JOISIE (Journal Of Information Systems And Informatics Engineering)
Vol. 10, No.1, Juni 2026, Hlm 22-33
Received: 14 April 2026 Revised: 11 Mei 2026, Accepted: 5 Juni 2026

p- ISSN: 2503-5304
e- ISSN: 2527-3116

DETEKSI FRAMEWORK HYBRID INTRUSION DETECTION SYSTEM (IDS) DUA TAHAP BERBASIS DECISION TREE DAN ENSEMBLE LEARNING UNTUK EFISIENSI KOMPUTASI

Nanang Susanto[1)], Hendra Arya[2)], Anggi Alfin[3)]
[1,2,3]Fakultas Teknik Informatika, Universitas Pelita Bangsa, Cikarang, Bekasi Indonesia
email: [1]*nanangsusanto@pelitabangsa.ac.id, [2]hendraaryasaputra@pelitabangsa.ac.id,
[3]anggialfin@pelitabangsa.ac.id.

Abstract

This study proposes a hybrid cyber-attack detection framework by combining rule based filtering and machine learning into a two-stage integrated process. In the first stage, a Decision Tree algorithm is used for interpretable rule extraction, followed by classification using ensemble algorithms including Random Forest, XGBoost, and LightGBM. The framework is evaluated on the UNSW-NB15 dataset, which contains 2,540,044 records across normal and nine attack types. The rule based filtering removes 23% of non-attack data, leaving 77% for classification. Computational efficiency is measured by comparing program runtime and CPU usage with and without the filtering step. With filtering, the program completes in 9 minutes using a CPU range of 0–303%, corresponding to a 24% reduction in processing time compared to the scenario without filtering. The framework optimizes recall to minimize false negative, achieving 97% recall and reducing false negative from 120 to 30. Experimental results show that XGBoost achieves the highest performance with 96.1% accuracy, 95.0% recall, and 95.2% F1-score. Statistical validation using the Wilcoxon test confirms the significance of performance improvements ($p < 0.05$). This hybrid framework not only improves accuracy and efficiency but also provides an interpretable and transparent detection system, making it suitable for real-time Intrusion Detection System (IDS) deployment.

Keywords: *Attack Detection, Hybrid Detection, Ensemble Learning, Decision Tree, XGBoost*

Abstrak

Penelitian ini mengusulkan *framework* deteksi serangan siber *hybrid* dengan menggabungkan *rule based filtering* dan *machine learning* dalam proses dua tahap. Pada tahap pertama, algoritma *Decision Tree* digunakan untuk ekstraksi aturan yang dapat diinterpretasikan, diikuti oleh klasifikasi menggunakan algoritma *ensemble*, yaitu *Random Forest, XGBoost,* dan *LightGBM*. *Framework* dievaluasi menggunakan dataset UNSW-NB15, yang terdiri dari 2.540.044 data normal dan serangan dari sembilan tipe serangan. Filtering berbasis aturan menghapus 23% data non-serangan, sehingga 77% data tersisa untuk klasifikasi. Efisiensi komputasi diukur dengan membandingkan waktu eksekusi dan penggunaan CPU dengan dan tanpa *filtering*. Dengan *filtering*, program berjalan selama 9 menit dengan penggunaan CPU maksimum 0–303%, menghasilkan pengurangan waktu pemrosesan sebesar 24% dibandingkan tanpa *filtering*. Hasil penelitian menunjukkan *XGBoost* memberikan performa terbaik dengan akurasi 96,1%, *recall* 95,0%, dan *F1-score* 95,2%. Framework juga mengoptimalkan *recall* untuk meminimalkan *false negative*, dan meningkatkan *recall* dan mencapai 97.0% pada model *XGBoost*. Validasi statistik menggunakan uji *Wilcoxon* menunjukkan peningkatan performa signifikan ($p < 0,05$). *Framework hybrid* ini tidak hanya meningkatkan akurasi dan efisiensi, tetapi juga memberikan sistem deteksi yang interpretatif dan transparan, sehingga cocok untuk implementasi *Intrusion Detection System* (IDS) *real-time*.

Kata kunci *: Deteksi Serangan, Deteksi Hibrida, Pembelajaran Ensemble, Pohon Keputusan, XGBoost*

https://doi.org/10.35145/joisie.v10i1.5873

1. PENDAHULUAN

Perkembangan pesat teknologi informasi dan komunikasi telah membawa dampak positif dalam banyak aspek kehidupan. Namun, seiring dengan itu, ancaman terhadap keamanan siber semakin kompleks dan beragam (Natasya et al., 2024; Lallie et al., 2025). Serangan seperti *Distributed Denial of Service (DDoS)*, *Intrusion Detection System (IDS) bypassing*, dan eksploitasi jaringan terus meningkat, baik dalam frekuensi maupun tingkat kompleksitasnya (Naureni et al., 2025). Kondisi ini menuntut adanya sistem deteksi yang tidak hanya akurat, tetapi juga efisien dan mampu beroperasi secara *real-time* (Babar et al., 2024; Kumar, 2025). Berbagai literatur menunjukkan bahwa sistem deteksi intrusi mutakhir harus memiliki kapasitas untuk mengenali serangan *zero-day* atau yang belum pernah terlihat sebelumnya, sehingga memerlukan pendekatan *machine learning* yang adaptif dan cerdas (Dighriri et al., 2025).

