

PERBANDINGAN RANDOM FOREST, K-NEAREST NEIGHBOR DAN SUPPORT VECTOR MACHINE UNTUK DETEKSI ANOMALI PADA CICIDS2017

Siti Haliza Zamili^{1*}, Dedy Kiswanto², Khodotun Hadawiyah Margolang³

¹²³Program Studi Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam,
Universitas Negeri Medan, Medan, Indonesia

e-mail: ¹sitihalizazamili08@gmail.com, ²dedykiswanto@unimed.ac.id

³khodotunhadawiyah@gmail.com

ABSTRAK

Perkembangan ancaman siber yang semakin kompleks mendorong kebutuhan akan sistem perlindungan jaringan yang mampu mendeteksi intrusi secara cerdas dan otomatis. Pendekatan berbasis machine learning telah banyak diadopsi sebagai solusi deteksi intrusi karena kemampuannya mengenali pola serangan tanpa bergantung pada aturan yang didefinisikan secara manual. Studi ini mengkaji efektivitas empat algoritma pembelajaran mesin, yakni Random Forest, Decision Tree, Stacked LSTM, dan AdaBoost, dalam tugas klasifikasi trafik jaringan menggunakan dataset CICIDS2017 sebagai tolok ukur evaluasi. Sebelum pelatihan model, data diproses melalui tahap normalisasi dan pemilihan fitur yang relevan guna memastikan representasi data yang optimal. Pengukuran kinerja model dilakukan menggunakan empat indikator evaluasi utama, yaitu akurasi, presisi, recall, dan F1-score, untuk memperoleh penilaian yang menyeluruh terhadap kemampuan klasifikasi masing-masing algoritma. Dari hasil pengujian, Decision Tree tampil menonjol dengan nilai presisi mencapai 99,87%, recall 99,59%, dan F1-score 99,32%, menandakan kemampuan klasifikasi per kelas yang sangat baik dengan tingkat kesalahan yang sangat rendah. Di sisi lain, Random Forest mencatatkan nilai akurasi keseluruhan tertinggi sebesar 99,90% disertai presisi 97,78%, recall 97,08%, dan F1-score 97,41%, yang mencerminkan keandalan model dalam mempertahankan performa secara konsisten di seluruh kelas. Kedua model berbasis Stacked LSTM dan AdaBoost menunjukkan performa yang lebih rendah dibandingkan pendekatan berbasis pohon dalam skenario pengujian yang sama. Secara keseluruhan, hasil penelitian ini menegaskan bahwa algoritma berbasis pohon keputusan menawarkan kombinasi akurasi tinggi dan stabilitas klasifikasi yang unggul, menjadikannya pilihan yang tepat untuk implementasi sistem deteksi intrusi jaringan berbasis pembelajaran mesin.

Kata kunci: random forest; k-nearest neighbor; support vector machine; deteksi anomali; intrusion detection system

ABSTRACT

The increasingly complex development of cyber threats has driven the need for network protection systems capable of detecting intrusions intelligently and automatically. Machine learning-based approaches have been widely adopted as intrusion detection solutions due to their ability to recognize attack patterns without relying on manually defined rules. This study examines the effectiveness of four machine learning algorithms, namely Random Forest, Decision Tree, Stacked LSTM, and AdaBoost, in network traffic classification tasks using the CICIDS2017

dataset as the evaluation benchmark. Before model training, the data was processed through normalization and relevant feature selection stages to ensure optimal data representation. Model performance was measured using four main evaluation metrics, namely accuracy, precision, recall, and F1-score, to obtain a comprehensive assessment of the classification capability of each algorithm. From the test results, Decision Tree stood out with a precision value reaching 99.87%, recall of 99.59%, and F1-score of 99.32%, indicating excellent per-class classification capability with a very low error rate. On the other hand, Random Forest recorded the highest overall accuracy value of 99.90%, accompanied by a precision of 97.78%, recall of 97.08%, and F1-score of 97.41%, reflecting the model's reliability in consistently maintaining performance across all classes. Both the Stacked LSTM and AdaBoost-based models showed lower performance compared to tree-based approaches in the same testing scenario. Overall, the results of this study confirm that decision tree-based algorithms offer a combination of high accuracy and superior classification stability, making them the right choice for implementing machine learning-based network intrusion detection systems.

Keywords: *random forests; k-nearest neighbors; support vector machines; anomaly detection; intrusion detection system*

1. PENDAHULUAN

Ancaman di dunia maya seperti serangan Denial of Service (DoS) dan Distributed DoS (DDoS) mengalami peningkatan yang signifikan dan menjadi salah satu masalah utama dalam keamanan jaringan. Serangan DDoS merupakan risiko besar bagi keselamatan jaringan, sedangkan cara deteksi tradisional seperti threshold-based detection dan signature-based detection memiliki kekurangan dalam mengidentifikasi pola serangan yang baru serta anomali lalu lintas yang rumit [1]. Serangan DoS beroperasi dengan membanjiri sumber daya dari sistem yang menjadi target, sehingga layanan tidak dapat diakses oleh pengguna yang berhak, dan akibatnya dapat menimbulkan kerugian yang besar baik dari segi finansial maupun reputasi bagi organisasi [2]. Situasi ini mendorong munculnya kebutuhan mendesak untuk sistem deteksi intrusi (Intrusion Detection System/IDS) yang dapat dengan tepat dan efisien mendeteksi anomali dalam trafik jaringan.

Pendekatan IDS yang didasarkan pada aturan konvensional memiliki batasan signifikan karena tidak dapat mengidentifikasi pola serangan baru yang belum tercatat dalam aturan yang ada, menjadikannya rentan terhadap serangan *zero-day*[3]. Di samping itu, pemeliharaan aturan yang dilakukan secara manual memerlukan banyak sumber daya dan tidak mampu beradaptasi dengan meningkatnya volume trafik saat ini. Metode yang menggunakan machine learning (ML) muncul sebagai solusi yang dapat mempelajari dari data sebelumnya dan beradaptasi dengan pola serangan baru secara otomatis, memberikan fleksibilitas yang jauh lebih baik dibandingkan metode tradisional[4].

Dalam konteks ini, pemilihan algoritma machine learning yang sesuai menjadi elemen penting yang mempengaruhi keberhasilan sistem Intrusion Detection System (IDS). Penelitian ini memilih tiga algoritma dengan karakteristik yang berbeda, yaitu Random Forest (RF), K-Nearest Neighbor (KNN), dan Support Vector Machine (SVM). Random Forest merupakan algoritma berbasis ensemble yang membangun sejumlah pohon keputusan dan menggabungkan hasil prediksinya melalui mekanisme voting

mayoritas. Pendekatan ini memungkinkan Random Forest menghasilkan performa klasifikasi yang tinggi, memiliki ketahanan terhadap overfitting, serta mampu menangani data berdimensi tinggi secara efektif. Selain itu, algoritma ini juga dapat digunakan untuk mengidentifikasi tingkat kepentingan fitur yang berkontribusi dalam proses klasifikasi[5].

