Jurnal Riset Rekayasa Elektro
Volume 7 No 2, Desember 2025, Hal. 193~202
DOI : 10.30595/jrre.v7i2.27866

P-ISSN: 2685 - 4341
E-ISSN: 2685 - 5313

Perbandingan Performa Algoritma *Random Forest* dan SVM Dalam Mendeteksi serangan DDoS di Jaringan *Cloud*

Comparison of the Performance of Random Forest and SVM Algorithms in Detecting DDoS Attacks in Cloud Networks

Muhammad Andika Fathurrahman[1], Dwi Wahyu Prabowo[2*]
[1,2] Program Studi S1 Sistem Informasi, Fakultas Ilmu Komputer, Universitas Darwan Ali
[1,2] Jl. Batu Berlian No. 10, Sampit, Kalimantan Tengah, Indonesia
email: [1]andika06092003@gmail.com, [2*]dwi.wahyu9@unda.ac.id

Informasi Artikel

Dikirim, 19 Agustus 2025
Diterima, 6 November 2025
Diterbitkan, 5 Desember 2025

Kata Kunci :
DDoS, *Machine Learning Random Forest*, Seleksi Fitur CFS

ABSTRAK

Distributed Denial of Service (DDoS) merupakan ancaman serius terhadap layanan jaringan, khususnya pada lingkungan *cloud* yang bersifat terbuka dan dinamis. Penelitian ini bertujuan untuk mendeteksi serangan DDoS menggunakan algoritma *machine learning*, yakni *Random Forest* (RF) dan *Support Vector Machine* (SVM), serta mengevaluasi pengaruh seleksi fitur menggunakan *Correlation-based Feature Selection* (CFS) dan *Rough Set* (RS). Eksperimen dilakukan menggunakan dataset dari Zenodo dengan validasi silang 10-Fold dan evaluasi berbasis metrik *accuracy*, *precision*, *recall*, *F1-score*, dan *kappa*. Hasil menunjukkan bahwa model *Random Forest* secara konsisten memberikan performa terbaik dibandingkan SVM. Skema terbaik yaitu *Random Forest* dengan seleksi fitur RS menghasilkan nilai *accuracy* 99.99%, *precision* 100%, *recall* 99.99%, *F1-score* 99.99%, dan *kappa score* 99.98%, yang menunjukkan efektivitas tinggi dalam mendeteksi serangan DDoS secara akurat dan andal.

Keyword :
Machine Learning, Random Forest, Feature Selection, CFS

ABSTRACT

Distributed Denial of Service (DDoS) is a serious threat to network services, especially in open and dynamic cloud environments. This study aims to detect DDoS attacks using machine learning algorithms, namely Random Forest (RF) and Support Vector Machine (SVM), and evaluate the effect of feature selection using Correlation-based Feature Selection (CFS) and Rough Set (RS). Experiments were conducted using datasets from Zenodo with 10-Fold cross-validation and evaluation based on accuracy, precision, recall, F1-score, and kappa metrics. The results show that the Random Forest model consistently provides the best performance compared to SVM. The best scheme, namely Random Forest with RS feature selection, produces an accuracy value of 99.99%, precision 100%, recall 99.99%, F1-score 99.99%, and kappa score 99.98%, which indicates high effectiveness in detecting DDoS attacks accurately and reliably.

1. PENDAHULUAN

Serangan Distributed *Denial of Service* (DDoS) merupakan ancaman serius yang berpotensi mengganggu ketersediaan layanan daring dengan membanjiri sistem target menggunakan lalu lintas data berlebih sehingga layanan menjadi lambat atau bahkan tidak dapat diakses sama sekali. Perkembangan teknologi *cloud computing*, yang menawarkan fleksibilitas tinggi dalam penyediaan sumber daya komputasi secara terdistribusi, secara bersamaan juga meningkatkan kerentanan sistem terhadap serangan DDoS. Oleh karena itu, deteksi dini dan mitigasi serangan DDoS pada lingkungan cloud menjadi aspek fundamental dalam menjaga ketersediaan (*availability*) dan keandalan (*reliability*) layanan digital.