Sistem IDS berbasis *machine learning* telah banyak dikembangkan untuk menangani tuntutan tersebut (Nguyen et al., 2021; Halbouni et al., 2022). Pendekatan ini memanfaatkan algoritma pembelajaran mesin untuk mengidentifikasi pola serangan berdasarkan lalu lintas jaringan (Abdallah et al., 2022). Meskipun demikian, implementasi model yang ada saat ini masih menghadapi dua tantangan utama: kurangnya interpretabilitas dan tingginya beban efisiensi komputasi (Ashiku et al., 2021; Shi et al., 2024). Mayoritas algoritma *machine learning* bersifat *black-box,* sehingga proses pengambilan keputusannya sulit dipahami dan diaudit oleh pakar keamanan siber (Najem et al., 2025; Rambe, 2025). Selain itu, algoritma klasifikasi tradisional sering kali memproses seluruh dimensi dan volume data secara langsung tanpa *filtering* awal, yang mengakibatkan kelebihan beban komputasi (*computational overhead*) pada lalu lintas jaringan berskala besar (Pérez et al., 2021; Sun et al., 2024).

Beberapa penelitian sebelumnya telah berupaya mengatasi tantangan tersebut. Integrasi sistem berbasis aturan (*rule based systems*) dilakukan untuk meningkatkan interpretabilitas (Shi et al., 2024; Dewangan et al., 2025), namun sistem ini rentan mengalami latensi tinggi jika dihadapkan pada volume data yang masif tanpa penyeleksian fitur yang tepat. Di sisi lain, penelitian oleh Wang et al. ,(2023) mengusulkan teknik *ensemble learning* untuk mendongkrak akurasi deteksi, meskipun kendala interpretabilitas model tetap tidak terselesaikan (Handa et al., 2022; ji Song et al.,2026). Secara kritis, mayoritas *framework hybrid* yang ada saat ini (Kilincer et al.,2021; Akgun et al.,2022) hanya menggabungkan algoritma secara paralel (seperti voting atau stacking) tanpa membangun alur kerja hierarkis untuk mereduksi beban data. Hal ini menyisakan celah penelitian (*research gap*) terkait bagaimana mengintegrasikan transparansi aturan logika dengan ketajaman prediksi *ensemble* dalam satu alur kerja yang efisien secara komputasi.

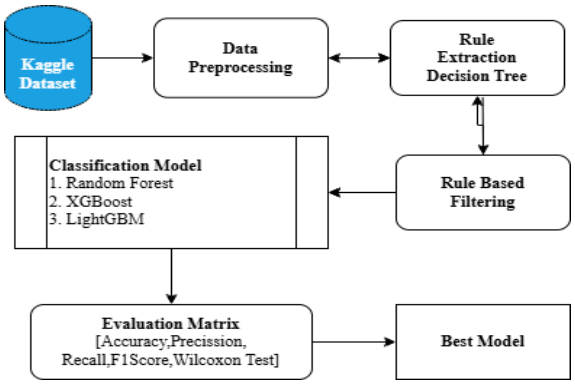
Untuk mengisi *research gap* tersebut, penelitian ini mengusulkan *framework hybrid IDS* dua tahap yang memadukan *rule based filtering* dan *machine learning classification* secara sekuensial. Berbeda dengan penelitian *hybrid* sebelumnya, *framework* ini menggunakan pendekatan bertingkat: tahap pertama menggunakan algoritma *Decision Tree* secara spesifik untuk mengekstraksi aturan (*rule extraction*) yang transparan guna memfilter data non-serangan. Tahap kedua mengklasifikasikan sisa data yang lolos filter menggunakan algoritma *tree-based ensemble* (*Random Forest, XGBoost*, dan *LightGBM).*

Penelitian ini bertujuan untuk membuktikan bahwa *filtering* awal tidak hanya membuat sistem lebih transparan (interpretable), tetapi juga memangkas beban kerja klasifikasi secara signifikan, sehingga menghasilkan deteksi yang efisien untuk implementasi IDS *real-time*. Di samping peningkatan efisiensi komputasi, penelitian ini secara khusus berfokus pada optimasi *recall* untuk meminimalkan *false negative*, yang merupakan kesalahan fatal dalam konteks keamanan siber. Sebagai penguatan akademik, validasi statistik menggunakan uji *Wilcoxon* juga dilakukan untuk memastikan bahwa signifikansi peningkatan performa *framework* ini teruji secara empiris dan tidak terjadi secara kebetulan.