K-Nearest Neighbor (KNN) merupakan algoritma pembelajaran nonparametrik yang melakukan klasifikasi berdasarkan kedekatan suatu data terhadap sejumlah tetangga terdekat pada data latih. Algoritma ini tidak membangun model secara eksplisit selama proses pelatihan, melainkan menentukan kelas data baru melalui perhitungan jarak terhadap data yang telah tersimpan. Karakteristik tersebut menjadikan KNN sederhana dan efektif dalam berbagai kasus klasifikasi. Namun, kinerja komputasinya cenderung menurun ketika diterapkan pada dataset berukuran besar karena seluruh data latih harus dipertimbangkan pada saat proses prediksi[6]. Sementara itu, Support Vector Machine bekerja dengan prinsip yang sama sekali berbeda, yaitu menemukan hyperplane pemisah dengan margin maksimum dalam ruang fitur yang memiliki dimensi tinggi. Dengan menggunakan fungsi kernel seperti Radial Basis Function (RBF), SVM mampu menangani data yang tidak dapat dipisahkan secara linear, tetapi kompleksitas pelatihannya, yang berkisar antara $O(n^2)$ hingga $O(n^3)$, membuatnya sangat sensitif terhadap ukuran data dan volume dataset[7].

Perbedaan dasar dalam paradigma ini merupakan alasan utama mengapa perbandingan ketiga algoritma tersebut memiliki makna ilmiah. RF menggambarkan pendekatan ensemble yang menggunakan pohon, KNN menunjukkan pendekatan yang berbasis pada kedekatan antar instans, sedangkan SVM merepresentasikan pendekatan yang berfokus pada margin geometris. Menganalisis ketiganya bukan hanya sekadar teknik, melainkan sebuah usaha untuk memahami cara masing-masing paradigma beradaptasi dengan sifat-sifat unik dari data trafik jaringan, yang biasanya memiliki dimensi tinggi, tidak seimbang antar kelas, dan mengandung beragam pola serangan dalam satu dataset yang sama.

Beberapa studi sebelumnya telah menyelidiki algoritma Machine Learning untuk sistem deteksi intrusi dengan hasil yang positif. Rosay et al.[8] melakukan kajian mendalam terhadap dataset CICIDS2017 dan mengungkapkan bahwa algoritma pembelajaran mesin seperti Random Forest dan KNN dapat mencapai tingkat klasifikasi yang sangat baik pada lalu lintas jaringan yang sesungguhnya, dengan catatan bahwa mutu fitur yang dihasilkan oleh CICFlowMeter sangat berpengaruh terhadap hasil deteksi. Rozam et al.[9] membandingkan beberapa algoritma machine learning, yaitu Random Forest, K-Nearest Neighbor (KNN), Support Vector Machine (SVM), Logistic Regression, dan Gaussian Naïve Bayes pada dataset CICIDS2017. Penelitian oleh Abushwereb et al. juga menunjukkan bahwa algoritma Random Forest memiliki kinerja yang sangat baik dalam mendeteksi intrusi pada lingkungan Internet of Things (IoT), dengan nilai F1-score mencapai 99,6%[10]. Rodríguez et al, [11] membuktikan bahwa metode berbasis decision tree, termasuk Random Forest, mendapatkan nilai F1 lebih dari 0,999 pada CICIDS2017. Kiswanto et al,[12] menerapkan model untuk mendeteksi serangan DDoS yang menggunakan Random Forest dan menilainya dengan mengacu pada metrik akurasi, precision, recall, F1-score, serta confusion matrix, dengan hasil yang memperlihatkan kemampuan yang baik dalam mengklasifikasikan trafik normal dan serangan dengan tepat.

Namun, ada beberapa aspek penting yang belum terjawab dalam literatur yang tersedia. Pertama, banyak penelitian sebelumnya menganalisis algoritma dengan menggunakan seluruh dataset CICIDS2017 tanpa membedakan skenario serangan tertentu, sehingga hasil yang dilaporkan merupakan rata-rata keseluruhan yang dapat menyembunyikan perilaku algoritma dalam konteks jenis serangan tertentu. Kedua, belum ada penelitian yang secara spesifik membandingkan RF, KNN, dan SVM secara bersamaan pada subset data DoS hari Rabu dari CICIDS2017, yang memiliki karakteristik unik yaitu ketidakseimbangan kelas yang ekstrem, mulai dari kelas dominan seperti Benign (66,9%) hingga kelas yang sangat jarang seperti Heartbleed (kurang dari 0,01%). Ketiga, penelitian-penelitian sebelumnya jarang membahas secara bersamaan efektivitas klasifikasi yang tepat per kelas, stabilitas antar fold validasi silang, serta kemampuan generalisasi pada data uji yang sama sekali belum pernah dilihat oleh model, padahal ketiga aspek ini sangat penting dalam mengevaluasi kesesuaian suatu algoritma untuk penerapan sistem IDS dalam praktik.

Dengan mempertimbangkan kekurangan ini, penelitian ini dimaksudkan untuk menjawab pertanyaan khusus berikut: di antara RF, KNN, dan SVM, algoritma mana yang paling efisien dalam mengklasifikasikan trafik jaringan dalam konteks DoS berdasarkan kelas, memiliki stabilitas terbaik dalam konsistensi performa lintas *fold*, dan paling mampu menggeneralisasi pada data uji yang belum pernah terlihat sebelumnya (*holdout test set*)? Dengan menetapkan ketiga dimensi evaluasi tersebut secara tegas, penelitian ini tidak hanya berupaya untuk menentukan algoritma terbaik berdasarkan akurasi saja, tetapi juga untuk memberikan panduan yang lebih menyeluruh bagi profesional keamanan jaringan dalam memilih algoritma yang paling sesuai untuk kebutuhan penerapan sistem deteksi intrusi berbasis machine learning yang baik dan stabil.

2. METODE PENELITIAN

Penelitian ini dirancang melalui rangkaian langkah sistematis untuk memastikan ketepatan dan konsistensi hasil eksperimen, dimulai dengan pengumpulan data, pemetaan serta analisis penyebaran data, pemrosesan awal, pembagian data untuk pelatihan dan pengujian, pelatihan model, sampai pada penilaian kinerja ketiga algoritma. Setiap langkah dibuat sedemikian rupa sehingga perbandingan antarg algoritma dilakukan dalam kondisi data yang sama, sehingga perbedaan hasil yang diperoleh hanya mencerminkan karakteristik masing-masing algoritma dan bukan karena perbedaan penanganan data.

