Jurnal Informatika: Jurnal pengembangan IT

Vol. 11, No. 2, 2026
ISSN : 2477-5126, DOI:10.30591/jpit.v11i2.10064

Optimasi Model XGBoost Dengan Seleksi Fitur Mutual Information Dan Threshold Tuning Untuk Deteksi Intrusi Jaringan

Vicola Nanda Pratama [1], Wildanil Ghozi [2]
[1,2]Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro
[1]111202214240@mhs.dinus.ac.id, [2]wildanil.ghozi@dsn.dinus.ac.id

Info Artikel

Riwayat Artikel:

Received 2026-01-01
Revised 2026-05-02
Accepted 2026-06-08

Abstract *– Network Intrusion Detection Systems (NIDS) are essential for protecting networks from evolving cyber threats. Although Machine Learning can be relied upon for NIDS, challenges remain in achieving high accuracy while maintaining low false alarm rates. This research proposes an optimized NIDS framework using the Extreme Gradient Boosting (XGBoost) algorithm, which is enhanced through systematic feature selection and hyperparameter tuning. The methodology integrates a two-stage feature selection process that combines ExtraTreesClassifier for initial importance analysis and SelectKBest with mutual information for identifying the optimal feature subset. Hyperparameter optimization is performed using RandomizedSearchCV with 5-fold cross-validation, followed by threshold calibration to balance the False Positive Rate (FPR) and False Negative Rate (FNR). The model is trained and evaluated on the UNSW-NB15 dataset, which contains 257,673 network traffic records with binary classification (normal vs. attack). The results of the experiment show that the optimized XGBoost model achieved an accuracy of 95.4%, precision of 94.81%, recall of 95.29%, F1-score of 95.04%, and a significantly reduced FPR of 5.09%. The feature selection process identified 37 most informative features from the original 42 features, which contributed to improved model performance and efficiency. These findings indicate that an integrated approach of adaptive feature selection and systematic model optimization effectively improves intrusion detection performance, offering a robust and balanced solution for modern network security applications.*

Keywords: *Feature Selection; Hyperparameter Tuning; Network Intrusion Detection System; UNSW-NB15; XGBoost*

Corresponding Author:
Wildanil Ghozi
Email:
wildanil.ghozi@dsn.dinus.ac.id

Abstrak *– Sistem Deteksi Intrusi Jaringan (NIDS) sangat penting untuk melindungi jaringan dari ancaman siber yang terus berkembang. Meskipun pembelajaran mesin bisa diandalkan untuk NIDS, tantangan tetap ada dalam mencapai akurasi tinggi sekaligus mempertahankan tingkat alarm palsu yang rendah. Penelitian ini mengusulkan kerangka kerja NIDS teroptimasi menggunakan algoritma Extreme Gradient Boosting (XGBoost), yang ditingkatkan melalui seleksi fitur sistematis dan penyetelan hyperparameter. Metodologi mengintegrasikan proses seleksi fitur dua tahap yang menggabungkan ExtraTreesClassifier untuk analisis kepentingan awal dan SelectKBest dengan mutual information untuk identifikasi subset fitur optimal. Optimasi hyperparameter dilakukan menggunakan RandomizedSearchCV dengan validasi silang 5-fold, dilanjutkan dengan kalibrasi threshold untuk menyeimbangkan False Positive Rate (FPR) dan False Negative Rate (FNR). Model dilatih dan dievaluasi pada dataset UNSW-NB15, yang berisi 257,673 catatan lalu lintas jaringan dengan klasifikasi biner (normal vs. serangan). Hasil eksperimen menunjukkan bahwa model XGBoost teroptimasi mencapai akurasi 95.4%, presisi 94.81%, recall 95.29%, F1-score 95.04%, dan FPR yang berkurang signifikan menjadi 5.09%. Proses seleksi fitur mengidentifikasi 37 fitur paling informatif dari 42 fitur asli, yang berkontribusi pada peningkatan performa dan efisiensi model. Temuan ini menunjukkan bahwa pendekatan terintegrasi dari seleksi fitur adaptif dan optimasi model sistematis secara efektif meningkatkan performa deteksi intrusi, menawarkan solusi yang kuat dan seimbang untuk aplikasi keamanan jaringan modern.*

Kata Kunci*: Seleksi Fitur, Hyperparameter Tuning, Sistem Deteksi Intrusi Jaringan, UNSW-NB15, XGBoost*

I. PENDAHULUAN

Perkembangan teknologi yang cepat di bidang Internet, IoT, dan sistem komunikasi telah meningkatkan kerentanan siber dengan memungkinkan serangan yang semakin kompleks[1]. Jaringan komputer sebagai tulang punggung pertukaran data menghadapi ancaman keamanan informasi yang terus mengintai, menjadikan keamanan siber sebagai kebutuhan yang tidak dapat ditawar[2]. Ketergantungan pada infrastruktur digital telah meningkatkan frekuensi dan kecanggihan ancaman siber, mengancam stabilitas jaringan dan data sensitif[3], [4]. Tantangan ini juga muncul dari dalam proses pengembangan solusi, misalnya pengembangan sistem deteksi intrusi berbasis pembelajaran mesin sering terhambat oleh masalah seperti ketidakseimbangan kelas (*class imbalance*) dan tumpang tindih antar kelas (*class overlap*) yang dapat menurunkan performa klasifikasi secara drastis[5].

Dalam konteks sistem *Industrial Internet of Things* (IIoT) yang kompleks, pengembangan teknik deteksi anomali yang efektif menjadi tantangan mendesak untuk memitigasi kerugian dari serangan berbahaya[6]. Solusi tradisional seperti *Intrusion Detection System* (IDS) berperan sebagai mekanisme pertahanan untuk memantau, mendeteksi, dan merespons aktivitas tidak sah, serta serangan siber[7]. Namun, pertumbuhan eksponensial perangkat IoT dan volume lalu lintas jaringan telah meningkatkan kompleksitas ancaman, membuat teknik mitigasi tradisional menjadi tidak efisien[8]. IDS konvensional berbasis tanda tangan (*signature-based*) memiliki kelemahan utama tidak mampu mendeteksi serangan baru (*zero-day attacks*)[9]. Pendekatan lain berbasis anomali (*anomaly-based*) dengan teknik *rule-based* juga memiliki keterbatasan dalam menemukan anomali yang beragam dan canggih[10]. Hal ini mendorong adopsi pendekatan berbasis kecerdasan buatan, khususnya *Machine Learning*, dalam pengembangan IDS modern[7], [8]. Algoritma *Machine Learning* unggul dalam memproses data berdimensi tinggi, mengenali pola kompleks, dan beradaptasi dengan perubahan, sehingga cocok untuk mendeteksi anomali jaringan yang tidak diketahui sebelumnya[8], [10]. Berbagai model seperti Random Forest (RF) dan XGBoost telah menunjukkan kinerja klasifikasi yang kuat untuk deteksi intrusi[9], [10].

Pendekatan berbasis *Machine Learning* semakin mendapat perhatian signifikan dalam pengembangan IDS modern[11]. Algoritma ML memanfaatkan kemampuannya belajar dari data, mengenali pola, dan membuat keputusan berbasis data untuk meningkatkan deteksi terhadap serangan baru[11]. Penelitian sering berfokus pada perbandingan algoritma klasifikasi seperti KNN dan *Decision Tree* untuk menentukan metode terbaik dalam mengenali ancaman[12]. Tren terkini mengeksplorasi model yang lebih canggih seperti *Large Language Models* (LLM) dengan pendekatan prompt-only yang tidak memerlukan pelatihan ulang intensif[13]. Eksperimen terhadap berbagai model dan varian hibridanya menunjukkan bahwa integrasi ML ke dalam IDS memanfaatkan kemampuannya mempelajari pola dari data kompleks[14]. eknik supervised *Machine Learning* banyak dieksplorasi untuk klasifikasi biner dan multi-kelas dalam deteksi intrusi, dengan metode rekayasa fitur seperti *Recursive Feature Elimination* (RFE) untuk mengurangi dimensi data[15].