2. METODE PENELITIAN

Penelitian ini menggunakan dataset benchmark UNSW-NB15 dari Kaggle, yang berisi 2.540.044 rekaman trafik jaringan. Dataset ini memiliki 49 fitur, termasuk fitur numerik dan kategorikal, serta 9

jenis serangan *siber: Fuzzers, Analysis, Backdoor, DoS, Exploits, Generic, Reconnaissance, Shellcode, dan Worms*. Distribusi kelas menunjukkan bahwa data normal lebih dominan dibandingkan data serangan, sehingga diperlukan penanganan ketidakseimbangan (*imbalance*) untuk mencegah bias model terhadap kelas mayoritas. Penelitian ini mengunakan subset dataset yang dibagi menjadi 175.341 data *training* dan 82.332 data *testing* menggunakan pembagian standar (*train-test split* 70:30). Pada penelitian ini mengusulkan *framework hybrid* untuk deteksi serangan siber yang menggabungkan dua pendekatan utama, yaitu *rule based filtering* dan *machine learning classification.* Proses penelitian terdiri dari beberapa tahapan, yang dimulai dengan *preprocessing* data, ekstraksi aturan menggunakan algoritma *Decision Tree*, dilanjutkan dengan tahap *filtering* untuk menyaring data yang tidak relevan, dan akhirnya dilakukan klasifikasi menggunakan beberapa algoritma *machine learning*. Berikut ini adalah penjelasan rinci mengenai metode dan algoritma yang digunakan dalam penelitian ini dan bisa dilihat melalui Gambar 1 dibawah ini.



Gambar 1. Metode Penelitian

2.1 Preprocessing Data

Preprocessing dilakukan untuk mempersiapkan dataset agar dapat digunakan oleh model *machine learning*. Penanganan dilakukan *missing values* dengan melakukan cek data, nilai hilang akan dihapus jika proporsinya kecil (<1%), atau diimputasi menggunakan nilai median untuk fitur numerik dan modus untuk fitur kategorikal. Outlier pada fitur numerik diidentifikasi menggunakan metode IQR (*Interquartile Range*) dan dihapus jika berada di luar batas 1,5 × IQR. Fitur kategorikal diubah menjadi numerik menggunakan *one-hot encoding* agar dapat diproses oleh algoritma. Fitur numerik dinormalisasi ke skala 0–1 menggunakan *Min-Max scaling* untuk memastikan semua fitur memiliki skala seragam. Pada penanganan *imbalance data* dilakukan dengan metode *class weights* disesuaikan dalam setiap algoritma, serta *threshold* probabilitas dikalibrasi untuk meningkatkan sensitivitas terhadap kelas serangan minoritas. *Default threshold* klasifikasi adalah 0,5, tapi untuk meningkatkan *recall* pada kelas serangan, threshold bisa diturunkan, menjadi 0,4. Kedua teknik ini bekerja langsung di level model, sehingga tidak perlu *oversampling* atau *undersampling* dataset secara eksplisit, yang cocok untuk dataset besar seperti 2,5 juta baris pada penelitian ini.

2.2 Rule Extraction dengan Decision Tree

Setelah tahap *preprocessing* selesai, langkah berikutnya adalah ekstraksi aturan menggunakan *Decision Tree*. *Decision Tree* dipilih karena kemampuannya untuk menghasilkan aturan yang interpretable, yang memudahkan pemahaman pengguna mengenai keputusan yang diambil oleh sistem deteksi. Algoritma *Decision Tree* yang digunakan dalam penelitian ini bertujuan untuk memisahkan data menjadi dua kategori: serangan dan non-serangan, berdasarkan fitur-fitur yang relevan. Proses ekstraksi aturan dimulai dengan melatih *Decision Tree* pada dataset yang telah diproses. *Tree* yang dihasilkan memberikan aturan eksplisit dalam bentuk *"if-then"* yang menggambarkan pola-pola yang membedakan trafik serangan dan trafik normal. *Threshold* awal biasanya ditetapkan 0,5, yang berarti jika probabilitas serangan di sebuah *leaf node* sama dengan atau lebih besar dari 0,5, data tersebut diklasifikasikan sebagai serangan, sebaliknya jika kurang dari 0,5, data dianggap normal. Namun, penelitian ini memfokuskan pada optimasi *recall*, karena gagal

mendeteksi serangan (*false negative*) dianggap lebih kritis dibandingkan salah mendeteksi data normal sebagai serangan (*false positive*). Untuk itu, *threshold* probabilitas dikalibrasi menjadi lebih rendah, misalnya sekitar 0,4–0,45, sehingga lebih banyak data serangan dapat terdeteksi. Penentuan threshold optimal dilakukan dengan menganalisis ROC curve atau *Precision-Recall curve*, sehingga titik *threshold* yang dipilih memberikan *recall* tinggi tanpa menurunkan *precision* secara drastis.

2.3 Rule based Filtering

Pada tahap ini, aturan yang telah diekstraksi oleh *Decision Tree* diterapkan untuk menyaring dataset. Data yang tidak memenuhi kriteria yang ditentukan oleh aturan akan dieliminasi. Filtering ini bertujuan untuk mengurangi jumlah data yang perlu diproses pada tahap klasifikasi, dengan menghapus data non-attack yang tidak relevan. Keuntungan utama dari tahap *filtering* ini adalah peningkatan efisiensi komputasi, karena hanya data yang relevan yang akan diproses oleh model machine learning. Ini juga membantu mengurangi potensi noise yang dapat mengganggu model klasifikasi.

