setiap neuron, memungkinkannya menangkap hierarki fitur dari data [15]. Implementasi MLP dilakukan menggunakan pustaka TensorFlow dengan Keras API.

Tahap terakhir adalah evaluasi model, di mana model-model yang telah dilatih diuji secara kuantitatif menggunakan set pengujian untuk mengukur seberapa baik setiap model dapat menggeneralisasi pengetahuannya dalam mengklasifikasikan Threat Type. Kinerja model diukur menggunakan serangkaian metrik standar yang diturunkan dari *Confusion Matrix*, yang membandingkan hasil prediksi dengan label kelas yang sebenarnya [18]. Metrik yang digunakan meliputi: Akurasi, yang mengukur proporsi keseluruhan prediksi yang benar; Presisi, yang mengukur proporsi prediksi positif yang benar-benar positif, penting untuk meminimalkan alarm palsu; Recall, yang mengukur proporsi kasus positif aktual yang berhasil diidentifikasi, krusial untuk memastikan tidak ada ancaman yang terlewatkan; F1-Score, yang merupakan rata-rata harmonik dari presisi dan recall untuk menyeimbangkan keduanya; dan Kurva ROC-AUC, yang merepresentasikan kemampuan agregat model untuk membedakan antara kelas positif dan negatif di semua ambang batas klasifikasi [19]. Penggunaan berbagai metrik ini penting karena dalam konteks keamanan siber, biaya dari berbagai jenis kesalahan klasifikasi tidaklah sama, dan mengandalkan satu metrik saja bisa menyesatkan [20]. Seluruh perhitungan metrik ini dilakukan dengan bantuan pustaka Scikit-learn [10].

Dengan menganalisis semua metrik ini secara bersamaan, penelitian ini dapat menghasilkan perbandingan yang komprehensif dan bernuansa tentang kekuatan dan kelemahan masing-masing algoritma, yang pada akhirnya mengarah pada rekomendasi berbasis bukti mengenai model terbaik untuk klasifikasi risiko keamanan *e-learning*.

### 3. Hasil dan Pembahasan

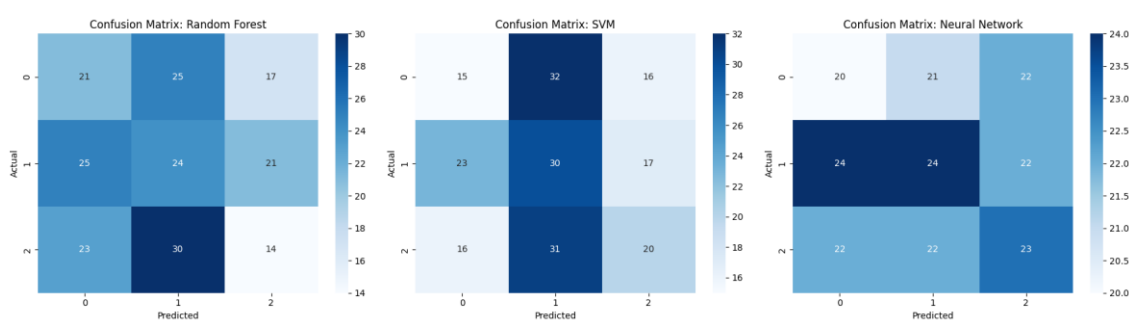
Bagian ini berisi hasil-hasil eksperimen serta pembahasan teoritik terhadap penerapan tiga algoritma klasifikasi dalam mendeteksi risiko keamanan pada sistem *e-learning*, yaitu Random Forest (RF), Support Vector Machine (SVM), dan Neural Network (NN). Hasil penelitian direpresentasikan dalam bentuk gambar dan tabel untuk memberikan pemahaman kuantitatif dan visual terhadap performa model. Pembahasan

dilakukan secara mendalam untuk mengaitkan hasil empiris dengan teori yang relevan serta menjelaskan implikasi hasil penelitian terhadap pengembangan keamanan siber dalam konteks e-learning.