Dataset

Dataset ini mencerminkan situasi serangan DoS yang telah tercatat pada hari Rabu dan terdiri dari 692.703 aliran jaringan yang belum diolah, dengan 79 atribut numerik yang diambil melalui CICFlowMeter[13]. Dataset ini menggambarkan keadaan serangan DoS yang terdokumentasi pada hari Rabu dan terdiri dari 692. 703 aliran jaringan yang belum diproses. Setelah dilakukan pembersihan data (penghapusan nilai yang hilang dan nilai infinit), dihasilkan 584. 991 aliran yang digunakan untuk analisis berikutnya, yang dibagi menjadi enam kategori label: Benign, DoS Hulk, DoS GoldenEye, DoS slowloris, DoS Slowhttptest, serta Heartbleed. Tiap sampel memiliki 79 fitur numerik yang diperoleh melalui CICFlowMeter, meliputi statistik paket, durasi koneksi, ukuran payload, dan flag TCP [14].

Pemetaan Data

Analisis tentang distribusi kelas dilakukan untuk mengetahui susunan dan sifat dari trafik sebelum tahapan *preprocessing*. Temuan distribusi mengungkapkan adanya ketidakseimbangan kelas yang cukup besar sebagai berikut:

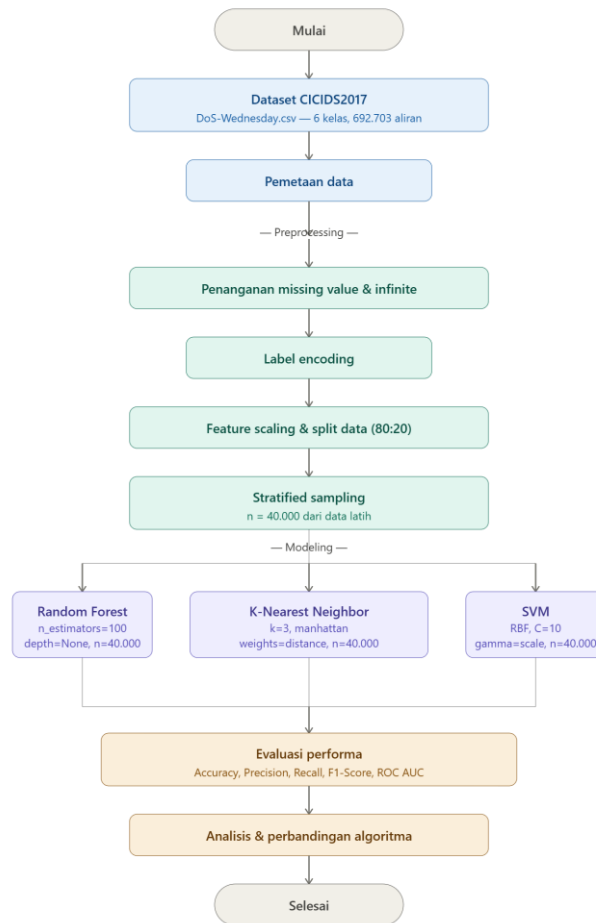
Tabel 1. Distribusi Kelas pada Dataset Trafik Jaringan (CIC-IDS2017)

Kelas	Jumlah	Persentase	Karakteristik
Benign	391.235	66,9%	Trafik normal
DoS Hulk	172.846	29,5%	Request HTTP acak masif, durasi pendek
DoS GoldenEye	10.286	1,8%	HTTP keep-alive, menghabiskan koneksi
DoS slowloris	5.385	0,9%	Koneksi lambat, lolos firewall
DoS Slowhttptest	5.228	0,9%	HTTP request sangat lambat
Heartbleed	11	<0,01%	Eksplorasi OpenSSL, bocorkan memori

Distribusi yang tertera pada Tabel 1 memperlihatkan ketidakseimbangan kelas yang sangat berat, di mana kelas Benign dan DoS Hulk mengambil hampir 96% dari keseluruhan data, sedangkan kelas Heartbleed hanya mempunyai 11 sampel dari total 584. 991 aliran. Situasi ini menjadi faktor penting yang harus diperhatikan saat membagi data dan mengevaluasi model di tahap berikutnya, terutama dalam memastikan bahwa setiap kelas terwakili secara proporsional dalam data latih dan data uji.

Alur Penelitian

Seluruh langkah-langkah penelitian, dari pengumpulan data hingga analisis perbandingan algoritma, diilustrasikan dalam bentuk diagram alir yang terlihat pada Gambar 1. Diagram ini menunjukkan urutan proses dengan menggunakan notasi standar, yaitu simbol *terminator* untuk menandai awal dan akhir penelitian, simbol kotak untuk setiap langkah proses pengolahan data, dan titik cabang paralel pada fase pemodelan, di mana ketiga algoritma (Random Forest, KNN, dan SVM) dilatih secara terpisah dengan menggunakan subsampel data yang sama sebelum hasilnya digabungkan kembali pada fase evaluasi.



Gambar 1. Flowchart Alur Penelitian

Sebagaimana diperlihatkan dalam Gambar 1, tahapan penelitian diawali dengan pengumpulan dataset CICIDS2017, diikuti dengan pemetaan data untuk mengetahui distribusi kelas. Setelah itu, proses *preprocessing* dilaksanakan, yang meliputi penanganan nilai yang hilang dan nilai tak terhingga, *encoding label*, penyesuaian skala fitur, serta pembagian data dengan perbandingan 80:20. Kemudian, dilakukan pengambilan sampel terstratifikasi sejumlah $n=40.000$ dari data pelatihan, yang berfungsi sebagai titik awal untuk ketiga algoritma dilatih secara bersamaan dengan pengaturan hyperparameter masing-masing. Hasil dari pelatihan ketiga model tersebut kemudian digabungkan pada tahap evaluasi kinerja menggunakan metrik akurasi, presisi, recall, F1-score, dan ROC AUC, sebelum akhirnya dianalisis dan dibandingkan untuk menemukan algoritma yang memiliki performa terbaik.

Preprocessing

Tahapan *preprocessing* dilakukan pada empat langkah penting. Pertama, penanganan nilai hilang dan nilai tak terhingga: semua baris yang memiliki nilai infinite diubah menjadi NaN, lalu baris-baris tersebut dihapus menggunakan `dropna()`, sehingga data bersih dari nilai yang tidak valid. Kedua, dilakukan Label Encoding dengan

memanfaatkan LabelEncoder dari scikit-learn untuk mengubah label kelas yang berbentuk string menjadi representasi numerik dalam bentuk integer. Ketiga, proses feature scaling diterapkan menggunakan StandardScaler untuk menstandarisasi semua fitur numerik agar memiliki distribusi dengan rata-rata nol dan deviasi standar satu, penting untuk kinerja KNN dan SVM karena keduanya merupakan algoritma berbasis jarak yang sensitif terhadap perbedaan rentang nilai antar fitur [15][16]. Keempat, pembagian data dilakukan dengan metode stratified menggunakan rasio 80% untuk data latih dan 20% untuk data uji (random_state=42), agar menjaga keseimbangan proporsi kelas di kedua subset.