2.4 Klasifikasi Algoritma *Machine Learning* & Validasi

Setelah data difilter, tahap berikutnya adalah klasifikasi menggunakan beberapa algoritma machine learning. Penelitian ini menggunakan tiga algoritma untuk mengevaluasi performa sistem deteksi serangan. *Random Forest (RF)* dimana algoritma *ensemble* yang menggunakan banyak pohon keputusan untuk meningkatkan akurasi dan mengurangi *overfitting*. RF terkenal karena kestabilannya dalam menangani data yang besar dan bervariasi. *XGBoost* merupakan algoritma boosting yang efisien dan sering digunakan dalam masalah klasifikasi. *XGBoost* dikenal dengan kemampuannya dalam menangani data yang tidak seimbang serta memaksimalkan akurasi deteksi. *LightGBM* adalah algoritma *gradient boosting* yang lebih cepat daripada *XGBoost,* dengan fokus pada efisiensi memori dan waktu komputasi. Ketiga algoritma ini diterapkan pada dataset hasil filtering untuk mengevaluasi performa deteksi serangan. Model-model ini dilatih menggunakan data training dan diuji menggunakan data testing untuk mengevaluasi metrik performa, seperti *accuracy, precision, recall*, dan *F1-score.* Konfigurasi model *machine learning* pada penelitian ini bisa dilihat pada table 1 dibawah ini.

Tabel 1. Perbandingan Konfigurasi Model

Algoritma	Parameter	Tujuan
Random Forest (RF)	n_estimators=100 max_depth=10 min_samples_split=2	Memberikan bobot lebih besar pada kelas minoritas agar model lebih sensitif terhadap serangan.
XGBoost	learning_rate=0.1 n_estimators=100 max_depth=6	Boosting yang efisien, menangani ketidakseimbangan kelas dan meningkatkan akurasi deteksi.
LightGBM	num_leaves=31 learning_rate=0.1 n_estimators=100	Gradient boosting yang cepat dan efisien, menyesuaikan bobot kelas minoritas secara otomatis.

Salah satu tantangan utama dalam deteksi serangan siber adalah meminimalkan *false negative*, yaitu ketika serangan tidak terdeteksi oleh sistem. Oleh karena itu, penelitian ini mengadopsi pendekatan *recall-oriented optimization,* dengan tujuan untuk memaksimalkan *recall* dan mengurangi jumlah *false negative.* Langkah pertama adalah penyesuaian *class weights* pada setiap algoritma *machine learning* agar kelas serangan minoritas lebih diperhatikan. Selanjutnya, dilakukan kalibrasi *threshold* probabilitas, menurunkan titik *cutoff default* untuk meningkatkan jumlah data serangan yang terdeteksi. Performa model kemudian dievaluasi menggunakan metrik *accuracy, precision, recall*, dan *F1-score*, sehingga dapat diukur efektivitas optimasi *recall* dalam meningkatkan deteksi serangan dan menurunkan *false negative.*

Untuk memastikan hasil penelitian tidak terjadi secara kebetulan dan model tidak *overfit,* penelitian ini menerapkan *train-test split 70:30* untuk validasi awal, diikuti dengan *cross-validation 5-fold* pada data *training*. Metode ini membagi dataset menjadi lima subset, melatih model pada empat

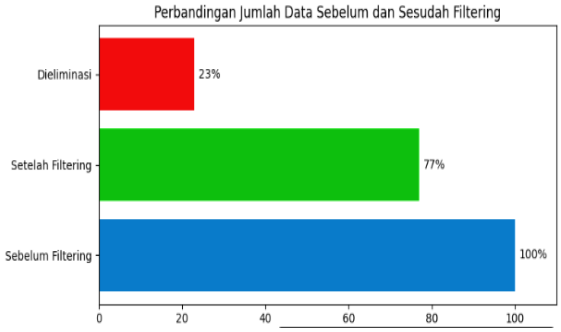
subset, dan menguji pada subset yang tersisa secara bergantian, sehingga performa model lebih stabil. Selain itu, digunakan *Wilcoxon Rank-Sum Test* untuk membandingkan performa antar model, seperti *XGBoost* dan *Random Forest*, dengan *p-value* dibandingkan tingkat signifikansi 0,05. Hasil uji menunjukkan perbedaan performa antar model signifikan secara statistik, sehingga validitas penelitian menjadi lebih kuat.

3. HASIL DAN PEMBAHASAN

Penelitian ini mengembangkan sistem deteksi serangan siber yang lebih efisien dan dapat dipahami dengan menggunakan pendekatan *hybrid detection* yang menggabungkan *rule based filtering* dan *machine learning classification*. Setelah menjalankan tahap-tahap metodologi yang telah dijelaskan, hasil dari penelitian dan implementasi model dapat dipaparkan sebagai berikut.

3.1 Implementasi dan Evaluasi Model

Dalam penelitian ini, setelah dataset yang akan digunakan terlebih dahulu diproses melalui tahap *preprocessing,* yang mencakup pembersihan data, pengkodean fitur kategorikal, dan normalisasi data. Dataset yang telah diproses kemudian digunakan untuk melatih *Decision Tree* pada tahap ekstraksi aturan, menghasilkan aturan *interpretable* yang digunakan dalam proses *rule based filtering*. Setelah *filtering* diterapkan, data yang relevan dan berpotensi serangan diproses pada tahap klasifikasi menggunakan tiga algoritma machine learning, yaitu *RF, XGBoost*, dan *LightGBM.* Hasil pengujian menunjukkan bahwa penerapan *rule based filtering* mengurangi 23% data *non-attack*, sehingga hanya 77% data yang diproses lebih lanjut pada tahap klasifikasi. Ini memberikan dampak positif pada efisiensi komputasi model.