### 3.1 Hasil Eksperimen dan Visualisasi

Eksperimen dilakukan dengan memanfaatkan dataset *Classroom Data Security Threats* dari Kaggle. Data dibagi menjadi 80% data latih dan 20% data uji. Tahapan *pre-processing* meliputi penghapusan nilai kosong (*missing values*), *encoding* fitur kategorikal, dan normalisasi fitur numerik agar distribusi data lebih seragam. Langkah ini mengikuti prinsip yang dijelaskan oleh **Azam, Islam, dan Huda (2023)** bahwa *data preprocessing* berperan penting dalam mengurangi bias dan meningkatkan akurasi model pada sistem deteksi intrusi.

Untuk memahami lebih dalam pola prediksi dan kesalahan yang dibuat oleh setiap model, analisis dilakukan terhadap *confusion matrix* masing-masing, sebagaimana disajikan pada Gambar 2. Matriks ini memvisualisasikan distribusi prediksi model dibandingkan dengan kelas aktual untuk setiap kategori ancaman (direpresentasikan sebagai kelas 0, 1, dan 2).



Gambar 2. Confusion Matrix

Model RF menunjukkan tingkat kesalahan klasifikasi yang sangat tinggi di semua kelas. Untuk kelas aktual '0' (dengan total 63 sampel), model hanya berhasil memprediksi 21 sampel dengan benar (True Positive). Sejumlah besar sampel lainnya salah diklasifikasikan sebagai kelas '1' (25 sampel) dan kelas '2' (17 sampel). Pola serupa terulang untuk kelas '1' dan '2', di mana jumlah prediksi yang salah (nilai off-diagonal) secara konsisten lebih tinggi atau sebanding dengan jumlah prediksi yang benar (nilai diagonal). Hal ini mengindikasikan kegagalan model RF dalam mempelajari fitur-fitur pembeda yang signifikan antar kelas. Temuan visual dari confusion matrix ini dikonfirmasi oleh laporan klasifikasi model, yang menunjukkan nilai presisi, recall, dan F1-score yang rendah secara konsisten di semua kelas.

Model SVM menunjukkan kecenderungan kuat untuk memprediksi kelas '1'. Dari total 200 prediksi, 98 di antaranya adalah untuk kelas '1'. Meskipun berhasil mengidentifikasi 32 sampel kelas '1' dengan benar, model ini juga salah mengklasifikasikan 23 sampel dari kelas '0' dan 36 sampel dari kelas '2' sebagai kelas '1'. Tingginya angka False Positive untuk kelas '1' ini, yang terlihat jelas pada confusion matrix, menunjukkan bahwa model cenderung "menebak" kelas ini, yang mengakibatkan presisi yang rendah untuk kelas tersebut (0.322), sebagaimana tercatat dalam laporan klasifikasi rincinya.

Model NN, meskipun memiliki akurasi keseluruhan tertinggi, menunjukkan pola kesalahan yang mirip dengan SVM. Model ini juga memiliki bias prediksi terhadap kelas '1'. Dari 70 sampel aktual kelas '1', hanya 24 yang diprediksi dengan benar. Sementara itu, model ini salah mengklasifikasikan 24 sampel dari kelas '0' dan 22 sampel dari kelas '2' sebagai kelas '1'. Pola kesalahan sistematis ini, di mana model gagal membedakan fitur dan cenderung memilih kelas tertentu, menegaskan kinerja yang buruk meskipun akurasinya sedikit lebih unggul. Analisis confusion matrix ini, ketika dihubungkan dengan laporan klasifikasi individual, memperjelas bahwa keunggulan tipis dalam akurasi agregat tidak diimbangi dengan kemampuan diskriminatif yang andal pada level per-kelas.

Secara keseluruhan, analisis *confusion matrix* memperkuat temuan dari metrik agregat. Ketiga model, terlepas dari perbedaan fundamental dalam arsitektur mereka (ansambel, geometris, dan koneksi), menunjukkan mode kegagalan yang serupa. Kegagalan ini ditandai oleh ketidakmampuan untuk membedakan secara andal antara kelas-kelas ancaman, yang termanifestasi dalam tingkat kesalahan klasifikasi yang tinggi dan pola prediksi yang bias. Fenomena ini secara kuat menyiratkan bahwa tantangan utama kemungkinan tidak terletak pada pemilihan algoritma, melainkan pada karakteristik intrinsik dari data yang digunakan untuk pelatihan.