Modeling

Ketiga algoritma pembelajaran mesin diterapkan dengan menggunakan sampel seragam sebesar $n=40.000$ data pelatihan yang diambil secara stratifikasi, untuk menjamin perbandingan yang adil di antara algoritma serta mengatasi batasan komputasi SVM [17][18]. Pengaturan masing-masing algoritma adalah sebagai berikut:

Random Forest

Dilatih pada subsampel $n=40.000$ yang diambil secara stratifikasi dari data pelatihan (random_state=42), serupa dengan KNN dan SVM untuk memastikan perbandingan yang adil. Proses optimasi hyperparameter dilakukan menggunakan GridSearchCV dengan ruang pencarian $n_estimators \in \{100, 200, 300\}$, $max_depth \in \{None, 20, 30\}$, dan $min_samples_split \in \{2, 5\}$. Parameter terbaik yang diperoleh adalah $n_estimators=100$, $max_depth=None$, dan $min_samples_split=5$. RF membangun koleksi pohon keputusan secara bersamaan dan mengkombinasikan prediksi dengan suara mayoritas, menjadikannya tahan terhadap gangguan dan overfitting [19][20].

K-Nearest Neighbor/KNN

Dilatih pada subsampel sebanyak $n=40.000$ yang serupa dengan RF dan SVM. Proses penyetelan hyperparameter dilakukan dengan GridSearchCV menggunakan ruang pencarian $n_neighbors \in \{3, 5, 7, 9, 11\}$, $metric \in \{euclidean, manhattan\}$, dan $weights \in \{uniform, distance\}$. Parameter optimal yang didapatkan adalah $n_neighbors=3$, $metric=manhattan$, dan $weights=distance$. KNN melakukan klasifikasi pada data baru berdasarkan kelas yang paling sering muncul dari k tetangga terdekat dalam ruang fitur tanpa memerlukan fase pelatihan yang eksplisit [21][22].

Support Vector Machine/SVM

Dilatih pada kumpulan sampel $n=40.000$ yang serupa dengan RF dan KNN. Pengaturan hyperparameter dilakukan dengan GridSearchCV menggunakan kisaran nilai $C \in \{1, 10\}$, $gamma = scale$, dan $kernel = rbf$. Nilai parameter terbaik yang ditemukan adalah $C=10$, $gamma=scale$, dan $kernel=rbf$. SVM berusaha menemukan hyperplane pemisah dengan margin tertinggi menggunakan kernel Radial Basis Function (RBF) untuk mengatasi data yang tidak linear. Mengingat kompleksitas pelatihan yang berkisar antara $O(n^2)$ hingga $O(n^3)$, subsampling menjadi suatu keharusan dalam metodologi [23][24].

Evaluasi

Evaluasi kinerja tiga algoritma dilakukan dalam dua langkah. Langkah pertama melibatkan 5-Fold Stratified Cross-Validation pada subset data pelatihan yang serupa ($n=40.000$), untuk menilai stabilitas dan konsistensi kinerja di setiap fold dengan menyajikan nilai rata-rata dan deviasi standar untuk setiap metrik. Langkah kedua adalah evaluasi akhir pada set uji holdout (20% dari total data) yang tidak pernah dimasukkan dalam proses pelatihan, sehingga dapat menggambarkan kemampuan model dalam menggeneralisasi terhadap data yang belum pernah ditemui sebelumnya [25][26].

Evaluasi kinerja ketiga algoritma dilakukan dengan menggunakan data pengujian yang belum pernah dihadapi oleh model saat fase pelatihan. Metrik yang digunakan meliputi: (1) Akurasi, yaitu rasio prediksi yang benar dibandingkan dengan jumlah total sampel pengujian; (2) Ketepatan (berbobot), yang mengukur akurasi prediksi positif dengan memperhatikan bobot masing-masing kelas; (3) Recall (berbobot), yang mengukur kemampuan model dalam mengenali kelas yang sebenarnya; (4) Skor F1 (berbobot), yang merupakan rata-rata harmonik dari ketepatan dan recall yang lebih relevan pada kelas-kelas yang tidak seimbang [27][28]; dan (5) ROC AUC, yang menilai seberapa baik model dapat membedakan antara kelas dengan menggunakan pendekatan One-vs-Rest

3. HASIL DAN PEMBAHASAN

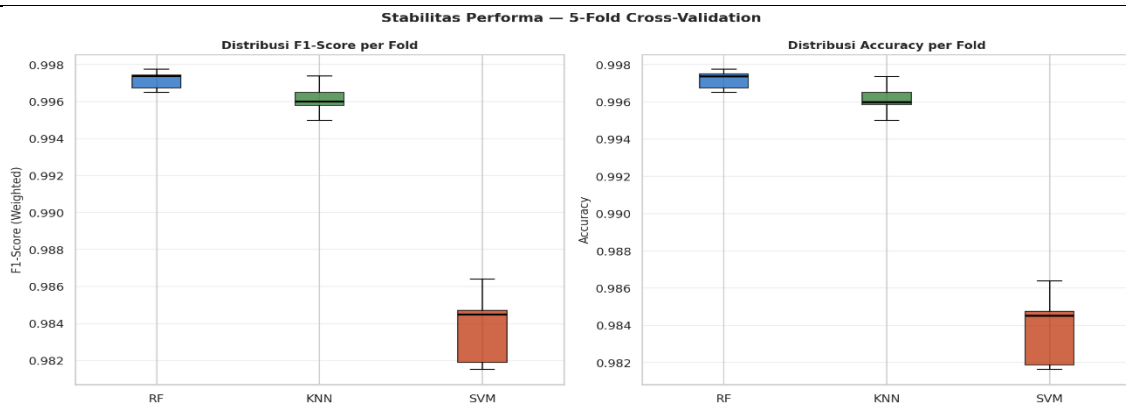
Hasil Cross-Validation

Tabel 2, menyajikan hasil 5-Fold Stratified Cross-Validation pada subsampel data latih yang identik ($n=40.000$) untuk ketiga algoritma. Penggunaan data latih yang sama pada ketiga model memastikan bahwa perbedaan performa semata-mata disebabkan oleh karakteristik algoritma, bukan oleh perbedaan data yang digunakan.