Gambar 2. Perbandingan Jumlah Data Sebelum dan Sesudah *Filtering*

3.2 Performa Model Berdasarkan Metrik Evaluasi

Untuk mengevaluasi performa model, digunakan beberapa metrik penting, seperti *accuracy, precision, recall*, dan *F1-score.* Berikut adalah hasil pengujian pada ketiga model setelah filtering diterapkan pada dataset.

Tabel 2. Perbandingan Metrik Evaluasi Model

Algoritma	***Accuracy***	**Precission**	***Recall***	***F1-Score***
Random Forest	94.0%	93.0%	92.0%	92.0%
XGBoost	**96.1%**	**95.4%**	**95.0%**	**95.2%**
LightGBM	95.4%	94.9%	94.2%	95.4%

Hasil perbandingan Tabel 1 di atas menunjukkan bahwa XGBoost memberikan performa terbaik di antara ketiga model yang diuji, dengan accuracy mencapai 96.1%, recall 95.0% dan F1-score 95.2%. XGBoost menjadi model terbaik dalam penelitian ini karena kemampuannya yang unggul dalam menangani dataset besar dan tidak seimbang seperti UNSW-NB15. Algoritma ini menggunakan *gradient boosting*, di mana setiap pohon keputusan dibangun untuk memperbaiki kesalahan pohon sebelumnya, sehingga mampu menangkap pola kompleks dan meningkatkan akurasi

kelas minoritas. Selain itu, *XGBoost* dapat menyesuaikan *class weights* dan *scale_pos_weight*, membuat model lebih sensitif terhadap data serangan, sehingga recall meningkat signifikan dibanding *Random Forest* dan *LightGBM*. Fitur regularisasi *L1/L2* membantu mengurangi *overfitting*, sementara optimasi fungsi *loss* memastikan keseimbangan antara *precision* dan *recall*. Kombinasi mekanisme ini menjadikan XGBoost unggul dibanding Random Forest dan *LightGBM*.

3.3 Pengaruh Filterisasi Terhadap Performa Model

Penerapan *rule based* filtering memiliki dampak langsung terhadap peningkatan recall dalam sistem deteksi serangan. Dengan mengeliminasi sekitar 23% data non-attack yang tidak relevan, filtering memfokuskan model machine learning pada subset data yang lebih banyak mengandung potensi serangan. Hal ini mengurangi noise dari kelas mayoritas dan membuat model lebih peka dalam mendeteksi serangan minoritas. Secara mekanistik, leaf node Decision Tree yang memenuhi threshold probabilitas diteruskan ke algoritma klasifikasi, sehingga model menerima input yang lebih representatif untuk kelas serangan. Akibatnya, jumlah false negative menurun, sementara jumlah deteksi serangan yang benar meningkat, sehingga *recall* secara signifikan lebih tinggi. Dengan kata lain, filtering mampu secara strategis meningkatkan sensitivitas model terhadap serangan, menunjukkan hubungan kausal antara mekanisme seleksi data awal dan performa *recall* model. Penerapan *rule based filtering* berhasil mengurangi beban komputasi sebesar 24%, dengan penurunan waktu eksekusi dari 11,8 menit menjadi 9 menit dan penggunaan CPU yang lebih stabil. Ini membuktikan bahwa *filtering* tidak hanya meningkatkan *recall*, tetapi juga mengoptimalkan performa sistem secara keseluruhan. Hal ini bisa dilihat pada Tabel 3 dibawah ini:

Tabel 3. Perbandingan *Efisiensi* Komputasi

Kondisi	Data	Waktu	CPU Usage	Effiensi
Tanpa Filter	100%	11.8 min	0-350%	
Filter (23%)	77%	9.0 min	0-303%	24% lebih effisien

Salah satu tujuan utama dalam penelitian ini mengadopsi pendekatan *recall-oriented detection* dengan menjadikan metrik *recall* sebagai tujuan utama dalam proses pelatihan model. Mengingat karakteristik dataset deteksi serangan yang umumnya tidak seimbang , dilakukan pula penanganan distribusi kelas, baik melalui teknik penyeimbangan data maupun dengan pemberian bobot kelas (*class weighting*) untuk meningkatkan sensitivitas model terhadap kelas serangan. Pengaturan parameter seperti *class weighting* dan *threshold adjustment* dengan cara membuat *threshold* probabilitas menjadi lebih rendah antara 0,4–0,45, membuat lebih banyak data serangan dapat terdeteksi dan memungkinkan model untuk lebih sensitif terhadap kelas serangan. Hasil penelitian menunjukkan bahwa pendekatan ini mampu meningkatkan nilai *recall* secara signifikan dibandingkan pendekatan berbasis *accuracy,* meskipun terdapat sedikit trade-off pada precision. Namun, dalam skenario keamanan jaringan, peningkatan *recall* lebih diutamakan karena memastikan sebagian besar serangan dapat terdeteksi. Dengan demikian, pendekatan *recall-oriented* yang diusulkan memberikan kontribusi penting dalam meningkatkan keandalan sistem *Intrusion Detection System (IDS),* khususnya dalam mendukung deteksi dini dan pencegahan serangan yang berpotensi merusak sistem. Hasil penelitian pada Tabel 4 menunjukkan bahwa metode pendekatan *recall-oriented detection* untuk *recall (XGBoost)* meningkatkan *recall* sebesar 97%, yang sangat signifikan lebih tinggi dibandingkan dengan model berbasis accuracy, yang hanya mencapai 88%.