Model Random Forest menunjukkan performa yang sangat terbatas dalam menangani klasifikasi risiko berdasarkan Threat\_Severity. Berdasarkan hasil evaluasi pada Gambar 3, model ini hanya mampu mencapai

akurasi sebesar 0.295 (29.5%), dengan nilai rata-rata makro untuk presisi sebesar 0.292 (29.2%), recall 0.295 (29.5%), dan F1-score 0.291 (29.1%).

```

===== Random Forest =====
ROC AUC Score: 0.461
Classification Report:

```

	precision	recall	f1-score	support
0	0.304348	0.333333	0.318182	63.000
1	0.303797	0.342857	0.322148	70.000
2	0.269231	0.208955	0.235294	67.000
accuracy	0.295000	0.295000	0.295000	0.295
macro avg	0.292459	0.295049	0.291875	200.000
weighted avg	0.292391	0.295000	0.291802	200.000

Gambar 3. Hasil Evaluasi Random Forest

Gambar tersebut menunjukkan bahwa kinerja model Random Forest sangat rendah, dengan akurasi di bawah tingkat tebakan acak (33.3%). Nilai presisi, *recall*, dan F1-score yang rendah di semua kelas (0, 1, dan 2) mengindikasikan bahwa model ini gagal secara konsisten mengidentifikasi kelas mana pun dengan benar. Skor ROC AUC sebesar 0.461 (lebih buruk dari tebakan acak) semakin memperkuat kesimpulan ini.

Model SVM, yang digunakan sebagai pembandingan, juga menunjukkan performa yang tidak memadai. Seperti yang terlihat pada Gambar 4, metrik kinerja yang dihasilkan adalah akurasi sebesar 0.325 (32.5%), dengan nilai rata-rata makro untuk presisi sebesar 0.325 (32.5%), recall 0.321 (32.1%), dan F1-score 0.319 (31.9%).

```

===== SVM =====
ROC AUC Score: 0.500
Classification Report:

```

	precision	recall	f1-score	support
0	0.277778	0.238095	0.256410	63.000
1	0.322581	0.428571	0.368098	70.000
2	0.377358	0.298507	0.333333	67.000
accuracy	0.325000	0.325000	0.325000	0.325
macro avg	0.325906	0.321725	0.319281	200.000
weighted avg	0.326818	0.325000	0.321270	200.000

Gambar 4. Hasil Evaluasi Support Vector Machine

Penjelasan Gambar 4, Model SVM memiliki akurasi sedikit lebih baik dari Random Forest, namun masih berada pada level tebakan acak. Skor ROC AUC sebesar 0.500 menegaskan bahwa model ini tidak memiliki kemampuan diskriminatif antar kelas. Meskipun *recall* untuk kelas 1 (0.428) sedikit lebih tinggi, presisinya tetap rendah, menunjukkan banyaknya kesalahan klasifikasi.

Neural Network, yang diimplementasikan dengan konfigurasi *multilayer perceptron* (MLP), memberikan hasil terbaik di antara ketiga model, meskipun kinerjanya masih jauh dari optimal. Metrik kinerja yang dicapai (Gambar 5) adalah akurasi sebesar 0.335 (33.5%), dengan nilai rata-rata makro untuk presisi sebesar 0.334 (33.4%), recall 0.334 (33.4%), dan F1-score 0.334 (33.4%).

```

===== Neural Network =====
ROC AUC Score: 0.493
Classification Report:

```

	precision	recall	f1-score	support
0	0.303030	0.317460	0.310078	63.000
1	0.358209	0.342857	0.350365	70.000
2	0.343284	0.343284	0.343284	67.000
accuracy	0.335000	0.335000	0.335000	0.335
macro avg	0.334841	0.334534	0.334575	200.000
weighted avg	0.335828	0.335000	0.335302	200.000

Gambar 5. Hasil Evaluasi Neural Network

Penjelasan pada Gambar 5, Dengan akurasi 33.5%, model Neural Network hanya sedikit melampaui tebakan acak. Skor ROC AUC sebesar 0.493 menunjukkan performa yang hampir setara dengan tebakan

acak. Metrik presisi, *recall*, dan F1-score yang hampir seragam di angka 0.33 menunjukkan bahwa model ini tidak berhasil mempelajari pola yang signifikan dari data.