Pendekatan berbasis *machine learning* (ML) telah banyak dikaji dan diterapkan dalam mendeteksi serangan DDoS karena kemampuannya dalam mengenali pola lalu lintas jaringan yang anomali secara otomatis [1]. Algoritma seperti *Random Forest* (RF) dan *Support Vector Machine* (SVM) menunjukkan performa yang

baik dalam melakukan klasifikasi antara trafik benign dan trafik serangan [2]. Namun demikian, efektivitas setiap algoritma sangat dipengaruhi oleh karakteristik dataset serta proses pre-processing data, termasuk tahap seleksi fitur dan standardisasi. Seleksi fitur berperan penting dalam menentukan atribut paling relevan sehingga model dapat melakukan proses pembelajaran secara efisien sekaligus mengurangi risiko *overfitting* [3]. Metode seperti *Correlation-based Feature Selection* (CFS) dan *Rough Set* (RS) mampu menyeleksi fitur yang memiliki kontribusi signifikan terhadap hasil klasifikasi. Selain itu, proses standardisasi data diperlukan terutama bagi algoritma yang sensitif terhadap skala fitur, seperti SVM. Dalam konteks validasi model, teknik *10-Fold Cross Validation* digunakan secara luas untuk menghasilkan estimasi performa yang stabil dan meminimalkan bias evaluasi [4]. Evaluasi performa model umumnya dilakukan dengan berbagai metrik, antara lain *accuracy, precision, recall, F1-score, dan kappa score*, guna memberikan gambaran komprehensif terhadap kemampuan model dalam mendeteksi serangan [5].

Penelitian sebelumnya telah menunjukkan keunggulan masing-masing algoritma dalam mendeteksi serangan DDoS pada berbagai dataset, seperti CICDDoS2019 dan dataset berbasis cloud [6] [7] [8]. Namun, sebagian besar studi tersebut hanya berfokus pada peningkatan akurasi tanpa mempertimbangkan efek seleksi fitur terhadap performa model maupun uji signifikansi statistik antar algoritma [9]. Selain itu, penelitian yang secara eksplisit membandingkan algoritma RF dan SVM dengan penerapan metode seleksi fitur CFS dan RS pada dataset *cloud* yang diunduh dari Zenodo masih sangat terbatas. Uji statistik non-parametrik seperti *Mann–Whitney U Test* untuk mengevaluasi perbedaan performa antar model juga jarang diintegrasikan ke dalam proses analisis hasil. Dengan demikian, penelitian ini mengisi celah penelitian (research gap) tersebut dengan melakukan analisis perbandingan performa RF dan SVM yang dilengkapi seleksi fitur CFS dan RS, serta pengujian signifikansi perbedaan hasil menggunakan uji statistik non-parametrik.

Perkembangan ekosistem komputasi *cloud-based* seperti Google Colab turut memberikan kemudahan dalam eksperimen dan pengembangan model ML secara interaktif, dengan dukungan akses terhadap dataset besar serta pustaka komputasi seperti *scikit-learn, pandas, dan matplotlib* [10]. Lingkungan ini mendukung replikasi eksperimen dan penerapan sistem deteksi secara praktis dalam skala besar [11]. Beberapa penelitian mutakhir juga menegaskan bahwa integrasi *machine learning* dengan metode seleksi fitur berbasis korelasi dan teori himpunan kasar dapat meningkatkan efisiensi deteksi anomali pada lalu lintas jaringan *cloud* [12] [13] [14].

Berdasarkan latar belakang tersebut, penelitian ini bertujuan untuk melakukan perbandingan performa dua algoritma *machine learning* utama, yaitu RF dan SVM, dalam mendeteksi serangan DDoS pada jaringan cloud. Eksperimen dilakukan menggunakan AISED Dataset on Cloud DDoS Attacks yang diunduh dari Zenodo (https://zenodo.org/records/14681803), dengan tahapan *pre-processing* data, seleksi fitur, dan validasi yang terukur untuk menghasilkan hasil yang valid, reliabel, serta aplikatif. Penelitian ini diharapkan dapat memberikan kontribusi signifikan dalam pengembangan sistem deteksi serangan DDoS berbasis *machine learning* yang efisien dan efektif di lingkungan *cloud computing*, sekaligus memperkaya kajian ilmiah dengan pembuktian signifikansi statistik atas performa antar model yang diuji.