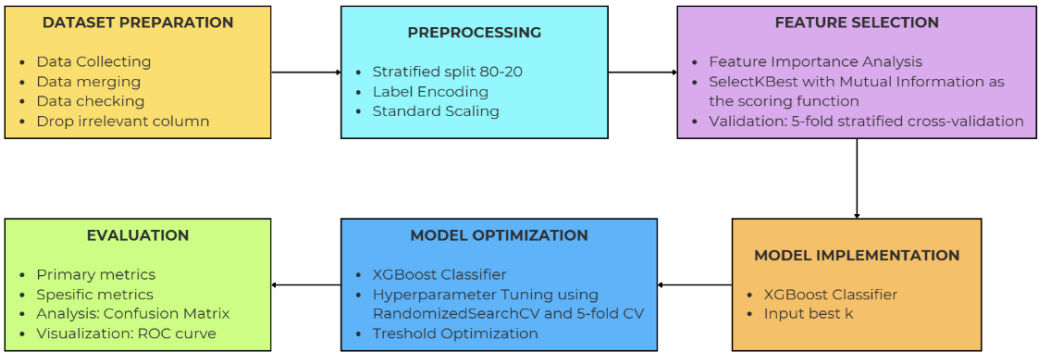
Sebagai benchmark utama, dataset UNSW-NB15 telah menjadi pilihan berbagai penelitian untuk menguji efektivitas model deteksi intrusi, menggantikan dataset lama seperti KDD99[16]. Berbagai metode telah diusulkan dengan capaian dan keterbatasan masing-masing. Penelitian Mebawondu et al.[3] mengimplementasikan CNN dengan akurasi menjanjikan, namun arsitektur deep learning ini cenderung kompleks secara komputasi dan kurang optimal dalam menangani ketidakseimbangan kelas. Pendekatan dengan DANets meningkatkan akurasi namun masih menghasilkan false positive yang tinggi[4]. Penelitian Pal et al.[9] mendemonstrasikan bahwa seleksi fitur berbasis XGBoost dapat secara signifikan mengurangi dimensi data dan menunjukkan bahwa seleksi fitur berbasis XGBoost dapat mengurangi dimensi dan waktu komputasi, namun masih menyisakan ruang peningkatan akurasi dan belum melakukan optimasi hyperparameter mendalam. Sementara itu, arsitektur hibrid CNN+RF berhasil mendorong akurasi lebih tinggi, namun sulit direplikasi karena deskripsi yang tidak rinci dan memiliki keterbatasan generalisasi[2]. Secara keseluruhan, tantangan utama adalah merancang model yang akurat, efisien, memiliki tingkat kesalahan rendah, mudah diinterpretasi, dan dapat digeneralisasi[17], [18], [19], [20].

Berdasarkan identifikasi celah tersebut, studi ini mengusulkan pendekatan optimasi komprehensif untuk meningkatkan performa sistem deteksi intrusi berbasis klasifikasi biner. Ide kuncinya adalah integrasi strategi seleksi fitur berbasis mutual information, optimasi hyperparameter sistematis, dan kalibrasi threshold dinamis pada kerangka XGBoost yang telah diakui efektif untuk tugas klasifikasi dalam IDS[7], [9]. Pendekatan ini dirancang untuk mengatasi tantangan mendasar dalam IDS modern: seleksi fitur cerdas untuk data berdimensi tinggi[21], adaptasi terhadap evolusi pola serangan[22], [23], dan keseimbangan optimal antara akurasi deteksi dengan minimalisasi false alarm[24].

Meskipun penelitian terdahulu pada dataset UNSW-NB15 telah menggunakan berbagai pendekatan, masih terdapat celah yang belum teratasi secara simultan. Pendekatan deep learning cenderung kompleks secara komputasi, memiliki *False Positive Rate* tinggi, dan sulit direplikasi. Sementara itu, pendekatan XGBoost yang lebih efisien masih terbatas pada seleksi fitur tanpa optimasi hyperparameter mendalam dan belum mempertimbangkan keseimbangan antara *False Positive Rate* dan *False Negative Rate*. Oleh karena itu, penelitian ini menawarkan pendekatan optimasi komprehensif pada XGBoost yang mengintegrasikan seleksi fitur berbasis mutual information, optimasi hyperparameter dengan RandomizedSearchCV, dan kalibrasi threshold dinamis untuk menghasilkan model deteksi intrusi yang akurat, efisien, memiliki false alarm rendah, dan siap diimplementasikan.

II. METODE

Metode penelitian ini dilakukan untuk mengembangkan sistem deteksi intrusi jaringan yang optimal dengan memanfaatkan algoritma XGBoost dan seleksi fitur mutual information kelas binary pada dataset UNSW-NB15. Penelitian ini mengimplementasikan pipeline sistematis yang terdiri dari enam tahapan utama seperti yang ditunjukkan pada Gambar 1.



Gambar 1. Alur Metode Penelitian

Alur penelitian dimulai dari pengumpulan dataset UNSW-NB15 yang diunduh dari Kaggle, dilanjutkan dengan preprocessing data yang meliputi pembersihan data duplikat, encoding fitur kategorikal, dan normalisasi. tahapan selanjutnya melakukan seleksi fitur menggunakan mutual information untuk menentukan subset fitur optimal, diikuti dengan optimasi hyperparameter XGBoost menggunakan RandomizedSearchCV, dan diakhiri dengan evaluasi komprehensif menggunakan berbagai metrik klasifikasi.

A. Dataset Preparation

Penelitian ini menggunakan dataset UNSW-NB15. Dataset ini merupakan campuran khusus antara aktivitas serangan sintetis modern dan lalu lintas jaringan modern yang nyata[4][10]. Dataset ini diperoleh dari Kaggle melalui tautan www.kaggle.com/datasets/mrwellsdavid/unsw-nb15. Dataset ini dikembangkan oleh University of New South Wales (UNSW) dan terdiri dari dua file utama *UNSW_NB15_training-set.csv* (82.332 record) dan *UNSW_NB15_testing-set.csv* (175.341 record). Kedua file tersebut digabungkan menjadi satu dataset yang berisi total 257.673 record. Setiap record dalam dataset memiliki 45 fitur yang merepresentasikan karakteristik teknis koneksi jaringan, termasuk fitur kategorikal seperti jenis protokol, layanan jaringan, dan status koneksi, serta fitur numerik yang menggambarkan metrik lalu lintas seperti jumlah paket, byte, dan statistik waktu seperti pada Tabel 1.[9].

TABEL 1
UKURAN DATASET

Komponen	Jumlah Record	Jumlah Feature
Training-set	82,332	45
Testing-set	175,341	45
Gabungan	257,673	45

Dataset ini berisi 9 macam serangan jaringan yaitu *Fuzzers*, *Analysis*, *Backdoors*, *DoS*, *Exploits*, *Generic*, *Reconnaissance*, *Shellcode*, dan *Worms*. Dalam penelitian ini, semua jenis serangan tersebut digabung menjadi satu kategori yaitu serangan (label 1), sedangkan jaringan normal menjadi kategori normal (label 0). Dataset ini memiliki fitur-fitur kategorikal seperti '*proto*' (jenis protocol), '*service*' (jenis layanan), dan '*state*' (status koneksi), serta fitur-fitur numerik yang merepresentasikan karakteristik teknis koneksi jaringan.