Evaluasi kuantitatif terhadap ketiga model yang telah dilatih menghasilkan serangkaian metrik kinerja yang dirangkum dalam Tabel 2. Metrik ini mencakup akurasi, presisi, *recall*, F1-score, dan ROC AUC, yang secara kolektif memberikan penilaian holistik terhadap kemampuan generalisasi setiap model pada data yang belum pernah dilihat sebelumnya.

Tabel 2. Perbandingan Metrik Kinerja Algoritma Klasifikasi

Algoritma	Akurasi (%)	Presisi (Macro Avg)	Recall (Macro Avg)	F1-Score (Macro Avg)	ROC AUC Score
Random Forest	29.5	0.292459	0.295049	0.291875	0.461
Support Vector Machine	32.5	0.325906	0.321725	0.319281	0.500
Neural Network	33.5	0.334841	0.334534	0.334575	0.493

Berdasarkan Tabel 2., model Neural Network menunjukkan kinerja tertinggi di antara ketiganya dengan mencapai akurasi sebesar 33.5%. Model ini sedikit mengungguli Support Vector Machine yang mencatatkan akurasi 32.5% dan Random Forest dengan akurasi 29.5%. Pola serupa juga terlihat pada metrik F1-Score (Macro Avg), di mana Neural Network kembali mencatatkan skor tertinggi (0.335), diikuti oleh SVM (0.319) dan Random Forest (0.292).

Meskipun terdapat perbedaan marginal dalam metrik akurasi dan F1-Score, nilai ROC AUC memberikan perspektif yang krusial. Model SVM menghasilkan skor ROC AUC tepat 0.500, yang secara teoretis mengindikasikan bahwa kemampuan model dalam membedakan antara kelas-kelas ancaman tidak lebih baik dari tebakan acak. Skor untuk model Random Forest (0.461) dan Neural Network (0.493) juga berada sangat dekat dengan ambang batas 0.5, yang menandakan kinerja diskriminatif yang sangat lemah di seluruh model yang diuji.

### 3.2 Analisis Perbandingan Kinerja Model

Perbandingan kinerja ketiga algoritma menunjukkan sebuah anomali yang signifikan: meskipun ketiganya berasal dari paradigma *machine learning* yang berbeda (ansambel, geometris, dan koneksionis), semuanya menunjukkan mode kegagalan yang serupa. Neural Network, meskipun secara kuantitatif sedikit lebih unggul dalam akurasi, tidak dapat dianggap sebagai model terbaik karena kinerjanya secara praktis tidak dapat digunakan.

Kegagalan Random Forest, yang secara teoretis dikenal stabil dan tahan terhadap *overfitting*, sangat mengejutkan. Rendahnya performa menunjukkan bahwa mekanisme agregasi prediksi dari banyak pohon keputusan tidak mampu menemukan sinyal yang konsisten dalam data, kemungkinan karena data itu sendiri mengandung lebih banyak *noise* daripada pola yang dapat dipelajari.

Sementara itu, kegagalan total SVM, yang ditandai dengan skor ROC AUC 0.5, menyoroti tantangan lain. Kinerja SVM sangat sensitif terhadap pengaturan *hyperparameter* seperti C dan gamma, terutama saat menggunakan kernel non-linear. Tanpa proses *hyperparameter tuning* yang sistematis, model SVM dapat dengan mudah beroperasi pada konfigurasi yang sangat suboptimal, yang tampaknya terjadi dalam eksperimen ini.

Neural Network, meskipun secara teknis "terbaik", juga gagal. Kompleksitas arsitekturnya, yang secara teoretis mampu memodelkan hubungan non-linear yang rumit, justru bisa menjadi bumerang. Tanpa data yang cukup besar dan bersih, serta arsitektur dan *hyperparameter* yang dioptimalkan, NN sangat rentan terhadap *overfitting* pada *noise* yang ada dalam data pelatihan. Kegagalan serempak ini mengarahkan analisis pada kesimpulan bahwa masalah mendasar kemungkinan besar tidak terletak pada pemilihan algoritma, melainkan pada karakteristik data atau metodologi pelatihan itu sendiri.