2. METODE PENELITIAN

Bagian ini menjelaskan tahapan penelitian yang dilakukan untuk mendeteksi serangan DDoS pada jaringan *cloud* menggunakan algoritma *machine learning*. Pendekatan yang digunakan dirancang untuk memastikan hasil eksperimen dapat diukur secara objektif dan direplikasi oleh penelitian selanjutnya.

2.1. Dataset

Penelitian ini menggunakan AIS Dataset on Cloud DDoS Attacks yang diperoleh dari repositori Zenodo (https://zenodo.org/records/14681803). Dataset ini dirancang untuk mendukung penelitian dalam bidang keamanan jaringan, khususnya dalam deteksi serangan DDoS pada lingkungan *cloud computing*. Data yang tersedia merepresentasikan lalu lintas jaringan dengan berbagai jenis fitur yang relevan untuk analisis serangan siber.

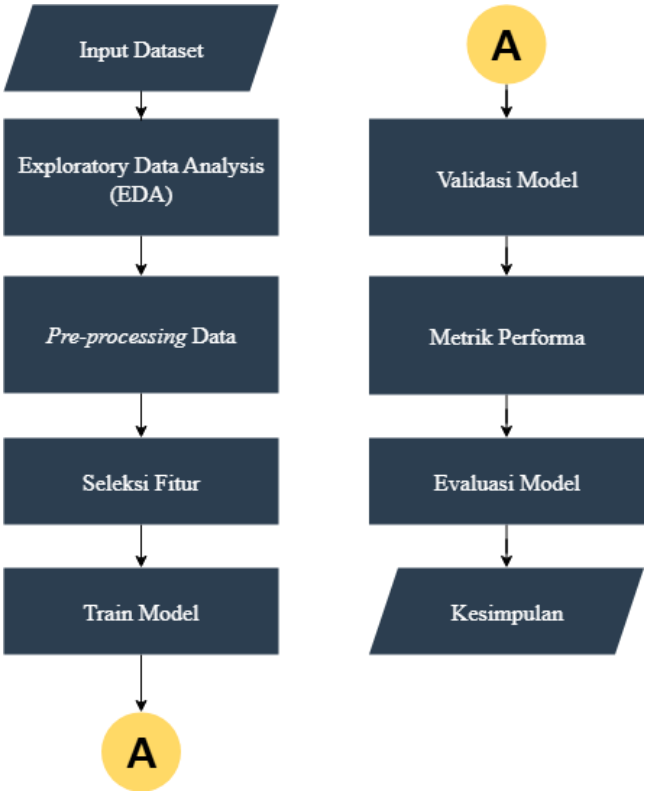
Dataset terdiri dari 52,318 *instance* dan 80 atribut (fitur), di mana setiap *instance* merepresentasikan satu aliran (*flow*) data jaringan yang dikumpulkan. Tidak ditemukan adanya nilai kosong (*missing value*) pada dataset ini, sehingga proses pre-processing data data dapat difokuskan pada pembersihan format dan normalisasi nilai. Fitur-fitur yang tersedia mencakup informasi detail terkait aliran paket, seperti:

a. Informasi jaringan seperti *Source Port, Destination Port*, dan *Protocol* yang mengindikasikan nomor port asal, port tujuan, serta protokol komunikasi yang digunakan [15].

b. Karakteristik lalu lintas seperti *Flow Duration*, *Total Fwd Packets*, *Total Backward Packets*, *Total Length of Fwd Packets*, dan *Total Length of Bwd Packets*, yang menggambarkan jumlah paket dan total panjang data yang dikirim ke arah maju maupun mundur [15].
c. Statistik paket seperti nilai maksimum, minimum, rata-rata, dan standar deviasi panjang paket, baik pada arah forward maupun backward.
d. Fitur berbasis waktu seperti Interval antar paket, *Flow Bytes/s*, dan *Flow Packets/s* yang memberikan gambaran intensitas lalu lintas dalam periode waktu tertentu [15].
e. Fitur berbasis header TCP/UDP Seperti jumlah flag URG, ACK, PSH, dan RST yang terdeteksi.
f. Indikator keamanan seperti Fitur-fitur yang dihasilkan dari analisis perilaku aliran, yang menjadi indikator potensi aktivitas berbahaya.
g. Label target pada dataset ini dibagi menjadi dua kelas utama, yaitu BENIGN (lalu lintas normal) dan DDoS (lalu lintas yang terindikasi sebagai serangan DDoS).