Setelah penggabungan, dilakukan pemeriksaan terhadap missing values dan duplikat data. Hasil pemeriksaan menunjukkan bahwa tidak terdapat *missing values* dalam dataset gabungan. Selanjutnya, dilakukan penghapusan kolom yang tidak relevan untuk klasifikasi biner. Kolom id dihapus karena hanya berfungsi sebagai identifier unik tanpa nilai prediktif. Kolom *attack_cat* juga dihapus karena penelitian ini berfokus pada klasifikasi biner (normal vs serangan), bukan klasifikasi multi-kelas berdasarkan jenis serangan. Setelah penghapusan, dataset tersisa dengan 43 fitur (40 fitur numerik dan 3 fitur kategorikal)[9].

B. Preprocessing

Proses preprocessing dilakukan secara sistematis melalui tiga tahap utama. Pembagian data, encoding data kategorikal, dan normalisasi data numerik. Pertama, dataset gabungan dibagi menjadi data training dan testing dengan rasio 80:20 menggunakan fungsi *train_test_split()* dari scikit-learn. Pembagian ini menggunakan stratified sampling untuk mempertahankan proporsi distribusi kelas asli antara data training dan testing.

Kedua, dilakukan transformasi data kategorikal menjadi format numerik menggunakan *Label Encoding*. Tiga fitur kategorikal diidentifikasi: *proto* (jenis protokol, 133 kategori), *service* (jenis layanan, 13 kategori), dan *state* (status koneksi, 11 kategori). Proses encoding dilakukan untuk mengubah mereka menjadi format numerik yang kompatibel dengan *Machine Learning*[7].

Ketiga, seluruh fitur numerik dinormalisasi menggunakan *StandardScaler* yang menerapkan standardisasi Z-score dengan rumus[12]:

$$z = \frac{x - \mu}{\sigma} \quad (1)$$

di mana x adalah nilai asli fitur, μ adalah mean (rata-rata) dari fitur tersebut, dan σ adalah standard deviation (simpangan baku) dari fitur tersebut. Proses standardisasi dilakukan secara terpisah untuk data training dan testing: parameter μ dan σ dihitung hanya dari data training, kemudian transformasi yang sama diterapkan ke data testing. Pendekatan ini mencegah terjadinya *data leakage* dan memastikan proses yang valid secara statistika. Hasil dari proses ini adalah distribusi data dengan mean = 0 dan standard deviation = 1 untuk setiap fitur numerik[7], [11].

C. Feature Selection

1) Extra Trees Classifier

Tahap pertama menggunakan metode *Extra Trees Classifier* dengan konfigurasi jumlah pohon sebanyak 200 untuk menganalisis tingkat kepentingan fitur (*feature importance*) secara komparatif. Extra Trees merupakan classifier yang kuat, meskipun performanya dapat menurun dengan adanya fitur-fitur yang tidak relevan[21]. Model *Extra Trees Classifier* dengan 200 pohon dilatih pada 42 fitur yang telah dinormalisasi untuk mengklasifikasikan traffic normal (label 0) dan serangan (label 1). Tingkat kepentingan setiap fitur dihitung berdasarkan kontribusinya dalam mengurangi tingkat ketidakmurnian pada pembagian data di seluruh pohon.

2) Seleksi dengan Mutual Information

Tahap kedua menggunakan mutual information (MI) untuk seleksi fitur yang lebih akurat. Mutual information mengukur ketergantungan statistik antara fitur independen dan variabel target dependen. Ketika digunakan untuk seleksi fitur, information gain disebut sebagai mutual information. Jika tidak ada hubungan antara fitur dan target, maka MI bernilai nol. Nilai MI yang lebih tinggi menandakan ketergantungan yang lebih kuat antara variabel target dan fitur yang bersangkutan dalam dataset. Secara matematis, mutual information antara fitur S dan variabel T dipresentasikan[25]:

$$MI(S,T) = H(T) - H(T|S) \quad (2)$$

Di mana H(T) Adalah entropi dari variable target T yang mengukur ketidakpastian awal, sedangkan H(T|S) adalah entropi bersyarat dari T setelah mengetahui nilai fitur S. Dengan demikian, MI mengukur seberapa besar ketidakpastian T berkurang setelah mengetahui S.

Seleksi fitur menggunakan SelectKBest dengan fungsi mutual_info_classif dari scikit-learn. SelectKBest merupakan metode *filter-based feature selection* yang memilih k fitur fitur dengan skor tertinggi berdasarkan fungsi skoring tertentu. Fungsi *mutual_info_classif* menghitung skor mutual information untuk setiap fitur terhadap label biner. Algoritma ini bekerja dengan menghitung skor MI untuk semua fitur, mengurutkan fitur berdasarkan skor, memilih k fitur dengan skor tertinggi untuk pemodelan[26]. Nilai k optimal ditentukan melalui validasi cross-validation.

3) Stratified 5-fold Cross-Validation

Penentuan nilai k optimal dilakukan menggunakan stratified 5-fold *cross-validation*. Data training dibagi menjadi 5 subset dengan proporsi kelas yang terjaga. Untuk setiap nilai k dari 10 hingga 42, fitur dipilih dengan SelectKBest berdasarkan skor mutual information. Model XGBoost dilatih dan dievaluasi 5 kali, dengan setiap iterasi menggunakan 4 subset untuk training dan 1 subset untuk validasi. Nilai k dengan akurasi rata-rata tertinggi dipilih sebagai optimal.

D. Model Implementation

1) XGBoost Untuk klasifikasi

XGBoost (*eXtreme Gradient Boosting*) merupakan algoritma ensemble learning berbasis gradient boosting yang dikenal efisien, fleksibel, dan ringkas, terutama dalam tugas klasifikasi dan regresi pada data terstruktur seperti dataset jaringan[6], [9]. Algoritma ini bekerja dengan membangun sekuensial pohon keputusan yang setiap pohon berikutnya berusaha mengoreksi kesalahan prediksi pohon sebelumnya. Keunggulan utama XGBoost terletak pada kemampuannya melakukan optimasi fungsi tujuan (*objective function*) menggunakan ekspansi Taylor orde kedua, yang mempercepat konvergensi dan meningkatkan akurasi. Secara matematis, fungsi tujuan dalam XGBoost pada iterasi ke-t didefinisikan pada rumus(3)[6]:

$$\zeta^{(t)} \approx \sum_{i=1}^{n}\left[\mathrm{l}\left(y_i, \hat{y}_i^{(t-1)}\right) + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i)\right] + \Omega(f_t) \quad (3)$$

Dimana ζ menjadi fungsi target, f menjadi parameter fitur kompleksitas[6]. Kombinasi penggunaan informasi gradien orde kedua dan term regularisasi membuat XGBoost tidak hanya cepat dalam pelatihan tetapi juga memiliki kemampuan generalisasi yang tinggi[6], [9].