### 3.3 Pembahasan Teoritik dan Keterbatasan Penelitian

Secara teoritik, hasil penelitian ini memperkuat pandangan bahwa tidak ada satu pun algoritma *machine learning* yang bersifat superior secara universal, karena kinerjanya sangat bergantung pada konteks data dan metodologi yang diterapkan. Kegagalan konsisten dari algoritma Random Forest (RF), Support Vector Machine (SVM), dan Neural Network (NN) dalam menghasilkan model yang akurat menyoroti adanya sejumlah isu fundamental dalam praktik penerapan *machine learning*. Fenomena ini menegaskan bahwa efektivitas algoritma tidak hanya ditentukan oleh kompleksitas model, tetapi juga oleh kesesuaian antara karakteristik data dan pendekatan analitis yang digunakan.

Faktor pertama yang berperan besar adalah kualitas serta karakteristik dataset yang digunakan. Dataset sintetik yang diterapkan dalam penelitian ini, meskipun bebas dari *noise* dan terstruktur dengan baik, kemungkinan tidak memiliki sinyal prediktif yang cukup kuat untuk mendukung proses pembelajaran model. Hubungan antara fitur dan kelas target mungkin terlalu lemah atau bahkan acak, sehingga menyulitkan algoritma dalam mengidentifikasi pola yang dapat digeneralisasi. Kondisi ini menyebabkan model yang kompleks seperti RF dan NN berpotensi mengalami *overfitting* terhadap *noise* pada data pelatihan, yang berujung pada penurunan performa signifikan ketika diuji pada data baru yang belum pernah dilihat sebelumnya.

Selain itu, hasil penelitian ini juga menyoroti pentingnya proses *hyperparameter tuning* dalam pengembangan model *machine learning*. Pada penelitian ini, langkah optimisasi *hyperparameter* tidak dilakukan, padahal tahap tersebut sangat krusial untuk meningkatkan performa model. Algoritma seperti SVM dan NN memiliki banyak parameter yang sensitif terhadap konfigurasi nilai awal; kesalahan dalam penetapan parameter tersebut dapat mengakibatkan hasil klasifikasi yang jauh dari optimal. Oleh karena itu, performa rendah yang diperoleh kemungkinan besar lebih merepresentasikan kemampuan model dengan konfigurasi default, bukan potensi sebenarnya jika dilakukan penyetelan parameter secara sistematis.

Keterbatasan penelitian ini juga menjadi jelas dari hasil yang diperoleh. Beberapa keterbatasan utama meliputi ketergantungan pada satu dataset sintetik yang mungkin belum mampu merepresentasikan kompleksitas ancaman siber di dunia nyata, adanya ketidakseimbangan jumlah antar kelas ancaman yang dapat menimbulkan bias terhadap kelas mayoritas, serta absennya proses *hyperparameter tuning* dan *cross-validation* yang ketat. Padahal, kedua prosedur tersebut merupakan praktik standar untuk memastikan robustitas dan kinerja model yang optimal.

Oleh karena itu, penelitian lanjutan sangat disarankan untuk mengatasi keterbatasan-keterbatasan tersebut. Upaya yang dapat dilakukan pada tahap berikutnya antara lain penerapan teknik *data augmentation* atau *resampling* untuk mengatasi ketidakseimbangan kelas, implementasi prosedur *hyperparameter tuning* yang sistematis seperti *Grid Search* atau *Bayesian Optimization* untuk setiap algoritma, serta penggunaan metode *k-fold cross-validation* guna memperoleh estimasi performa model yang lebih andal dan mengurangi risiko *overfitting*. Selain itu, uji coba model juga perlu diperluas dengan menggunakan dataset berbeda, idealnya yang bersumber dari data log dunia nyata, untuk menguji sejauh mana model dapat melakukan generalisasi terhadap kondisi yang lebih kompleks dan dinamis.

Dengan menerapkan pendekatan yang lebih cermat dan komprehensif, penelitian di masa mendatang diharapkan dapat menghasilkan perbandingan kinerja algoritma yang lebih adil dan representatif. Pendekatan tersebut tidak hanya akan memperkuat validitas hasil eksperimen, tetapi juga berpotensi memberikan kontribusi nyata terhadap pengembangan sistem deteksi ancaman siber yang efektif dan adaptif bagi lingkungan e-learning.