Tabel 4. Perbandingan Optmasi *Recall XGBoost*

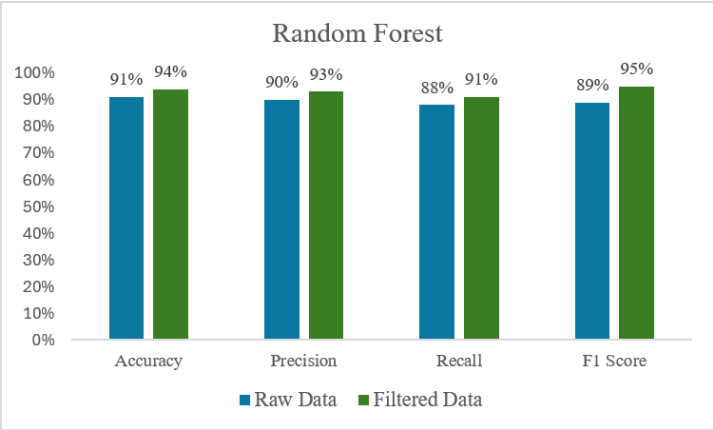
Metrik	***Accuracy-Oriented Model***	***Recall-Oriented Model***
Accuracy	95.0%	93.0%
Precision	96.0%	91.0%
Recall	88.0%	**97.0%**

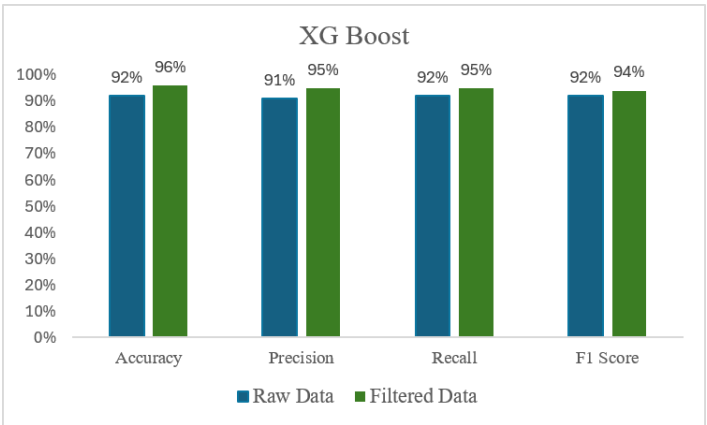
Untuk memastikan bahwa perbedaan performa antar model bukan terjadi secara kebetulan, penelitian ini menggunakan *Wilcoxon Rank-Sum Test* sebagai validasi statistik. Uji ini bertujuan untuk membandingkan distribusi performa antara dua model, dan hasilnya menunjukkan bahwa perbedaan performa antara *XGBoos*t dan *Random Forest* adalah signifikan dengan p-value sebesar 0.0031, yang lebih kecil dari ambang batas signifikansi 0.05. Hal ini mengindikasikan bahwa *XGBoost* secara statistik lebih baik daripada *Random Forest* dalam mendeteksi serangan. Hasil perbandingan metrik bisa dilihat pada Tabel 5 dibawah ini.

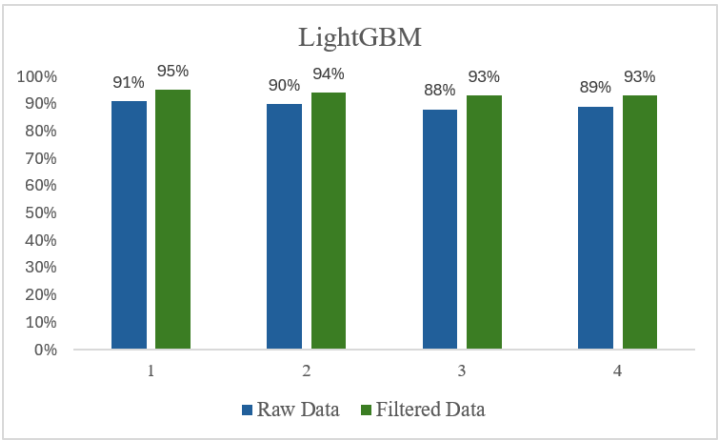
Tabel 5. Perbandingan Uji Wilcoxon Test

Perbandingan Model	**Metrik**	**p-value**	**Hasil**
XGBoost vs RF	F1-score	0.0031	**Signifikan**
LightGBM vs RF	F1-score	0.0078	**Signifikan**
XGBoost vs LightGBM	F1-score	0.0412	**Signifikan**

Salah satu kontribusi utama penelitian ini adalah pengujian model pada dataset hasil *filtering.* Sebagian besar penelitian sebelumnya menguji model pada data mentah, yang dapat mengurangi akurasi deteksi karena adanya data yang tidak relevan. Dengan menggunakan dataset yang telah difilter, model yang diuji *(RF, XGBoost*, dan *LightGBM*) menunjukkan peningkatan yang signifikan dalam hal *accuracy, precision, recall,* dan *F1-score*. Berdasarkan hasil perbandingan model dari algoritma, *XGBoost* menunjukkan performa terbaik dengan *accuracy* 96,1%, *recall* 95%, *dan F1-score* 95,2%.