Tabel 2. Hasil 5-Fold Stratified Cross-Validation (mean $\pm$ std)

Algoritma	Accuracy	Precision	Recall	F1-Score
Random Forest	0,9972 $\pm$ 0,0005	0,9972 $\pm$ 0,0005	0,9972 $\pm$ 0,0005	0,9972 $\pm$ 0,0005
K-Nearest Neighbor	0,9961 $\pm$ 0,0008	0,9961 $\pm$ 0,0008	0,9961 $\pm$ 0,0008	0,9961 $\pm$ 0,0008
Support Vector Machine	0,9838 $\pm$ 0,0018	0,9841 $\pm$ 0,0018	0,9838 $\pm$ 0,0018	0,9838 $\pm$ 0,0018

Berdasarkan Tabel 2, Random Forest mencapai F1-Score CV tertinggi sebesar $0,9972 \pm 0,0005$, diikuti KNN sebesar $0,9961 \pm 0,0008$, dan SVM sebesar $0,9838 \pm 0,0018$. Nilai standar deviasi RF yang paling kecil mengindikasikan performa paling stabil antar fold. Sebaliknya, SVM menunjukkan variabilitas tertinggi yang dapat dikaitkan dengan sensitivitasnya terhadap distribusi data pada tiap fold. Gambar 2 memperlihatkan distribusi skor per fold secara visual melalui boxplot.



Gambar 2. Distribusi F1_Score dan Accuracy per Fold (5-Fold Cross-Validation)

Gambar 2, mengkonfirmasi superioritas RF dengan box paling sempit dan posisi tertinggi, menandakan kestabilan dan konsistensi yang unggul. KNN memiliki distribusi lebih lebar namun tetap di kisaran tinggi. SVM memiliki rentang terlebar (0,981–0,987), mengindikasikan variabilitas yang lebih besar.

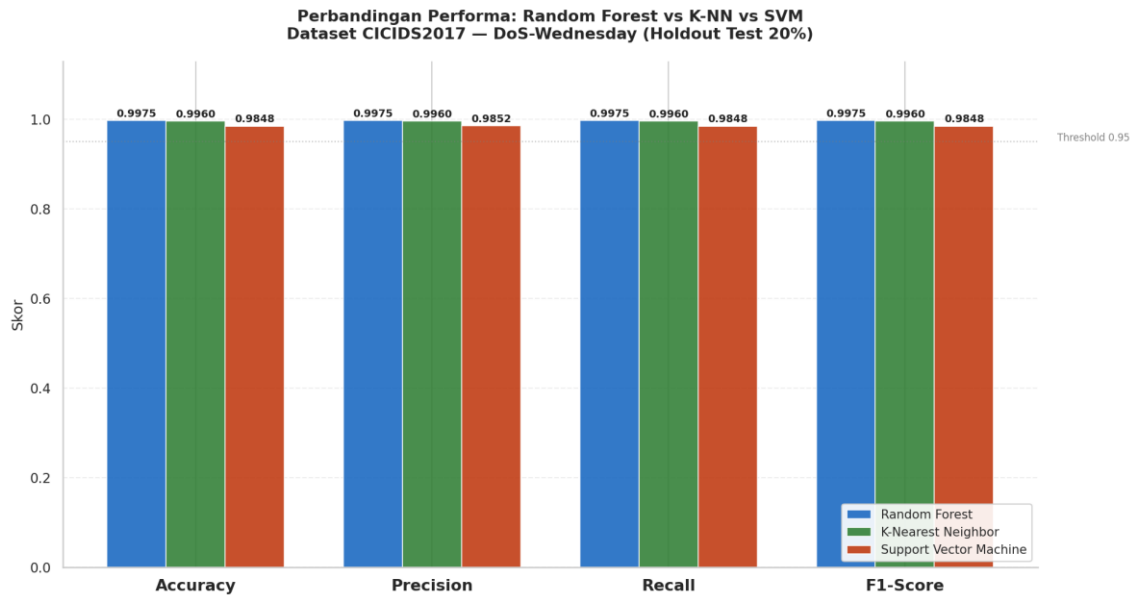
Perbandingan Performa pada Data Pengujian

Tabel 3, menyajikan hasil evaluasi final pada holdout test set (20%) yang tidak pernah diikuti sertakan dalam proses training maupun tuning, sehingga mencerminkan kemampuan generalisasi model terhadap data trafik jaringan yang belum pernah dilihat sebelumnya.

Tabel 3. Perbandingan Metrik Evaluasi pada Data Pengujian (Holdout 20%)

Algoritma	Accuracy	Precision	Recall	F1-Score	ROC AUC
Random Forest	0,9975	0,9975	0,9975	0,9975	0,9999
K-Nearest Neighbor	0,9960	0,9960	0,9960	0,9960	0,9987
Support Vector Machine	0,9848	0,9852	0,9848	0,9848	0,9998

Berdasarkan Tabel 3, Random Forest meraih performa terbaik pada seluruh metrik dengan Accuracy, Precision, Recall, dan F1-Score masing-masing sebesar 0,9975, serta ROC AUC 0,9999. K-Nearest Neighbor menempati posisi kedua dengan F1-Score 0,9960 dan ROC AUC 0,9987. Support Vector Machine memperoleh F1-Score 0,9848 dan ROC AUC 0,9998. Meskipun ROC AUC SVM (0,9998) lebih tinggi dari KNN (0,9987), nilai F1-Score KNN lebih unggul, menunjukkan KNN lebih baik dalam klasifikasi aktual per kelas.

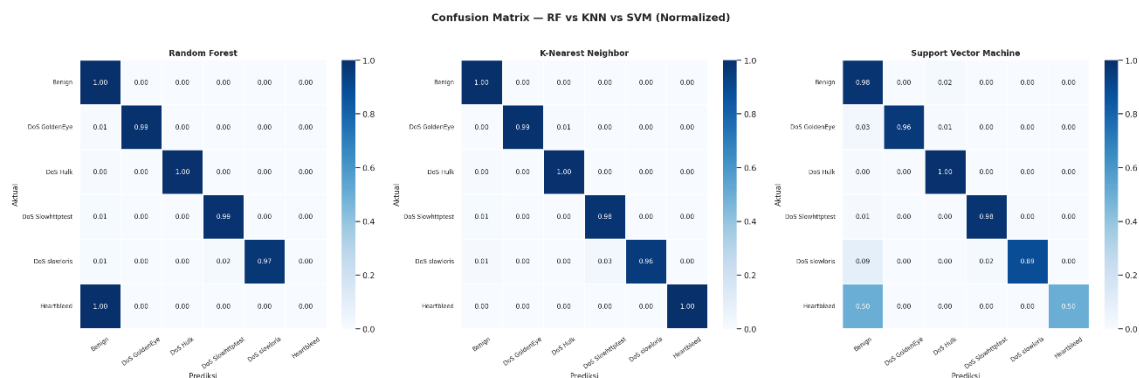


Gambar 3. Perbandingan Metrik Evaluasi RF, KNN, dan SVM pada Data Pengujian

Gambar 3, memperlihatkan bahwa ketiga algoritma melampaui threshold 0,95 pada seluruh metrik, mengindikasikan bahwa pola trafik jaringan pada dataset CICIDS2017 dapat dipelajari dengan sangat baik oleh ketiga pendekatan. Selisih F1-Score antara RF dan KNN relatif kecil ($\approx 0,0015$), namun konsisten pada seluruh metrik, memperkuat keunggulan RF sebagai kandidat utama implementasi sistem deteksi intrusi.