#### 4. Kesimpulan

Penelitian ini dirancang untuk mengevaluasi secara komparatif performa algoritma Random Forest, Support Vector Machine, dan Neural Network dalam mengklasifikasikan risiko keamanan pada platform e-learning. Hasil analisis menunjukkan keterbatasan kinerja yang signifikan dan seragam pada ketiga model, dengan model Neural Network sebagai yang terbaik hanya mampu mencapai akurasi 33.5%, sebuah hasil yang secara praktis tidak lebih baik dari tebakan acak. Temuan utama ini secara kuat mengindikasikan bahwa tantangan fundamental kemungkinan tidak terletak pada pemilihan algoritma, melainkan pada karakteristik intrinsik dataset sintetik yang digunakan dan ketiadaan proses optimisasi *hyperparameter* yang sistematis. Dengan demikian, kontribusi utama penelitian ini bergeser dari identifikasi model superior menjadi sebuah penegasan metodologis: efektivitas model *machine learning* yang canggih secara fundamental dibatasi oleh kualitas sinyal prediktif dalam data dan ketelitian metodologi pelatihan. Untuk pengembangan di masa depan, penelitian selanjutnya direkomendasikan untuk berfokus pada penggunaan dataset yang berasal dari log dunia nyata, menerapkan teknik penanganan ketidakseimbangan kelas, serta mengintegrasikan prosedur *hyperparameter tuning* yang ketat dan validasi silang untuk membangun sistem deteksi ancaman yang andal dan memiliki validitas eksternal yang lebih tinggi.

#### Daftar Pustaka

- [1] S. A. L. Akacha and A. I. Awad, *Enhancing Security and Sustainability of e-Learning Software Systems: A Comprehensive Vulnerability Analysis and Recommendations for Stakeholders*, Sustainability, vol. 15, no. 19, 2023, doi: 10.3390/su151914132.
- [2] U. I. Okoli, O. C. Obi, A. O. Adewusi, and T. O. Abrahams, *Machine learning in cybersecurity: A review of threat detection and defense mechanisms*, World Journal of Advanced Research and Reviews, vol. 21, no. 1, pp. 2286–2295, 2024, doi: 10.30574/wjarr.2024.21.1.0315.