2) Implementasi model XGBoost

Dalam penelitian ini, XGBoost diimplementasikan menggunakan XGBClassifier dari pustaka XGBoost dengan konfigurasi awal seperti yang ditunjukkan pada tabel. Algoritma XGBoost dipilih karena telah terbukti efisien, fleksibel, dan ringkas untuk tugas klasifikasi data tabular seperti dataset UNSW-NB15 [6], [9]. Model ini berperan ganda: pertama, sebagai evaluator dalam proses seleksi fitur berbasis mutual information untuk menentukan jumlah fitur optimal (k=37) berdasarkan akurasi cross-validation. Kedua, sebagai model dasar sebelum dilakukan optimasi hiperparameter lebih lanjut menggunakan RandomizedSearchCV. Input model adalah subset fitur terpilih dari tahap seleksi fitur, sedangkan outputnya berupa probabilitas klasifikasi biner dengan fungsi loss logistik biner (*binary logistic loss*), di mana nilai 0 menunjukkan traffic normal dan nilai 1 mengindikasikan aktivitas serangan[9]. Setelah melalui tahap optimasi hiperparameter dan kalibrasi threshold (menjadi 0.520), model akhir dilatih pada subset fitur terpilih dan siap dievaluasi pada data uji dengan parameter pada Tabel 2.

TABEL 2
PARAMETER SELEKSI FITUR

Parameter	Nilai	Keterangan
n_estimators	300	Jumlah Pohon dalam ensemble
learning_rate	0.1	Kedalaman maksimum tiap pohon
max_depth	6	Tingkat pembelajaran
subsample	1.0	Fraksi sampel setiap pohon
colsample_bytree	1.0	Fraksi fitur setiap pohon
reg_alpha	0	Regularisasi L1
reg_lambda	1	Regularisasi L2

E. Model Optimitazion

1) Hyperparameter Tuning

Proses optimasi hyperparameter menggunakan RandomizedSearchCV dengan 60 iterasi pencarian dan validasi 5-fold stratified cross-validation. Ruang parameter yang dieksplorasi mencakup tujuh parameter utama XGBoost seperti ditunjukkan pada Tabel 3.

TABEL 3
PARAMETER OPTIMASI MODEL

Parameter	Nilai
n_estimators	400-1000
learning_rate	0.01-0.2
max_depth	6-12
subsample	0.7-1.0
colsample_bytree	0.7-1.0
reg_alpha	0-1
reg_lambda	0-1

2) Threshold Optimization

Setelah konfigurasi hiperparameter optimal diperoleh, dilakukan kalibrasi threshold klasifikasi menggunakan probabilitas prediksi dari 5-fold *cross-validation* pada data pelatihan. Berdasarkan distribusi probabilitas hasil cross-validation yang seluruhnya berada di sekitar 0.5, rentang *threshold* yang dievaluasi ditetapkan dari 0.48 hingga 0.52 dengan interval 0.005, meliputi sembilan nilai: 0.480, 0.485, 0.490, 0.495, 0.500, 0.505, 0.510, 0.515, dan 0.520. Pada setiap nilai *threshold*, probabilitas dikonversi menjadi label biner dengan ketentuan probabilitas ≥ *threshold* diklasifikasikan sebagai serangan (kelas 1), sedangkan probabilitas < threshold diklasifikasikan sebagai normal (kelas 0).

Akurasi digunakan sebagai metrik pemilih *threshold* karena dataset memiliki tingkat ketidakseimbangan kelas yang moderat (rasio serangan:normal = 1.77:1), sehingga akurasi tetap representatif. *Threshold* yang menghasilkan akurasi validasi tertinggi dipilih sebagai *threshold* optimal dan kemudian diterapkan pada probabilitas prediksi model final untuk konversi menjadi label biner pada evaluasi data pengujian.

3) Final Model Configuration

Model final dilatih menggunakan konfigurasi optimal yang diperoleh dari proses hyperparameter tuning dan threshold optimization. Konfigurasi ini diterapkan pada subset fitur terpilih untuk pelatihan dan evaluasi akhir.

F. Evaluation

Evaluasi performa model dilakukan secara komprehensif menggunakan berbagai metrik evaluasi yang memberikan perspektif berbeda tentang kualitas model. False Positive (FP) adalah lalu lintas normal yang salah diklasifikasikan sebagai serangan, sedangkan False Negative (FN) adalah serangan yang salah diklasifikasikan sebagai normal. Metrik utama yang digunakan meliputi akurasi, precision, recall, dan F1-score untuk mengukur kinerja klasifikasi secara keseluruhan. Selain itu, digunakan juga ROC-AUC score untuk mengevaluasi kemampuan model dalam membedakan antara kedua kelas.

1) *Accuracy*: Mengukur persentase prediksi benar dari total prediksi klasifikasi [27]

$$\frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

2) *Precision*: Mengukur prediksi serangan sebenarnya dari total serangan diterima [27]

$$\frac{TP}{TP + FP} \tag{5}$$

3) *Recall*: Mengukur kemampuan menemukan serangan sebenarnya dari klasifikasi dengan jumlah total true positive dan false negative [27]

$$\frac{TP}{TP + FN} \tag{6}$$

4) *F1-Score*: Mencari rata-rata precision dan recall [27]

$$2 \times \frac{Precison \times Recall}{Precision + Recall} \tag{7}$$

5) FPR: Mengukur proporsi traffic normal yang salah diklasifikasikan sebagai serangan[18]

$$\frac{FP}{FP + TN} \tag{8}$$

Analisis performa dilengkapi dengan pengukuran *False Positive Rate* (FPR) dan *False Negative Rate* (FNR) untuk memahami karakteristik error model. Dalam konteks deteksi intrusi, analisis FPR dan FNR sangat kritis karena kedua jenis error tersebut memiliki implikasi operasional yang berbeda. False positive dapat mengakibatkan gangguan operasional akibat alarm yang tidak perlu, sementara false negative berpotensi membiarkan serangan tidak terdeteksi.

Evaluasi model menggunakan confusion matrix untuk menganalisis distribusi prediksi across kedua kelas. Analisis ini memberikan pemahaman mendalam tentang pola kesalahan klasifikasi dan membantu mengidentifikasi area yang perlu diperbaiki. Threshold klasifikasi yang telah dioptimasi sebelumnya digunakan dalam evaluasi akhir untuk memastikan keseimbangan yang tepat antara sensitivitas dan spesifisitas model.

Proses evaluasi dilakukan pada data testing yang tidak digunakan selama training maupun optimasi model, sehingga memberikan estimasi yang unbiased terhadap kemampuan generalisasi model. Hasil evaluasi komprehensif merepresentasikan kinerja model dalam skenario deteksi intrusi yang realistis, dengan mempertimbangkan berbagai aspek performa yang relevan untuk penerapan dalam lingkungan operasional.

III. HASIL DAN PEMBAHASAN

A. Hasil Preprocessing Data

1) Identifikasi pada Dataset awal

Dataset UNSW-NB15 yang digunakan terdiri dari total 257,673 record dengan 45 fitur setiap record. Dataset ini menggabungkan data training awal (82,332 record) dan data testing awal (175,341 record) yang tersedia. Analisis distribusi label menunjukkan komposisi yang tidak seimbang dengan 164,673 record (63.91%) termasuk dalam kelas serangan (label 1) dan 93,000 record (36.09%) yang merupakan traffic normal (label 0). Pemeriksaan missing values mengkonfirmasi bahwa tidak ada nilai yang hilang pada keseluruhan dataset.

Pembagian data dilakukan menggunakan fungsi *train_test_split()* dengan parameter test_size=0.2 dan *stratify=df['label']*. Metode stratified random sampling ini memastikan proporsi kelas asli tetap terjaga pada kedua subset data. Hasil pembagian menghasilkan data training sebanyak 206,138 record dan data testing sebanyak 51,535 record, dengan komposisi kelas yang konsisten seperti ditunjukkan pada Tabel 4.