2.2. Tahapan Penelitian

Pada penelitian ini, *flowchart* digunakan untuk memvisualisasikan proses deteksi serangan DDoS pada jaringan *cloud* menggunakan algoritma *machine learning*. Visualisasi ini membantu memetakan langkah-langkah yang ditempuh, mulai dari pengumpulan dataset, *preprocessing* data, pemilihan dan penerapan algoritma, hingga evaluasi performa model. *Flowchart* tahapan penelitian dalam studi ini disajikan pada Gambar 1. yang kemudian akan dijelaskan secara rinci pada bagian berikutnya.



Gambar 1. Tahapan Penelitian

Proses penelitian dilakukan melalui beberapa tahapan yang sistematis dan terstruktur sesuai dengan alur kerja yang telah dirancang dalam *flowchart* pada Gambar 1. Penjelasan rinci setiap tahap pada flowchart adalah sebagai berikut:

1. Menginput dataset: Dataset *AIS Dataset on Cloud DDoS Attacks* dimuat ke lingkungan komputasi (Google Colab) dari penyimpanan google drive.
2. Melakukan *exploratory data analysis* (EDA): Tahap ini memeriksa dimensi data, konsistensi tipe/format, distribusi label, serta nilai unik pada kolom target. Analisis korelasi antarfitur dilakukan untuk mengidentifikasi atribut yang saling sangat berkaitan (indikasi redundansi) dan memberi gambaran awal bagi strategi seleksi fitur.
3. Melakukan *preprocessing* data, pada langkah ini mencakup 4 bagian yaitu:

a. *Encoding label*: Label kelas dikodekan ke bentuk numerik biner agar kompatibel dengan algoritma *machine learning* (mis. 0 = BENIGN, 1 = DDOS). Pengkodean yang eksplisit menjaga konsistensi selama pelatihan dan evaluasi.
b. Pemisahan fitur dan label: Seluruh kolom prediktor dikelompokkan sebagai matriks X, sementara kolom target sebagai vektor y. Pemisahan ini menjadi dasar penerapan *pipeline* pelatihan, validasi dan evaluasi.
c. Standardisasi: Fitur numerik distandarisasi (rata-rata 0, deviasi baku 1) menggunakan *StandardScaler*. Standardisasi menurunkan skala yang tidak seragam antarfitur, penting terutama bagi model yang sensitif pada skala (seperti SVM), serta membantu konvergensi dan stabilitas numerik.
d. Rekonstruksi *data frame*: Hasil penskalaan dikembalikan ke bentuk *data frame* dengan nama kolom yang terjaga. Langkah ini memudahkan penelusuran fitur, pelaporan, dan integrasi dengan modul seleksi fitur.