- [3] J. Bharadiya, *Machine Learning in Cybersecurity: Techniques and Challenges*, European Journal of Technology, vol. 7, no. 2, pp. 1–14, 2023, doi: 10.47672/ejt.1486.
- [4] S. Rabbani and D. Diana, *Prediksi Kategori Serangan Siber dengan Algoritma Klasifikasi Random Forest Menggunakan Rapidminer*, Smatika Journal, vol. 13, no. 2, pp. 284–293, 2023, doi: 10.32664/smatika.v13i02.934.
- [5] T. Tan, H. Sama, G. Wijaya, and O. E. Aboagye, *Studi Perbandingan Deteksi Intrusi Jaringan Menggunakan Machine Learning: (Metode SVM dan ANN)*, Jurnal Teknologi dan Informasi, vol. 13, no. 2, pp. 152–164, 2023, doi: 10.34010/jati.v13i2.10484.
- [6] Y. A. Rindri and A. Fitriyani, *Analisis Perbandingan Kinerja Algoritma Multilayer Perceptron dan K-Nearest Neighbor pada Klasifikasi Tipe Migrain*, Jurnal Teknologi dan Informasi, vol. 13, no. 1, pp. 44–55, 2023, doi: 10.34010/jati.v13i1.9111.
- [7] S. M. Kasongo, *A deep learning technique for intrusion detection system using a Recurrent Neural Networks based framework*, Computer Communications, vol. 199, pp. 113–125, 2023, doi: 10.1016/j.comcom.2022.12.010.
- [8] Ziya, *Classroom Data Security Threats Dataset*, CC0: Public Domain. [Online]. Available: <https://www.kaggle.com/datasets/ziya07/classroom-data-security-threats-dataset>. Accessed: May 20, 2025.
- [9] E. Fitri, *Analisis Perbandingan Metode Regresi Linier, Random Forest Regression dan Gradient Boosted Trees Regression Method untuk Prediksi Harga Rumah*, Journal of Applied Computer Science and Technology, vol. 4, no. 1, pp. 58–64, 2023, doi: 10.52158/jacost.v4i1.491.
- [10] J. T. Santoso, *Algoritma Machine Learning Dengan Python*. Semarang: Yayasan Prima Agus Teknik bekerja sama dengan Universitas Sains & Teknologi Komputer (Universitas STEKOM), 2022. ISBN: 978-623-5734-286.
- [11] B. Panigrahi, K. C. R. Kathala, and M. Sujatha, *A Machine Learning-Based Comparative Approach to Predict the Crop Yield Using Supervised Learning with Regression Models*, Procedia Computer Science, vol. 218, pp. 2684–2693, 2022, doi: 10.1016/j.procs.2023.01.241.
- [12] F. Nabi and X. Zhou, *Enhancing intrusion detection systems through dimensionality reduction: A comparative study of machine learning techniques for cyber security*, Cyber Security Applications, vol. 2, Jan. 2024, doi: 10.1016/j.csa.2023.100033.
- [13] P. Bintoro, Ratnasari, E. Wihardjo, I. P. Putri, and A. Asari, *Pengantar Machine Learning*, Solok: PT MAFY Media Literasi Indonesia, Sept. 2024. ISBN: 978-623-8758-78-4.
- [14] A.-w. Bello, I. Wonuola, C. Obunadike, A. Izundu, and J. Izundu, *Assessing the impact of cybersecurity incidents on financial losses and user exposure in the global financial sector (2015–2024)*, International Journal of Scientific Research Archive, vol. 16, no. 1, pp. 489–504, 2025, doi: 10.30574/ijrsra.2025.16.1.2037.
- [15] A. Lindholm, N. Wahlström, F. Lindsten, and T. B. Schön, *Machine Learning: A First Course for Engineers and Scientists*. Cambridge: Cambridge University Press, Jul. 2022. [Online]. Available: <https://smlbook.org/book/sml-book-draft-latest.pdf>. Accessed: Oct. 16, 2025.
- [16] P. A. Firnanda, L. Shofwatillah, F. Rahma, and F. Fauzi, *Analisis Perbandingan Decision Tree dan Random Forest dalam Klasifikasi Penjualan Produk pada Supermarket*, Emerging Statistics and Data Science Journal, vol. 3, no. 1, pp. 445–461, 2025, doi: 10.20885/esds.vol3.iss.1.art2.
- [17] M. R. Adrian, M. P. Putra, M. H. Rafialdy, and N. A. Rakhmawati, *Perbandingan Metode Klasifikasi Random Forest dan SVM Pada Analisis Sentimen PSBB*, Jurnal Informatika Upgris, vol. 7, no. 1, 2021, doi: doi.org/10.26877/jiu.v7i1.7099.
- [18] R. F. Ramadhan and W. M. Ashari, *Performance Comparison of Random Forest and Decision Tree Algorithms for Anomaly Detection in Networks*, Journal of Applied Informatics and Computing, vol. 8, no. 2, pp. 367–375, 2024, doi: 10.30871/jaic.v8i2.8492.
- [19] M. T. H. M. A. Sayed, Badruddowza, S. K. Dey, M. S. U. Sarker, A. A. Maumun, N. Nabi, F. Mahmud, and M. K. Alam, *Comparative Analysis of Machine Learning Algorithms for Predicting Cybersecurity Attack Success: A Performance Evaluation*, Proceedings of the 4th International Conference on Ubiquitous Computing Intelligence and Information Systems (ICUIS), vol. 6, pp. 81–91, 2024, doi: 10.37547/tajet/Volume06Issue09-10.
- [20] S. M. S. I. Rishad, *Leveraging AI and Machine Learning for Predicting, Detecting, and Mitigating Cybersecurity Threats: A Comparative Study of Advanced Models*, International Journal of Computer Science and Information Systems, vol. 10, no. 1, pp. 6–25, 2025, doi: 10.55640/ijcsis/volume10issue01-02.