TABEL 4
HASIL IDENTIFIKASI DATASET

Komponen	Jumlah Record	Serangan (1)	Normal (0)
Data Training	206,138	131,738	74,400
Data Testing	51,535	32,935	18,600
Total Dataset	257,673	164,673	93,000

Distribusi kelas yang tidak seimbang (rasio serangan:normal = 1.77:1) menunjukkan potensi bias model terhadap kelas mayoritas. Namun, penggunaan stratified sampling telah menjaga proporsi ini tetap konsisten pada data training dan testing, sehingga model tetap dapat belajar dari kedua kelas secara proporsional.

2) Transformasi dan Feature Encoding

Proses preprocessing diawali dengan penghapusan kolom id dan attack_cat yang menghasilkan dataset dengan total 43 fitur, terdiri atas 40 fitur numerik dan 3 fitur kategorikal (proto, service, dan state). Transformasi fitur kategorikal menggunakan Label Encoding menghasilkan 133 kategori pada proto, 13 kategori pada service, dan 11 kategori pada state. Normalisasi fitur numerik dengan StandardScaler mentransformasi distribusi data ke dalam skala seragam dengan mean 0 dan standard deviation 1, diterapkan secara terpisah pada data training dan testing. Dimensi akhir dataset disajikan pada Tabel 5.

TABEL 5
HASIL TRANSFORMASI DAN FEATURE ENCODING

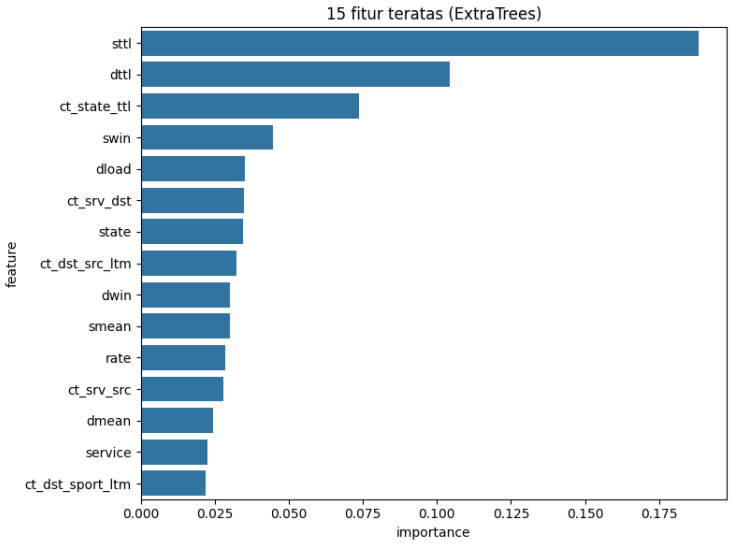
Komponen	Jumlah Record	Jumlah Fitur
Data Training	206,138	42
Data Testing	51,535	42

Seluruh fitur telah berada dalam format numerik dengan skala yang seragam. Pemisahan parameter normalisasi antara data training dan testing mencegah terjadinya data leakage, sehingga evaluasi model nantinya mencerminkan kemampuan generalisasi yang valid.

B. Analisis dan Seleksi Feature

1) Evaluasi Feature importance dengan Extra Trees Classifier

Dengan konfigurasi 200 estimator, analisis *feature importance* menghasilkan rentang skor dari 0.1884 hingga 0.0037. Fitur *sttl* (source time to live) menjadi fitur paling informatif (0.1884), diikuti *dttl* (destination time to live) dengan skor 0.1043, dan *ct_state_ttl* (state time to live count) dengan skor 0.0738 seperti ditunjukkan pada Gambar 2.

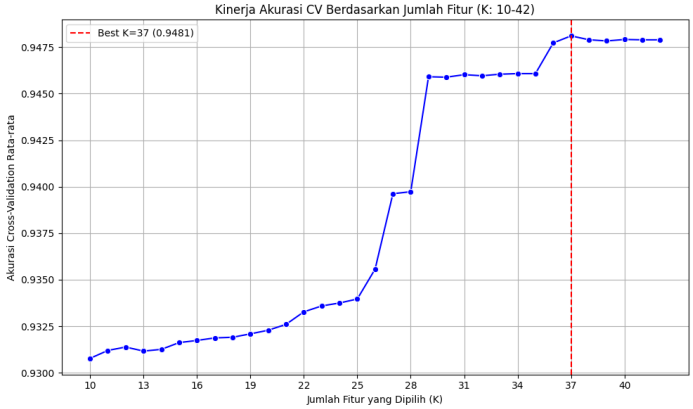


Gambar 2. Hasil *Feature importance* dari Extra Trees Clasiffier

Distribusi skor importance menunjukkan bahwa lima fitur teratas menyumbang 52.04% dari total *importance score*, sementara sepuluh fitur teratas menyumbang 69.94%. Hal ini menunjukkan bahwa kontribusi fitur tidak merata, di mana sebagian kecil fitur memiliki pengaruh dominan. Dominasi fitur *sttl, dttl*, dan *ct_state_ttl* mengindikasikan bahwa karakteristik waktu hidup paket (time to live) dan interaksi antar host merupakan pembeda utama antara lalu lintas normal dan serangan, sesuai dengan karakteristik serangan seperti DoS atau scanning yang cenderung menghasilkan pola TTL tidak wajar.

2) Optimasi Seleksi Fitur dengan Mutual Information

Evaluasi jumlah fitur optimal dilakukan pada 33 kandidat nilai k dari 10 hingga 42 menggunakan 5-fold cross-validation. Hasil evaluasi menunjukkan peningkatan akurasi seiring bertambahnya jumlah fitur, dengan titik optimal pada k=37 yang menghasilkan akurasi *cross-validation* tertinggi 94.81% seperti pada Gambar 3.



Gambar 3. Kinerja Akurasi CV k = 10 - 42

Perbedaan akurasi antara nilai k optimal (37) dengan menggunakan seluruh fitur (42) adalah 0.02%. Pola peningkatan akurasi hingga k=37 menunjukkan bahwa fitur-fitur tambahan hingga titik tersebut berkontribusi positif. Stagnasi akurasi setelah k=37 mengindikasikan bahwa 5 fitur sisanya bersifat redundan atau tidak informatif. Eliminasi fitur-fitur ini tidak berdampak signifikan terhadap performa, namun berhasil mengurangi dimensi data.

3) Hasil Seleksi Fitur

Berdasarkan optimasi, dipilih 37 fitur dari 42 fitur awal. Tabel 6 menyajikan sepuluh fitur dengan skor mutual information tertinggi, di mana *sbytes* (0.4551), *dbytes* (0.3404), dan *sload* (0.3377) menempati peringkat teratas. Sepuluh fitur ini menyumbang 70.15% dari total skor keseluruhan. Seleksi fitur berhasil mengidentifikasi subset yang mencakup 13 fitur traffic/volume data, 8 fitur waktu/statistik, 7 fitur pola koneksi, 5 fitur parameter TCP, dan 4 fitur kategorikal seperti pada Tabel 6.

TABEL 6
SKOR MUTUAL INFORMATION

Fitur	Skor MI
sbytes	0.4551
dbytes	0.3404
sload	0.3377
smean	0.3345
ct_state_ttl	0.3241
sttl	0.3215
dttl	0.3151
rate	0.3100
dur	0.2997
dmean	0.2855

Tingginya skor MI pada *sbytes*, *dbytes*, dan *sload* mengindikasikan bahwa volume lalu lintas (*bytes* dan *load*) merupakan indikator paling kuat dalam membedakan serangan dari lalu lintas normal. Hal ini sesuai dengan karakteristik serangan jaringan yang umumnya melibatkan transfer data dalam jumlah besar atau pola trafik tidak wajar. Reduksi dimensi sebesar 11.90% (42 ke 37 fitur) berhasil mempertahankan 99.98% akurasi maksimum.