4. Menetapkan seleksi fitur: Pada penelitian ini menggunakan dua seleksi fitur yaitu :
   a. *Correlation-based Feature Selection* (CFS), memilih subset fitur dengan korelasi tinggi terhadap kelas namun rendah antarfitur, sehingga mengurangi redundansi.
   b. *Rough Set* mengevaluasi *dependency degree* dan menghasilkan *reduct* (subset minimal) yang memadai untuk mempertahankan kemampuan klasifikasi.
5. Pelatihan model (*random forest* dan SVM): Tiga skenario dilatih untuk tiap algoritma yaitu *baseline*, CFS, dan *Rough Set*. Pemilihan algoritma didasarkan pada karakteristik dan kemampuan representatif masing-masing metode. *Random Forest* dipilih karena kemampuannya dalam menangani data berdimensi tinggi, toleransinya terhadap outlier, serta kemampuannya mengatasi hubungan non-linear antar fitur melalui ensemble *decision tree*. Sementara itu, *Support Vector Machine* (SVM) digunakan sebagai pembanding karena sifatnya yang berbasis margin maksimum dan efektif dalam menangani permasalahan klasifikasi biner serta deteksi anomali pada lalu lintas jaringan. *Random Forest* dimanfaatkan untuk menangani hubungan nonlinier dan interaksi fitur, sementara SVM dipilih sebagai pembanding margin-maksimum.
6. Validasi model (*10-fold cross validation*): Proses pelatihan dievaluasi menggunakan 10-*fold cross validation* yang stratified terhadap label. Skema ini meminimalkan *overfitting* dan memberikan estimasi kinerja yang lebih stabil melalui agregasi performa lintas fold.
7. Metrik performa: Kinerja diukur menggunakan *accuracy*, *precision*, *recall*, *F1-Score*, dan *kappa Score*. *Accuracy* mengukur persentase prediksi benar dari seluruh data uji. *Precision* mengukur ketepatan prediksi kelas positif oleh model. *Recall* mengukur kemampuan model dalam menangkap semua sampel kelas positif. *F1-Score* adalah rata-rata harmonis dari *precision* dan *recall*, memberikan penilaian seimbang antara keduanya. *Kappa score* digunakan untuk mengukur tingkat kesepakatan hasil klasifikasi model terhadap label sebenarnya diluar kemungkinan kesepakatan acak. Kelima metrik ini saling melengkapi untuk memberikan gambaran lengkap mengenai performa sebuah model klasifikasi pada berbagai aspek evaluasi [12].
8. Evaluasi Model (Uji *Mann–Whitney U*): Perbandingan antar skenario dilakukan menggunakan *Mann-Whitney U* yaitu metode uji non-parametrik yang cocok diterapkan pada data yang tidak berdistribusi normal [13]. Uji ini dilakukan untuk menilai apakah perbedaan kinerja bersifat signifikan secara statistik ($\alpha = 0.05$).

3. HASIL DAN PEMBAHASAN

3.1. Hasil Ekperimen

Sub-bab ini disajikan hasil eksperimen dari implementasi dua algoritma *machine learning* yaitu *random forest* dan SVM)dalam mendeteksi serangan DDoS di jaringan *cloud*. Setiap model dievaluasi menggunakan metode *10-Fold Cross Validation* untuk memastikan keandalan dan menghindari *overfitting*.

3.1.1. Random Forest

Pada sub-bab ini, *random forest* diimplementasikan dalam tiga skenario berbeda, yaitu *baseline*/tanpa seleksi fitur, seleksi fitur menggunakan *correlation-based feature selection* (CFS) dan seleksi fitur menggunakan *rough set*. Hasil evaluasi kinerja model dalam skenario tersebut dapat dilihat pada Tabel 1.

Tabel 1. Evaluasi Rata-Rata 10-Fold Model Random Forest

Model	*Accuracy*	*Precision*	*Recall*	*F1-Score*	Kappa
Baseline	0.9998	0.9999	0.9997	0.9998	0.9996
CFS	0.9998	0.9999	0.9997	0.9998	0.9997
Rough Set	0.9999	1.0000	0.9999	0.9999	0.9998

Berdasarkan hasil evaluasi kinerja pada Tabel 1, model *random forest* menunjukkan performa yang sangat tinggi pada ketiga skenario pengujian. Pada skenario *baseline*, model mencapai nilai akurasi sebesar 0.9998, *precision* 0.9999, *recall* 0.9997, *F1-Score* 0.9998, dan nilai Kappa 0.9996. Hasil ini menunjukkan bahwa tanpa penerapan seleksi fitur sekalipun, *random forest* mampu mendeteksi serangan DDoS dengan tingkat keakuratan yang sangat tinggi. Sementara itu, penerapan metode seleksi fitur *correlation-based feature selection* (CFS) menghasilkan nilai akurasi yang sama dengan skenario *baseline*, namun nilai Kappa sedikit meningkat menjadi 0.9997. Peningkatan ini mengindikasikan bahwa CFS mampu memberikan kontribusi positif dalam meningkatkan kesepakatan prediksi model terhadap label sebenarnya.