Analisis Confusion Matrix

Gambar 4, menyajikan confusion matrix yang dinormalisasi (proporsi per baris) untuk ketiga algoritma secara berdampingan, memberikan gambaran mendalam mengenai kemampuan setiap model dalam mengklasifikasikan setiap jenis trafik jaringan.



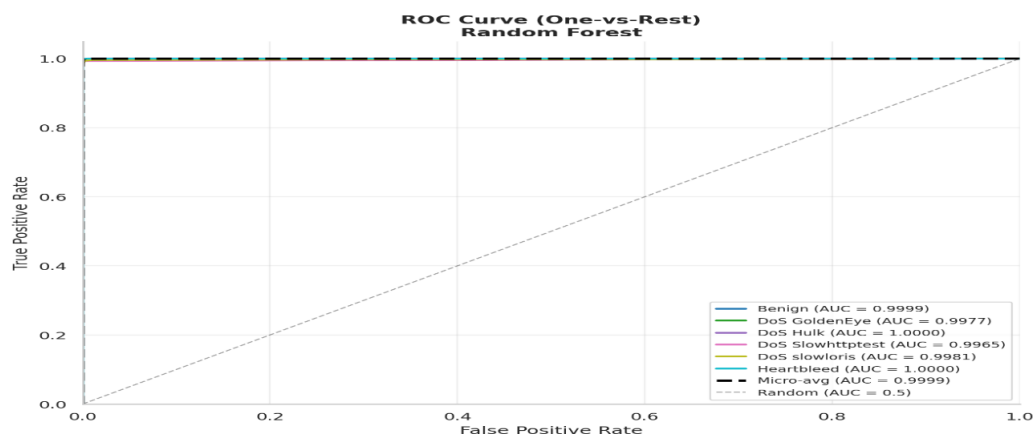
Gambar 4. Confusion Matrix Normalized RF, KNN, SVM

Berdasarkan Gambar 4, RF dan KNN mengklasifikasikan hampir seluruh kelas dengan benar, dibuktikan nilai diagonal mendekati 1,00. RF mengklasifikasikan Benign (1,00), DoS Hulk (1,00), dan DoS GoldenEye (0,99) dengan sempurna. SVM menunjukkan penurunan performa lebih signifikan pada Benign (0,98) dan DoS slowloris (0,89), mengindikasikan lebih banyak kesalahan klasifikasi. Kasus menarik terjadi pada kelas Heartbleed: RF gagal mendeteksi seluruh sampel Heartbleed di data uji (diagonal 0,00), KNN berhasil mendeteksi seluruhnya (diagonal 1,00), dan SVM mendeteksi 50%.

Fenomena ini menunjukkan bahwa algoritma berbasis kedekatan jarak lokal (KNN) memiliki keunggulan pada kelas dengan sampel sangat terbatas, sementara RF yang dioptimasi secara global cenderung mengabaikan kelas minor dengan representasi ekstrem.

Analisis ROC Curve dan AUC

ROC Curve dianalisis menggunakan strategi One-vs-Rest (OvR) untuk setiap kelas. Gambar 5 menampilkan ROC Curve Random Forest sebagai model dengan performa terbaik.

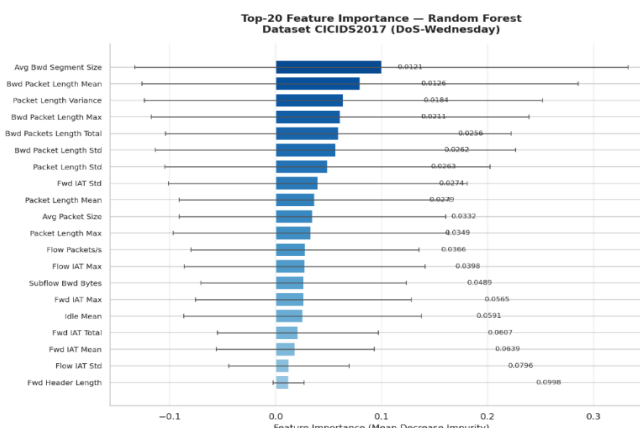


Gambar 5. ROC Curve Random Forest (One-vs-Rest)

Gambar 5 menunjukkan seluruh kurva ROC RF berada sangat dekat sudut kiri atas, dengan micro-average AUC sebesar 0,9999. Kelas DoS Hulk dan Heartbleed mencapai AUC sempurna (1,0000), sementara DoS Slowhttptest memperoleh nilai terendah (0,9965) namun tetap sangat tinggi. KNN dan SVM juga mencapai micro-average AUC masing-masing 0,9987 dan 0,9998, mengkonfirmasi kemampuan diskriminasi yang sangat baik dari ketiga algoritma dalam memisahkan kelas trafik jaringan.

Analisis Feature Importance

Gambar 6, menyajikan 20 fitur dengan kontribusi tertinggi terhadap keputusan klasifikasi Random Forest berdasarkan metode Mean Decrease Impurity (MDI).



Gambar 6. Top-20 Feature Importance — Random Forest

Berdasarkan Gambar 6, fitur paling berpengaruh adalah Fwd Header Length (0,0998), Flow IAT Std (0,0796), dan Fwd IAT Mean (0,0639). Fitur-fitur ini mencerminkan karakteristik temporal aliran jaringan, khususnya variasi waktu antar paket (Inter-Arrival Time/IAT). Serangan DoS menghasilkan pola IAT yang sangat berbeda dari trafik normal karena membanjiri server dengan paket dalam interval sangat pendek dan seragam. Fitur berbasis ukuran paket seperti Bwd Packet Length Mean, Packet Length Variance, dan Avg Bwd Segment Size juga berkontribusi signifikan, konsisten dengan karakteristik serangan DoS yang sering menggunakan paket berukuran tidak normal.

Pembahasan

Secara keseluruhan, Random Forest mengungguli KNN dan SVM pada hampir seluruh metrik evaluasi, konsisten dengan temuan dalam literatur deteksi intrusi jaringan. Keunggulan RF dikaitkan dengan sifat ensemble-nya yang mampu menangkap pola kompleks data multidimensi, ketahanan terhadap noise dan outlier, serta kemampuan seleksi fitur otomatis melalui mekanisme random feature selection. Hasil ini menunjukkan bahwa RF merupakan pilihan yang tepat untuk implementasi sistem IDS berbasis machine learning pada dataset CICIDS2017.