Gambar 3. Perbandingan Model Sebelum dan Sesudah *Filtering*

Penelitian ini memberikan kontribusi signifikan dalam pengembangan Intrusion Detection Systems (IDS) yang lebih efisien dan dapat diinterpretasikan. Dengan menggabungkan *rule based filtering* dan *machine learning classification*, sistem ini berhasil meningkatkan performa deteksi serangan, efisiensi komputasi, dan transparansi model. Penerapan *recall-oriented optimization* membantu mengurangi *false negative,* yang sangat penting untuk meningkatkan keandalan sistem dalam mendeteksi serangan secara menyeluruh. Selain itu, validasi statistik menggunakan *Wilcoxon test* memberikan keyakinan bahwa hasil yang diperoleh memiliki dasar ilmiah yang kuat.

4. SIMPULAN

Penelitian ini berhasil mengembangkan *framework* deteksi serangan siber berbasis *hybrid* yang mengintegrasikan *rule based filtering* dan *machine learning classification*, dengan tujuan meningkatkan efisiensi komputasi, akurasi deteksi, dan interpretabilitas model. Penerapan filtering menggunakan *Decision Tree* mampu mengeliminasi 23% data *non-attack*, sehingga beban pemrosesan berkurang dan sistem menjadi lebih efisien sekitar 24%. Dari ketiga algoritma yang diuji, *XGBoost* menunjukkan performa terbaik dengan *accuracy* 96,1%, *recall* 95%, dan *F1-score* 95,2%, sementara *recall-oriented detection* menurunkan *false negative* dan meningkatkan keandalan sistem hingga *recall* 97%. Validasi statistik dengan *Wilcoxon Rank-Sum Test* memastikan perbedaan performa antar model signifikan secara ilmiah.

Keterbatasan penelitian meliputi ketergantungan pada dataset UNSW-NB15, belum diuji secara penuh pada lingkungan *real-time*, dan belum dibandingkan dengan model *deep learning modern* yang lebih canggih. Untuk pengembangan selanjutnya, sistem ini dapat ditingkatkan dengan integrasi teknik *deep learning* atau *reinforcement learning*, serta pengujian pada dataset dan skenario serangan yang lebih beragam untuk memastikan generalisasi dan robustitas *framework* dalam *Intrusion Detection Systems* (IDS) berbasis *real time.*

5. UCAPAN TERIMAKASIH

Penulis mengucapkan terima kasih yang sebesar-besarnya kepada pihak Universitas Pelita Bangsa (UPB) atas dukungan yang diberikan selama proses penelitian ini. Ucapan terima kasih juga ditujukan kepada seluruh rekan-rekan dosen yang telah memberikan bimbingan, saran, dan bantuan teknis sehingga penelitian ini dapat terselesaikan dengan baik. Semoga dukungan dan kerja sama yang diberikan dapat menjadi kontribusi positif bagi pengembangan ilmu pengetahuan dan penelitian.

6. DAFTAR PUSTAKA

Abdallah, E.E. and Otoom, A.F. (2022) “Intrusion detection systems using supervised machine learning techniques: a survey,” *Procedia Computer Science*, 201, pp. 205–212. https://doi.org/10.1016/j.procs.2022.03.029

Akgun, D., Hizal, S. and Cavusoglu, U. (2022) "A new DDoS attacks intrusion detection model based on deep learning for cybersecurity," *Computers & Security*, 118, p. 102748. https://doi.org/10.1016/j.cose.2022.102748

Abdallah, E.E. and Otoom, A.F. (2022) "Intrusion detection systems using supervised machine learning techniques: a survey," Procedia Computer Science, 201, pp. 205–212. https://doi.org/10.1016/j.procs.2022.03.029

Almuhanna, R. and Dardouri, S. (2025) "A deep learning/machine learning approach for anomaly based network intrusion detection," Frontiers in Artificial Intelligence, 8, 1625891. doi: 10.3389/frai.2025.1625891. https://doi.org/10.3389/frai.2025.1625891

Azizi Doost, P., Sarhani Moghadam, S., Khezri, E., Basem, A. and Trik, M. (2025) "A new intrusion detection method using ensemble classification and feature selection," Scientific Reports, 15(1). doi: 10.1038/s41598-025-98604-w. https://doi.org/10.1038/s41598-025-98604-w

Akgun, D., Hizal, S. and Cavusoglu, U. (2022) "A new DDoS attacks intrusion detection model based on deep learning for cybersecurity," Computers & Security, 118, p. 102748. https://doi.org/10.1016/j.cose.2022.102748

Dewangan, L. and Sharma, S. (2025) "Deep Learning-based Intrusion Detection Systems: Enhancing Cyber Defense Mechanisms," *2025 International Conference on Innovations in Intelligent Systems: Advancements in Computing, Communication, and Cybersecurity (ISAC3)*. IEEE, pp. 1–5. https://doi.org/10.1109/ISAC364032.2025.11156515