Penerapan metode seleksi fitur *rough set* memberikan hasil yang sedikit lebih baik dibandingkan kedua skenario sebelumnya. Pada skenario ini, model mencapai akurasi sebesar 0.9999, *precision* 1.0000, *recall* 0.9999, *F1-Score* 0.9999, dan Kappa 0.9998. Nilai presisi sempurna menunjukkan bahwa model dengan *rough set* dapat mengklasifikasikan serangan DDoS tanpa menghasilkan kesalahan positif (*false positive*). Selain itu, kenaikan nilai Kappa pada skenario ini mengindikasikan peningkatan konsistensi prediksi model. Secara keseluruhan, meskipun perbedaan performa antar skenario relatif kecil, penerapan metode seleksi fitur, khususnya *rough set*, memberikan hasil yang lebih optimal dibandingkan *baseline* dan CFS. Oleh karena itu, *rough set* dapat menjadi pilihan metode seleksi fiture yang unggul dalam mendeteksi serangan DDoS di jaringan *cloud*.

3.1.2. *Support Vector Machine*

Pada sub-bab ini, SVM juga diimplementasikan dalam tiga skenario berbeda, yaitu *baseline*/tanpa seleksi fitur, seleksi fitur menggunakan CFS dan seleksi fitur menggunakan *Rough Set*. Hasil evaluasi kinerja model dalam ketiga skenario tersebut dapat dilihat pada Tabel 2.

Tabel 2. Evaluasi Rata-Rata 10-Fold Model SVM

Model	Accuracy	Precision	Recall	F1-Score	Kappa
Baseline	0.9980	0.9977	0.9984	0.9980	0.9960
CFS	0.9982	0.9973	0.9991	0.9982	0.9964
Rough Set	0.9979	0.9979	0.9981	0.9980	0.9954

Berdasarkan hasil evaluasi pada Tabel 2, model SVM menunjukkan kinerja yang sangat tinggi pada seluruh skenario pengujian, baik pada *baseline* maupun setelah dilakukan seleksi fitur menggunakan metode CFS dan *Rough Set*. Pada *baseline*, nilai akurasi mencapai 0.9980 dengan *precision* sebesar 0.9977, *recall* sebesar 0.9984, dan *F1-Score* sebesar 0.9980. Nilai Kappa yang diperoleh sebesar 0.9960, menunjukkan tingkat kesepakatan yang sangat kuat antara prediksi model dan label sebenarnya. Penerapan seleksi fitur menggunakan metode CFS sedikit meningkatkan akurasi menjadi 0.9982 dan *recall* menjadi 0.9991, yang mengindikasikan bahwa metode ini dapat membantu model mengidentifikasi serangan DDoS dengan lebih sensitif tanpa mengurangi ketepatan prediksi secara signifikan.

Sementara itu, penggunaan metode *rough set* menghasilkan akurasi sebesar 0.9979 dengan *precision* 0.9979, *recall* 0.9981, dan *F1-Score* 0.9980. Meskipun sedikit lebih rendah dibandingkan metode CFS, perbedaan yang dihasilkan relatif kecil, sehingga efektivitas kedua metode seleksi fitur dapat dikatakan setara dalam konteks ini. Secara keseluruhan, kinerja SVM pada semua skenario berada pada tingkat yang konsisten tinggi, dengan perbedaan metrik yang sangat tipis. Hal ini menunjukkan bahwa SVM memiliki ketahanan performa yang baik baik pada data mentah maupun data yang telah melalui proses seleksi fitur, sehingga dapat diandalkan untuk mendeteksi serangan DDoS di lingkungan jaringan *cloud*.

3.2. Pembahasan Hasil Ekperiment

Pada sub-bab ini disajikan pembahasan hasil eksperimen dari implementasi dua algoritma *machine learning*, yaitu *random forest* dan SVM dalam mendeteksi serangan DDoS. Setelah memperoleh hasil dari kedua evaluasi performa model, dilakukan analisis lebih lanjut dengan membandingkan performa antar model melalui uji statistik *Mann-Whitney U* untuk mengetahui apakah terdapat perbedaan signifikan antar model.