Terdapat beberapa keterbatasan yang perlu dicatat. Pertama, ketiga model dilatih pada subsampel $n=40.000$ karena keterbatasan komputasi SVM; RF berpotensi mencapai performa lebih tinggi pada data penuh. Kedua, kelas Heartbleed dengan hanya 11 data pada keseluruhan dataset menjadi kasus khusus di mana KNN lebih unggul dari RF, menunjukkan bahwa pendekatan berbasis jarak lokal memiliki kelebihan pada kelas dengan distribusi sangat terbatas. Ketiga, evaluasi terbatas pada satu file dataset (DoS-Wednesday), sehingga generalisasi ke hari lain atau jenis serangan berbeda memerlukan validasi lebih lanjut. Untuk penelitian mendatang, disarankan penerapan teknik resampling seperti SMOTE untuk menangani class imbalance ekstrem, serta eksplorasi pendekatan deep learning untuk menangkap pola sekuensial pada trafik jaringan.

4. KESIMPULAN

Penelitian ini menganalisis perbandingan antara tiga algoritma pembelajaran mesin dalam mendeteksi anomali trafik jaringan pada dataset CICIDS2017 (DoS-Wednesday). Algoritma Random Forest (RF) berhasil menunjukkan kinerja terbaik dengan F1-Score mencapai 0,9975 dan ROC AUC 0,9999 saat diuji, didukung hasil validasi silang yang menunjukkan deviasi standar terkecil, menandakan stabilitas tertinggi. K-Nearest Neighbor (KNN) berada di posisi kedua dengan F1-Score 0,9960, tetapi berhasil mengungguli RF pada kategori Heartbleed yang memiliki sampel terbatas, menegaskan keunggulan algoritma berbasis jarak lokal dalam situasi data yang sangat tidak seimbang. Support Vector Machine (SVM) memiliki F1-Score terendah di antara ketiganya, yakni 0,9848, dengan performa klasifikasi kelas yang tidak optimal, terutama pada serangan DoS slowloris dan Heartbleed. Analisis pentingnya fitur pada RF menunjukkan bahwa fitur temporal seperti Fwd Header Length dan Flow IAT Std adalah yang paling dapat membedakan serangan DoS dari trafik biasa. Secara keseluruhan, ketiga algoritma telah melampaui nilai akurasi 0,95, sehingga RF direkomendasikan

sebagai mesin utama untuk deteksi intrusi karena kinerjanya yang terbaik, stabilitasnya, dan kemampuan untuk menginterpretasi model. Untuk penelitian mendatang, disarankan untuk menggunakan SMOTE guna mengatasi ketidakseimbangan kelas, mengevaluasi keseluruhan file dataset CICIDS2017, serta menggali penggunaan pendekatan pembelajaran mendalam seperti LSTM dan CNN-1D.

UCAPAN TERIMA KASIH

Penelitian ini menganalisis perbandingan antara tiga algoritma pembelajaran mesin dalam mendeteksi anomali trafik jaringan pada dataset CICIDS2017 (DoS-Wednesday). Algoritma Random Forest (RF) berhasil menunjukkan kinerja terbaik dengan F1-Score mencapai 0,9975 dan ROC AUC 0,9999 saat diuji, didukung hasil validasi silang yang menunjukkan deviasi standar terkecil, menandakan stabilitas tertinggi. K-Nearest Neighbor (KNN) berada di posisi kedua dengan F1-Score 0,9960, tetapi berhasil mengungguli RF pada kategori Heartbleed yang memiliki sampel terbatas, menegaskan keunggulan algoritma berbasis jarak lokal dalam situasi data yang sangat tidak seimbang. Support Vector Machine (SVM) memiliki F1-Score terendah di antara ketiganya, yakni 0,9848, dengan performa klasifikasi kelas yang tidak optimal, terutama pada serangan DoS slowloris dan Heartbleed. Analisis pentingnya fitur pada RF menunjukkan bahwa fitur temporal seperti Fwd Header Length dan Flow IAT Std adalah yang paling dapat membedakan serangan DoS dari trafik biasa. Secara keseluruhan, ketiga algoritma telah melampaui nilai akurasi 0,95, sehingga RF direkomendasikan sebagai mesin utama untuk deteksi intrusi karena kinerjanya yang terbaik, stabilitasnya, dan kemampuan untuk menginterpretasi model. Untuk penelitian mendatang, disarankan untuk menggunakan SMOTE guna mengatasi ketidakseimbangan kelas, mengevaluasi keseluruhan file dataset CICIDS2017, serta menggali penggunaan pendekatan pembelajaran mendalam seperti LSTM dan CNN-1D.

DAFTAR PUSTAKA

- [1] Maulana and Alamsyah, “48231-124397-1-SM 1,” *Indonesian Journal of Mathematics and Natural Sciences*, vol. 46, no. 2, pp. 83–92, Oct. 2023, doi: <http://journal.unnes.ac.id/nju/index.php/JM>.
- [2] M. Ramzan *et al.*, “Distributed Denial of Service Attack Detection in Network Traffic Using Deep Learning Algorithm,” *Sensors (Basel)*, vol. 23, no. 20, Oct. 2023, doi: 10.3390/s23208642.
- [3] U. Ahmed *et al.*, “Signature-based intrusion detection using machine learning and deep learning approaches empowered with fuzzy clustering,” *Sci. Rep.*, vol. 15, no. 1, Dec. 2025, doi: 10.1038/s41598-025-85866-7.
- [4] M. Ismail Mohmand *et al.*, “A Machine Learning based Classification and Prediction Technique for DDoS Attacks,” vol. 4, pp. 1–13, 2022, doi: 10.1109/ACCESS.2017.DOI.
- [5] Y. Yang, Y. Zhang, L. Yang, and Y. Wang, “Wideband Direction-of-Arrival Estimation Based on Hierarchical Sparse Bayesian Learning for Signals with the Same or Different Frequency Bands,” *Electronics (Switzerland)*, vol. 12, no. 5, Mar. 2023, doi: 10.3390/electronics12051123.