C. Hasil Pemodelan dan Optimasi

1) Optimasi Hyperparameter dengan RandomizedSearchCV

Proses optimasi hyperparameter dilakukan secara sistematis menggunakan RandomizedSearchCV dengan 60 iterasi dan validasi silang 5-fold pada subset fitur optimal (K=37). Eksperimen ini berhasil mengidentifikasi konfigurasi optimal XGBoost yang meningkatkan performa model dengan nilai optimal pada Tabel 7.

TABEL 7
HASIL OPTIMASI HYPERPARAMETER

Parameter	Nilai Optimal	Rentang
n_estimators	437	400-1000
max_depth	11	6-12
learning_rate	0.060	0.01-0.2
subsample	0.972	0.7-1.0
colsample_bytree	0.891	0.7-1.0
reg_alpha	0.979	0-1
reg_lambda	0.487	0-1

Nilai *reg_alpha* yang mendekati 1 (0.979) menunjukkan regularisasi L1 bekerja aktif, sehingga model cenderung lebih ringkas dengan mengurangi bobot fitur yang kurang penting. *learning_rate* moderat (0.06) dikombinasikan dengan *n_estimators* cukup besar (437) menghasilkan model yang stabil. Nilai subsample (0.972) dan colsample_bytree (0.891) yang mendekati 1 mengindikasikan bahwa model menggunakan hampir seluruh data dan fitur pada setiap pohon, menandakan tidak ada indikasi overfitting yang serius.

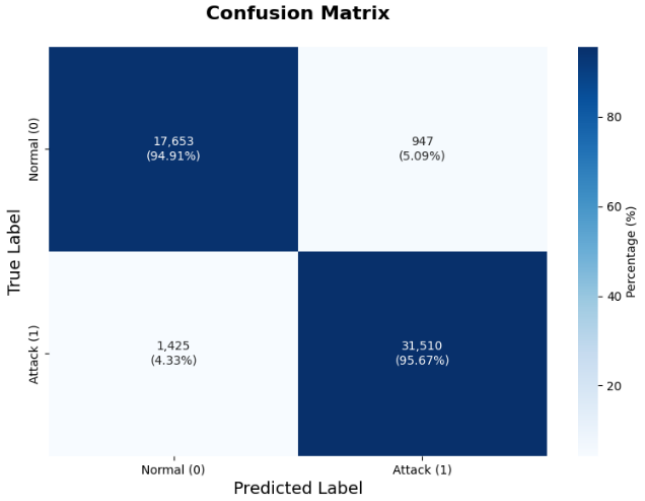
2) Threshold Optimization dan Evaluasi Model Final

Analisis threshold optimal dilakukan melalui cross-validation. Evaluasi pada rentang threshold 0,48 hingga 0,52 mengidentifikasi 0,52 sebagai titik optimal, sedikit lebih tinggi dari threshold default 0,5. Setelah threshold optimal ditetapkan, model dievaluasi pada data pengujian. Tabel 8 menyajikan hasil evaluasi.

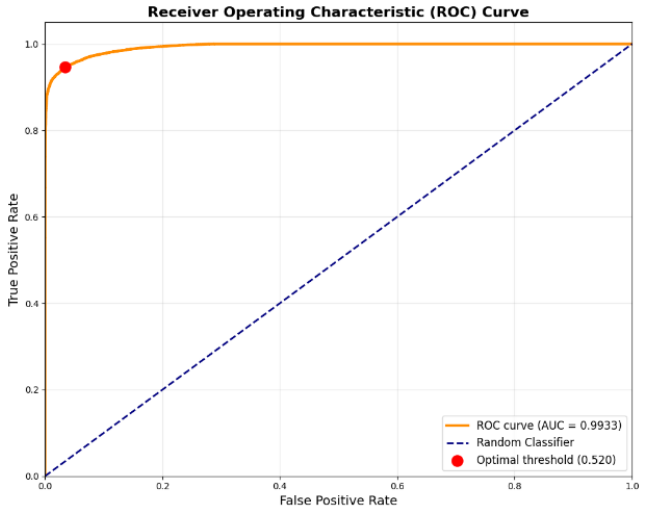
TABEL 8
AKURASI KESELURUHAN

Metrik	Normal (0)	Serangan (1)	Macro Avg	Weighted Avg
Precision	92.53	97.08	94.81	95.44
Recall	94.91	95.67	95.29	95.40
F1-Score	93.70	96.37	95.04	95.41
Support	18,600	32,935	51,535	51,535

Model mencapai akurasi keseluruhan 95.40% dengan ROC-AUC score 99.33% (Gambar 5). Confusion matrix pada Gambar 4 menunjukkan 31,510 true positive, 17,653 true negative, 947 false positive, dan 1,425 false negative.



Gambar 4. Confusion Matrix



Gambar 5. ROC Curve

3) Analisis Metrik Evaluasi Evaluasi

Hasil evaluasi menunjukkan bahwa model XGBoost yang telah melalui tahap seleksi fitur berbasis mutual information dan optimasi hiperparameter berhasil mencapai kinerja klasifikasi yang sangat baik pada data pengujian. Model ini mampu menjaga keseimbangan antara kemampuan mendeteksi ancaman (recall) dan keakuratan deteksi (precision), yang tercermin dari metrik-metrik evaluasi pada Tabel 9.

TABEL 9
METRIK EVALUASI

Overall Accuracy	**ROC-AUC Score**	***False Positive Rate* (FPR)**	***False Negative Rate* (FNR)**
95.40 %	99.33 %	5.09 %	4.33 %

Berdasarkan hasil evaluasi, model mencapai *precision macro average* sebesar 94.81% dan r*ecall macro average* sebesar 95.29%. Nilai F1-score *macro average* mencapai 95.04%. Secara spesifik untuk kelas serangan, model menghasilkan precision 97.08%, recall 95.67%, dan F1-score 96.37%. Pada kelas normal, model menghasilkan precision 92.53%, recall 94.91%, dan F1-Score 93.70%. Akurasi keseluruhan model adalah 95.40% dengan skor ROC-AUC 99.33%.

FNR (4.33%) yang lebih rendah dari FPR (5.09%) mencerminkan prioritas model dalam meminimalkan serangan yang tidak terdeteksi, meskipun dengan konsekuensi alarm palsu sedikit lebih tinggi. Ini sesuai

dengan kebutuhan IDS di mana *false negative* (serangan lolos) jauh lebih berbahaya daripada *false positive* (alarm palsu).

4) Perbandingan Internal Model

Untuk melihat perbedaan kinerja model, dilakukan perbandingan antara model sebelum (Baseline model) dan sesudah proses optimasi. Hasil perbandingan disajikan pada Tabel 10.