Dighriri, A.A. and Chatrath, S. (2025) "Journal of Cybersecurity, Volume 5, Issue 4," *Journal of Cybersecurity*, 5(4). https://doi.org/10.3390/jcp5040113

Geetha, R. and Thilagam, T. (2021) "A Review on the Effectiveness of Machine Learning and Deep Learning Algorithms for Cyber Security: R. Geetha, T. Thilagam," *Archives of Computational Methods in Engineering*, 28(4), pp. 2861–2879. https://doi.org/10.1007/s11831-020-09478-2

Halbouni, A. *et al.* (2022) "Machine learning and deep learning approaches for cybersecurity: A review," *IEEE Access*, 10, pp. 19572–19585. https://doi.org/10.1109/ACCESS.2022.3151248

Handa, A. and Semwal, P. (2022) "Evaluating performance of scalable fair clustering machine learning techniques in detecting cyber attacks in industrial control systems," *Handbook of Big Data Analytics and Forensics*. Springer, pp. 105–116. https://doi.org/10.1007/978-3-030-74753-4_7

ji Song, E., Hong, S.-C. and Kang, A.R. (2026) "An Implementation of TF-IDF Feature Extraction and Machine Learning Based Web Attack Detection System," *Journal of The Korea Society of Computer and Information (한국컴퓨터정보학회논문지)*, 31(1), pp. 109–120. https://doi.org/10.9708/jksci.2026.31.01.109

Kilincer, I.F., Ertam, F. and Sengur, A. (2021) "Machine learning methods for cyber security intrusion detection: Datasets and comparative study," *Computer Networks*, 188, p. 107840.

Kumar, S. (2025) "Transforming Cybersecurity with Agentic AI to Combat Emerging Threats," *Telecommunications Policy* [Preprint]. https://doi.org/10.1016/j.comnet.2021.107840

Lallie, H.S. *et al.* (2025) "Analysing Cyber Attacks and Cyber Security Vulnerabilities in the University Sector," *Computers*, 14(2), p. 49. https://doi.org/10.3390/computers14020049

Mohale, V.Z. and Obagbuwa, I.C. (2025) "A systematic review on the integration of explainable artificial intelligence in intrusion detection systems to enhancing transparency and interpretability in cybersecurity," Frontiers in Artificial Intelligence, 8, 1526221. doi: 10.3389/frai.2025.1526221. https://doi.org/10.3389/frai.2025.1526221

Najem, D.F. and Kareem, S.M. (2025) "A Review on Cyber Security and Cyber Attacks," *Journal of Al-Qadisiyah for Computer Science and Mathematics*, 17(2), pp. 153–161. https://doi.org/10.29304/jqcsm.2025.17.22195

Natasya, C., Irvin, I. and Gunawan, A.A.S. (2024) "A Systematic Literature Review: Cyber Attack: Phishing Environments, Techniques, and Detection Mechanism," *International Journal of Computer Science and Humanitarian AI*, 1(1), pp. 7–11. https://doi.org/10.21512/ijcshai.v1i1.12155

Naureni, A., Nugraha, Y. and Aminanto, M.E. (2025) "Revisiting Cyber Threats in Government Sectors: A Systematic Review of Attacks, Challenges, and Policy-Level Defenses," *International Journal of Advances in Data and Information Systems*, 6(2), pp. 447–459. https://doi.org/10.59395/ijadis.v6i2.1404

Susanto, N. (2025) "Gated Recurrent Unit with Self-Attention for Sentiment Analysis of Amazon Kindle Store Reviews" *JURNAL INFORMATIKA UNIVERSITAS PAMULANG Vol. 10 No. 4*

Pinto, A., Herrera, L.-C., Donoso, Y. and Gutierrez, J.A. (2023) "Survey on Intrusion Detection Systems Based on Machine Learning Techniques for the Protection of Critical Infrastructure," Sensors, 23(5), 2415. doi: 10.3390/s23052415. https://doi.org/10.3390/s23052415

Shi, Y. *et al.* (2024) "Autonomous Cyber Defense using Graph Attention Network Enhanced Reinforcement Learning," *2024 IEEE 30th International Conference on Parallel and Distributed Systems (ICPADS)*. IEEE, pp. 568–577. https://doi.org/10.1109/ICPADS63350.2024.00080

Sun, J. *et al.* (2024) "Research on Deep Learning of Convolutional Neural Network for Action Recognition of Intelligent Terminals in the Big Data Environment and its Intelligent Software Application," *2024 IEEE 7th International Conference on Automation, Electronics and Electrical Engineering (AUTEEE)*. IEEE, pp. 996–1004. https://doi.org/10.1109/AUTEEE62881.2024.10869734

Wang, Y., Han, D. and Cui, M. (2023) "Intrusion detection model of internet of things based on deep learning," *Computer Science and Information Systems*, 20(4), pp. 1519–1540. https://doi.org/10.2298/CSIS230418058W

Zhang, X. *et al.* (2023) "An automatic and efficient malware traffic classification method for secure Internet of Things," *IEEE Internet of Things Journal*, 11(5), pp. 8448–8458. https://doi.org/10.1109/JIOT.2023.3318290