- [6] A. Momand, S. U. Jan, and N. Ramzan, "A Systematic and Comprehensive Survey of Recent Advances in Intrusion Detection Systems Using Machine Learning: Deep Learning, Datasets, and Attack Taxonomy," 2023, *Hindawi Limited*. doi: 10.1155/2023/6048087.
- [7] M. K. Ngueajio, G. Washington, D. B. Rawat, and Y. Ngueabou, "Intrusion Detection Systems Using Support Vector Machines on the KDDCUP'99 and NSL-KDD Datasets. A Comprehensive Survey," 2022. doi: arXivpreprint,arXiv:2209.05579.
- [8] A. Rosay, E. Cheval, F. Carlier, and P. Leroux, "Network Intrusion Detection: A Comprehensive Analysis of CIC-IDS2017," in *International Conference on Information Systems Security and Privacy*, Science and Technology Publications, Lda, 2022, pp. 25–36. doi: 10.5220/0010774000003120.
- [9] N. F. Rozam, T. N. Sari, M. R. A. Yudianto, and D. F. Rahman, "Machine Learning-Based Network Traffic Anomaly Detection Using the CIC-IDS2017 Dataset," *UPGRADE : Jurnal Pendidikan Teknologi Informasi*, vol. 3, no. 2, Apr. 2026, doi: 10.30812/upgrade.v3i2.6174.
- [10] M. Abushwereb, M. Al-Kasassbeh, M. Almseidin, and M. Mustafa, "An accurate IoT Intrusion Detection Framework using Apache Spark," 2022. doi: arXivpreprintarXiv:2203.04347.
- [11] M. Rodríguez, Á. Alesanco, L. Mehavilla, and J. García, "Evaluation of Machine Learning Techniques for Traffic Flow-Based Intrusion Detection," *Sensors*, vol. 22, no. 23, Dec. 2022, doi: 10.3390/s22239326.
- [12] D. Kiswanto, F. Ramadhani, N. M. Surbakti, N. A. Nasution, F. Matematika, and I. P. Alam, "Pengembangan dan Implementasi Sistem Deteksi Serangan DDoS Berbasis Algoritma Random Forest," *Bulletin of Information Technology (BIT)*, vol. 6, no. 3, pp. 247–256, 2025, doi: 10.47065/bit.v5i2.2203.
- [13] S. S. Tripathy and B. Behera, "A Review of Various Datasets for Machine Learning Algorithm-Based Intrusion Detection System: Advances and Challenges," May 2024. doi: www.ijisae.org.
- [14] A. W. Ningrum, M. P. Aji, E. S. Wijaya, and E. A. Pambudi, "Deteksi dan Klasifikasi Ancaman pada Log Serangan Siber Menggunakan Algoritma K-Nearest Neighbor (KNN) dan Random Forest (RF)," *Jurnal Pendidikan dan Teknologi Indonesia*, vol. 5, no. 12, pp. 3610–3619, Jan. 2026, doi: 10.52436/1.jpti.1197.
- [15] A. Singh, J. Amutha, J. Nagar, S. Sharma, and C. C. Lee, "LT-FS-ID: Log-Transformed Feature Learning and Feature-Scaling-Based Machine Learning Algorithms to Predict the k-Barriers for Intrusion Detection Using Wireless Sensor Network," *Sensors*, vol. 22, no. 3, Feb. 2022, doi: 10.3390/s22031070.
- [16] J. M. H. Pinheiro *et al.*, "The Impact of Feature Scaling in Machine Learning: Effects on Regression and Classification Tasks," *IEEE Access*, vol. 11, pp. 199903–199931, 2025, doi: 10.1109/ACCESS.2025.3635541.
- [17] S. Sekhar Tripathy and B. Behera, "Hyperparameter Tuning-based Optimized Analysis of Machine Learning Algoritma For Network Intrusion Detection," *International Journal of Network Security & Its Applications*, vol. 17, no. 6, pp. 01–26, Nov. 2025, doi: 10.5121/ijnsa.2025.17601.
- [18] T. H. Chua and I. Salam, "Evaluation of Machine Learning Algorithms in Network-Based Intrusion Detection Using Progressive Dataset," *Symmetry (Basel)*, vol. 15, no. 6, Jun. 2023, doi: 10.3390/sym15061251.

-
- [19] D. P. Sari, Z. Halim, I. Irlon, B. Waseso, and S. Saromah, "Implementasi Machine Learning untuk Deteksi Intrusi pada Jaringan Komputer," *Jurnal Minfo Polgan*, vol. 13, no. 2, pp. 1389–1394, Sep. 2024, doi: 10.33395/jmp.v13i2.14074.
- [20] M. T. Abdelaziz *et al.*, "Enhancing Network Threat Detection with Random Forest-Based NIDS and Permutation Feature Importance," *Journal of Network and Systems Management*, vol. 33, no. 1, Mar. 2025, doi: 10.1007/s10922-024-09874-0.
- [21] A. Alqahtani and H. Alshaher, "Anomaly-Based Intrusion Detection Systems Using Machine Learning," *Journal of Cybersecurity and Information Management*, vol. 14, no. 1, pp. 20–33, 2024, doi: 10.54216/JCIM.140102.
- [22] Md. A. Talukder *et al.*, "Machine learning-based network intrusion detection for big and imbalanced data using oversampling, stacking feature embedding and feature extraction," Jan. 2024, [Online]. Available: <http://arxiv.org/abs/2401.12262>
- [23] A. Sapaatullah and M. Darip, "Implementasi One-Class Support Vector Machine untuk Deteksi Serangan Jaringan Kampus," *Bulletin of Information Technology (BIT)*, vol. 7, no. 1, pp. 40–49, 2026, doi: 10.47065/bit.v5i2.2602.
- [24] A. T. Zy, A. T. Sasongko, and A. Z. Kamalia, "Penerapan Naïve Bayes Classifier, Support Vector Machine, dan Decision Tree untuk Meningkatkan Deteksi Ancaman Keamanan Jaringan," *Media Online*, vol. 4, no. 1, pp. 610–617, 2023, doi: 10.30865/klik.v4i1.1134.
- [25] K. A. Nugroho, T. Hariguna, and A. S. Barkah, "Optimizing Early Network Intrusion Detection: A Comparison of LSTM and LinearSVC with SMOTE on Imbalanced Data," *Jurnal Teknik Informatika (Jutif)*, vol. 6, no. 6, pp. 5349–5370, Dec. 2025, doi: 10.52436/1.jutif.2025.6.6.4672.
- [26] H. Haeruddin, E. Erick, and H. W. Aripadono, "Perbandingan Support Vector Machine, Random Forest Classifier, dan K-Nearest Neighbour dalam Pendeteksian Anomali pada Jaringan DDos," *JTIM : Jurnal Teknologi Informasi dan Multimedia*, vol. 7, no. 1, pp. 23–33, Jan. 2025, doi: 10.35746/jtim.v7i1.628.
- [27] S. Rabbani and D. Diana, "Prediksi Kategori Serangan Siber dengan Algoritma Klasifikasi Random Forest Menggunakan Rapidminer," *SMATIKA JURNAL*, vol. 13, no. 02, pp. 284–293, Dec. 2023, doi: 10.32664/smatika.v13i02.934.
- [28] E. Altulaihan, M. A. Almaiah, and A. Aljughaiman, "Anomaly Detection IDS for Detecting DoS Attacks in IoT Networks Based on Machine Learning Algorithms," *Sensors*, vol. 24, no. 2, Jan. 2024, doi: 10.3390/s24020713.