TABEL 10

PERBANDINGAN INTERNAL

Model	Akurasi (%)	F1-Score (%)
Sebelum Optimasi	94.89	95.97
Sesudah Optimasi	95.40	95.04

Model setelah optimasi menunjukkan peningkatan akurasi dari 94.89% menjadi 95.40%. Di sisi lain, F1-Score mengalami penurunan kecil dari 95.97% menjadi 95.04%. Peningkatan akurasi menunjukkan bahwa model hasil optimasi lebih baik dalam menghasilkan prediksi yang benar secara keseluruhan. Sementara itu, penurunan F1-Score yang relatif kecil mengindikasikan adanya perubahan keseimbangan performa antar kelas. Secara keseluruhan, hasil ini menunjukkan bahwa model setelah optimasi memiliki kinerja yang lebih stabil dengan peningkatan pada akurasi tanpa penurunan signifikan pada F1-Score

D. Keterbatasan Penelitian

Berdasarkan pelaksanaan eksperimen, terdapat beberapa keterbatasan pada penelitian ini. Evaluasi model hanya dilakukan pada satu dataset (UNSW-NB15). Meskipun dataset ini representatif untuk lalu lintas jaringan modern, generalisasi model terhadap dataset lain seperti CIC-IDS2017, NSL-KDD, atau data lalu lintas jaringan dari lingkungan yang berbeda belum teruji.

Kedua, pencarian threshold optimal dilakukan pada rentang 0,48 hingga 0,52. Rentang ini dipilih karena seluruh probabilitas hasil cross-validation berada di sekitar nilai 0,5, sehingga pencarian diluar rentang tersebut tidak dilakukan. Konsekuensinya, kemungkinan terdapat nilai threshold optimal di luar rentang yang terlewatkan.

E. Perbandingan dengan State-Of-The-Art

Untuk mengevaluasi posisi dan kontribusi penelitian ini, dilakukan perbandingan kinerja model dengan empat penelitian *State-Of-The-Art* (SOTA) terkini yang juga menggunakan dataset UNSW-NB15 untuk klasifikasi biner. Perbandingan berfokus pada metode yang digunakan serta metrik evaluasi kunci, yaitu akurasi, F1-Score, dan *False Positive Rate* (FPR). Ringkasan perbandingan disajikan dalam Tabel 11.

TABEL 11
KOMPARASI HASIL

Penelitian	Metode	Akurasi (%)	F1-Score (%)
[3]	CNN	81.07	85.18
[4]	DANets	82.95	77.74
[9]	XGBoost + FS	87.56	87.00
[2]	Hybrid CNN + RF	93.00	91.50
Diusulkan	**XGBoost + MI FS + Optimasi**	**95.40**	**95.04**

Berdasarkan tabel, model berbasis *Convolutional Neural Network* (CNN) dan *Deep Abstract Networks* (DANets) masing-masing mencapai akurasi sebesar 81.07% dan 82.95%. Pendekatan XGBoost dengan seluruh fitur berhasil meningkatkan akurasi menjadi 87.56%, sementara arsitektur Hybrid CNN + Random Forest mencatatkan akurasi tertinggi di antara penelitian pembanding, yaitu 93.00%.

Model yang diusulkan dalam penelitian ini, yang mengimplementasikan XGBoost dengan seleksi fitur Mutual Information dan optimasi hyperparameter sistematis, berhasil mengungguli semua penelitian pembanding. Model ini mencapai akurasi tertinggi sebesar 95.40%, yang merupakan peningkatan sebesar 2.40% dari penelitian terdekat (Hybrid CNN+RF). Selain itu, model ini juga mencatatkan F1-Score tertinggi sebesar 95.04%, yang mengindikasikan keseimbangan kinerja antara precision dan recall yang lebih baik dibandingkan metode lainnya. Hasil ini menunjukkan bahwa pendekatan optimasi komprehensif pada algoritma XGBoost efektif dalam meningkatkan akurasi klasifikasi dan kualitas prediksi binary class pada dataset UNSW-NB15.

IV. SIMPULAN

Penelitian ini berhasil mengembangkan sistem deteksi intrusi jaringan berbasis optimasi komprehensif algoritma XGBoost untuk klasifikasi biner. Pendekatan ini mengatasi tantangan keseimbangan antara akurasi deteksi (precision) dan kemampuan identifikasi serangan (recall) dengan menggunakan seleksi fitur berbasis Mutual Information yang mengidentifikasi 37 fitur informatif serta optimasi hiperparameter sistematis dengan RandomizedSearchCV. Evaluasi pada dataset UNSW-NB15 menunjukkan performa unggul dengan akurasi 95.40%, precision serangan 94.81%, recall 95.29%, F1-score 95.04%, serta tingkat kesalahan rendah (*False Positive Rate* 5.09 % dan *False Negative Rate* 4.33%). Hasil confusion matrix mengonfirmasi ketepatan klasifikasi dengan 31,510 true positive, 17,653 true negative, dan kesalahan minimal (947 false positive serta 1,425 false negative). Perbandingan dengan penelitian State-Of-The-Art menunjukkan bahwa model ini mengungguli pendekatan lain, termasuk metode yang lebih kompleks, dalam hal akurasi dan keseimbangan metrik. Dengan demikian, optimasi XGBoost yang terintegrasi berhasil meningkatkan efektivitas sistem deteksi intrusi, menjadikannya solusi yang siap implementasi untuk lingkungan jaringan yang dinamis. Untuk penelitian selanjutnya, disarankan untuk menguji model pada dataset yang berbeda seperti CIC-IDS2017 guna mengukur kemampuan generalisasi. Selain itu, pengembangan model ke dalam bentuk klasifikasi multi-kelas dapat dilakukan untuk mengidentifikasi jenis serangan spesifik. Pengujian pada lingkungan real-time juga perlu dilakukan untuk mengevaluasi kelayakan implementasi model dalam skenario deteksi intrusi yang sesungguhnya.

DAFTAR PUSTAKA

[1] N. Pansari, S. Srivastava, R. H. Raghavendra, and M. Agarwal, "Attack Classification using *Machine Learning* on UNSW-NB 15 dataset using XGBoost Feature Selection & Ablation Analysis," in *2024 IEEE 9th International Conference for Convergence in Technology, I2CT 2024*, Institute of Electrical and Electronics Engineers Inc., 2024. doi: 10.1109/I2CT61223.2024.10543523.

[2] V. L. Rismawan and E. Rahmawan Pramudya, "Network *Intrusion Detection System* Using Convolutional Neural Network And Random Forests Classifiers," *JURNAL INOVTEK POLBENG*, vol. 10, no. 2, pp. 753–761, 2025.

[3] O. J. Mebawondu, "Enhancing *Intrusion Detection System*s with Efficient Deep Learning Techniques," in *IEEE International Conference on Emerging and Sustainable Technologies for Power and ICT in a Developing Society, NIGERCON*, Institute of Electrical and Electronics Engineers Inc., 2024. doi: 10.1109/NIGERCON62786.2024.10927178.

[4] S. Kottilingal, "Deep Learning Based Network *Intrusion Detection System*: A Deep Abstract Networks (DANets) Model Approach," *IRJCS: International Research Journal of Computer Science*, vol. 11, no. 07, pp. 539–544, 2024, doi: 10.26562/irjcs.

[5] Z. Zoghi and G. Serpen, "Building an *Intrusion Detection System* on UNSW-NB15: Reducing the margin of error to deal with data overlap and imbalance," *Concurr. Comput.*, vol. 36, no. 25, Nov. 2024, doi: 10.1002/cpe.8242.

[6] Z. Chen, Z. W. Li, J. Huang, S. Z. Liu, and H. X. Long, "An effective method for anomaly detection in industrial Internet of Things using XGBoost and LSTM," *Sci. Rep.*, vol. 14, no. 1, Dec. 2024, doi: 10.1038/s41598-024-74822-6.