3.2.1. Perbandingan Performa Antar Model Yang Sama

3.2.1.1 *Random Forest*

Pada bagian ini dilakukan analisis perbandingan performa model *random forest baseline* dengan dua model hasil seleksi fitur, yaitu *random forest* + CFS dan *random forest + rough set*. Tujuan dari perbandingan ini adalah untuk mengevaluasi apakah proses seleksi fitur dapat memberikan peningkatan performa yang signifikan dibandingkan model *baseline* tanpa seleksi fitur. Hasil dari uji statistik dapat dilihat pada Tabel 3.

Tabel 3. Uji *Mann-Whitney* pada Model RF

Statistik *Mann-Whitney U*		Baseline RF vs RF CFS	Baseline RF vs RF RS	RF CFS vs RF RS
Accuracy	*U-Statistic*	44.5	38.5	46
	P-Value	0.6937	0.3878	0.7813
	Signifikan	Tidak	Tidak	Tidak
Precision	*U-Statistic*	44	39	44.5
	P-Value	0.5842	0.2558	0.5428
	Signifikan	Tidak	Tidak	Tidak
Recall	*U-Statistic*	50	48.5	48.5
	P-Value	1.0000	0.9346	0.9346
	Signifikan	Tidak	Tidak	Tidak
F1-Score	*U-Statistic*	45	39.5	46.5
	P-Value	0.7258	0.4351	0.8128
	Signifikan	Tidak	Tidak	Tidak
Kappa	*U-Statistic*	44	37	45
	P-Value	0.6692	0.3326	0.7242
	Signifikan	Tidak	Tidak	Tidak

Berdasarkan hasil uji statistik *Mann-Whitney* pada Tabel 3, model *random forest* dengan tiga skenario pembandingan, yaitu *baseline* RF vs RF CFS, *baseline* RF vs RF RS, serta RF CFS vs RF RS, diperoleh bahwa tidak terdapat perbedaan yang signifikan secara statistik pada seluruh metrik evaluasi yang diuji, meliputi *accuracy*, *precision, recall, F1-score*, dan *kappa*. Hal ini ditunjukkan oleh nilai *p-value* yang secara konsisten berada di atas ambang signifikansi 0.05 pada semua perbandingan, dengan rentang *p-value* antara 0.2558 hingga 1.0000. Temuan ini mengindikasikan bahwa penerapan metode seleksi fitur, baik CFS maupun *rough set*, tidak memberikan perbedaan performa yang signifikan dibandingkan model *baseline random forest* dalam mendeteksi serangan DDoS pada dataset yang digunakan. Dengan demikian, meskipun metode seleksi fitur berpotensi menyederhanakan kompleksitas model melalui pengurangan jumlah atribut, dampaknya terhadap peningkatan kinerja model *random forest* pada kasus ini tidak terdeteksi secara signifikan secara statistik.

Temuan ini dapat dijelaskan oleh sifat dasar algoritma *Random Forest* yang memiliki kemampuan bawaan dalam melakukan *feature subsampling* dan *bagging*, sehingga model ini relatif tahan terhadap kehadiran fitur-fitur yang kurang relevan (*robust against irrelevant features*). Mekanisme pemilihan subset fitur secara acak pada setiap *Decision Tree* membuat *Random Forest* mampu menjaga performanya bahkan ketika seluruh fitur disertakan tanpa proses seleksi eksplisit. Dengan demikian, pada dataset ini, proses seleksi fitur CFS dan *Rough Set* tidak memberikan keuntungan yang berarti karena model *Random Forest* sudah secara inheren melakukan reduksi kompleksitas secara internal. Implikasi praktis dari hasil ini adalah bahwa penerapan seleksi fitur tambahan pada model *Random Forest* mungkin tidak diperlukan jika tujuan utama adalah efisiensi komputasi tanpa mengorbankan akurasi deteksi.