[7] P. Ferreira, E. Martins, J. Silva, and P. Teixeira, "Feature Selection and XGBoost for Enhanced Intrusion Detection: A Comparative Study Across Benchmark Datasets," in *ISDFS 2025 - 13th International Symposium on Digital Forensics and Security*, Institute of Electrical and Electronics Engineers Inc., 2025. doi: 10.1109/ISDFS65363.2025.11012060.

[8] V. Nwachukwu and A. John-Otumu, "Improved *Machine Learning*-Based Model for Network Traffic Anomaly Detection," in *2024 IEEE SmartBlock4Africa: Emerging and Resilient Technologies in Building and Securing African Nations*, Institute of Electrical and Electronics Engineers Inc., 2024. doi: 10.1109/SmartBlock4Africa61928.2024.10779548.

[9] K. K. Pal, A. V. Eriksen, and N. Dinh, "XGBoost Feature Selection for Multi-class and Binary Classification on UNSW-NB15 Dataset," in *Digest of Technical Papers - IEEE International Conference on Consumer Electronics*, Institute of Electrical and Electronics Engineers Inc., 2025. doi: 10.1109/ICCE63647.2025.10930023.

[10] T. Agustina, M. Masrizal, and I. Irmayanti, "Performance Analysis of Random Forest Algorithm for Network Anomaly Detection using Feature Selection," *Sinkron : Jurnal dan Penelitian Teknik Informatika*, vol. 8, no. 2, Apr. 2024, doi: 10.33395/sinkron.v8i2.13625.

[11] Z. P. Putra, "Evaluating the Performance of Classification Algorithms on the UNSW-NB15 Dataset for Network Intrusion Detection," *Jurnal Ilmiah FIFO*, vol. 16, no. 1, p. 84, Jun. 2024, doi: 10.22441/fifo.2024.v16i1.009.

[12] B. Nugraha Kasmara, E. Tri Esti Handayani, N. Dian Nathasia, and U. Nasional, "PERBANDINGAN KINERJA ALGORITMA K-NEAREST NEIGHBORS (K-NN) DAN DECISION TREE DALAM DETEKSI PAKET MALIS PADA JARINGAN," *STRING (Satuan Tulisan Riset dan Inovasi Teknologi)*, pp. 320–329, Apr. 2024.

[13] M. A. Rehman, S. I. A. Shah, A. Anwar, and N. Islam, "From Flows to Words: Can Zero-/Few-Shot LLMs Detect Network Intrusions? A Grammar-Constrained, Calibrated Evaluation on UNSW-NB15," *arXiv:2510*, Oct. 2025, [Online]. Available: http://arxiv.org/abs/2510.17883

[14] D. Chen, Q. Song, Y. Zhang, Q. Yu, L. Li, and Z. Yang, "Evaluating the Effectiveness of Various Model Combinations for Network Intrusion Detection on UNSW-NB15," in *2024 6th International Academic Exchange Conference on Science and Technology Innovation*, Institute of Electrical and Electronics Engineers Inc., 2024, pp. 312–317. doi: 10.1109/IAECST64597.2024.11117865.

[15] F. O. Albasheer Mohamed and M. Agarwal, "Using Recursive Feature Elimination Feature Selection based *Machine Learning* Classifier for Attack Classification on UNSW-NB 15 dataset," in *2024 IEEE 9th International Conference for Convergence in Technology*, Institute of Electrical and Electronics Engineers Inc., 2024. doi: 10.1109/I2CT61223.2024.10544076.

[16] A. Kumar, K. Guleria, R. Chauhan, and D. Upadhyay, "Advancing Intrusion Detection with *Machine Learning*: Insights from the UNSW-NB15 Dataset," in *2024 IEEE International Conference on Information Technology, Electronics and Intelligent Communication Systems, ICITEICS 2024*, Institute of Electrical and Electronics Engineers Inc., 2024. doi: 10.1109/ICITEICS61368.2024.10625148.

[17] K. S. Arlandy, A. Faqih, and A. R. Rinaldi, "Mengoptimalkan Kinerja Naïve Bayes Pada Ancaman Modern Dengan Menggunakan PCA Pada Data *Intrusion Detection System* (IDS)," *Jurnal Ilmu Komputer dan Informatika*, pp. 25–37, 2025.

[18] I. H. Putro, "PERFORMANCE COMPARISON OF SVM KERNELS FOR *INTRUSION DETECTION SYSTEM* USING UNSW-NB15 DATASET," *Jurnal Teknik Elektro*, vol. 17, no. 2, 2024.

[19] D. Shafir, A. Wahyu, P. Putra, and I. Made Suartana, "Perbandingan Kinerja Model Deteksi Serangan Pada *Intrusion Detection System* Dengan Tuning Hyperparameter," *Journal of Informatics and Computer Science*, vol. 06, pp. 984–993, 2025.

[20] M. Uppal, D. Gupta, S. Juneja, S. Mapari, C. N. Vanitha, and S. Saini, "Intrusion Detection using Feedforward Neural Network for Enhancing Network Security," in *International Conference on Computing and Intelligent Reality Technologies*, Institute of Electrical and Electronics Engineers Inc., 2024, pp. 361–366. doi: 10.1109/ICCIRT59484.2024.10921882.

[21] M. M. Abualhaj, H. Al-Mimi, A. Al-Allawee, Q. Y. Shambour, and M. Anbar, "Enhancing Intrusion Detection Using Dragonfly Algorithm-Based Feature Selection and Extra Trees for Classification," in *2025 5th International Conference on Emerging Smart Technologies and Applications, eSmarTA 2025*, Institute of Electrical and Electronics Engineers Inc., 2025. doi: 10.1109/eSmarTA66764.2025.11132251.

[22] Chalana B Arun, Anusha M, Ashwini Kodipalli, Trupthi Rao, Rohini B R, and Gargi N, "Enhancing Network Intrusion Detection using Artificial Neural Networks: An Analysis of the UNSW-NB15 Dataset," *2024 IEEE Canadian Conference on Electrical and Computer Engineering*, 2024.

[23] A. M. Jose *et al.*, "Multi-Class SVM & Random Forest Based Intrusion Detection Using UNSW-NB15 Dataset," in *2024 15th International Conference on Computing Communication and Networking Technologies*, Institute of Electrical and Electronics Engineers Inc., 2024. doi: 10.1109/ICCCNT61001.2024.10725989.

[24] R. F. Ramadhan and W. M. Ashari, "Performance Comparison of Random Forest and Decision Tree Algorithms for Anomaly Detection in Networks," 2024. [Online]. Available: http://jurnal.polibatam.ac.id/index.php/JAIC

[25] S. Patil and R. Bansode, "A Hybrid Feature Selection Approach Incorporating Mutual Information and Genetics Algorithm for Web Server Attack Detection," *Indian J. Sci. Technol.*, pp. 325–332, 2024, doi: 10.17485/IJST/v17i4.2820.

[26] G. Suchetha and K. Pushpalatha, "Optimizing Botnet Detection in IoT Networks: Feature Selection Analysis on the UNSW-NB15 Dataset," in *2024 IEEE International Conference on Distributed Computing*, Institute of Electrical and Electronics Engineers Inc., 2024, pp. 120–125. doi: 10.1109/DISCOVER62353.2024.10750583.

[27] M. A. N. Anargya, W. Ghozi, and F. A. Rafrastara, "Optimizing IoV Attack Detection using Random Under Sampling Techniques," *Jurnal Informatika: Jurnal Pengembangan IT*, vol. 10, no. 1, pp. 11–19, Jan. 2025, doi: 10.30591/jpit.v10i1.8034.