3.2.1.2 *Support Vector Machine*

Pada bagian ini dilakukan analisis perbandingan performa model SVM *baseline* dengan dua model hasil seleksi fitur, yaitu SVM + CFS dan SVM + *rough set*. Tujuan dari perbandingan ini adalah untuk mengevaluasi apakah proses seleksi fitur dapat memberikan peningkatan performa yang signifikan dibandingkan model *baseline* tanpa seleksi fitur. Hasil dari uji statistik dapat dilihat pada Tabel 4.

Tabel 4. Uji *Mann-Whitney* pada Model SVM

Statistik Mann-Whitney		Baseline SVM vs SVM CFS	Baseline SVM vs SVM RS	SVM CFS vs SVM RS
Accuracy	*U-Statistic*	38	56	63.5
	P-Value	0.3835	0.6758	0.3231
	Signifikan	Tidak	Tidak	Tidak
Precision	*U-Statistic*	57	54	48
	P-Value	0.6226	0.7904	0.9093
	Signifikan	Tidak	Tidak	Tidak
Recall	*U-Statistic*	19.5	58.5	92
	P-Value	0.0223	0.5422	0.0015
	Signifikan	Signifikan	Tidak	**Signifikan**
F1-Score	*U-Statistic*	37	54	69
	P-Value	0.3447	0.7908	0.1613
	Signifikan	Tidak	Tidak	Tidak
Kappa	*U-Statistic*	40	56.5	60
	P-Value	0.4726	0.6499	0.4725
	Signifikan	Tidak	Tidak	Tidak

Berdasarkan hasil uji statistik *Mann-Whitney* pada Tabel 4., model SVM dengan tiga skenario perbandingan (*baseline* vs CFS, *baseline* vs *rough set*, dan CFS vs *rough set*) terhadap lima metrik evaluasi, diperoleh temuan bahwa perbedaan signifikan hanya terjadi pada metrik Recall. Pada skenario *baseline* vs CFS, nilai *p-value* sebesar 0.0223 (< 0.05) menunjukkan bahwa seleksi fitur CFS memberikan perubahan signifikan terhadap kemampuan model dalam mengenali kelas positif dibandingkan *baseline*. Sementara itu, pada skenario CFS vs *rough set*, nilai *p-value* sebesar 0.0015 (< 0.05) juga mengindikasikan adanya perbedaan signifikan dalam metrik yang sama, yang berarti kedua metode seleksi fitur tersebut menghasilkan performa deteksi yang berbeda pada aspek sensitivitas model. Adapun pada metrik *accuracy*, *precision*, *F1-Score*, dan *kappa*, seluruh skenario menghasilkan nilai *p-value* di atas 0.05, sehingga tidak terdapat perbedaan signifikan secara statistik. Temuan ini mengindikasikan bahwa pada model SVM, perubahan metode seleksi fitur lebih berpengaruh terhadap *recall* dibandingkan metrik evaluasi lainnya, yang dapat berkaitan dengan sifat seleksi fitur dalam mempertahankan atribut yang relevan untuk mendeteksi serangan.

Hasil ini menunjukkan bahwa algoritma SVM lebih sensitif terhadap perubahan komposisi fitur yang berpengaruh langsung pada kemampuan model dalam mengenali pola minoritas atau kelas positif, dalam hal ini lalu lintas DDoS. SVM bekerja dengan prinsip margin maksimum, sehingga fitur-fitur yang paling relevan terhadap pemisahan antar kelas akan sangat menentukan posisi *hyperplane*. Seleksi fitur menggunakan CFS dan *Rough Set* dapat mengubah ruang fitur secara signifikan, yang pada akhirnya memengaruhi *recall* lebih kuat daripada metrik lainnya. Dengan kata lain, metode seleksi fitur mampu memperbaiki sensitivitas deteksi serangan, meskipun tidak selalu meningkatkan akurasi keseluruhan. Dari sisi praktis, hal ini menunjukkan bahwa untuk skenario deteksi ancaman siber di mana kesalahan tipe II (gagal mendeteksi serangan) memiliki konsekuensi tinggi, penerapan seleksi fitur pada SVM dapat menjadi strategi efektif untuk meningkatkan tingkat deteksi tanpa mengorbankan stabilitas metrik lain. Sebaliknya, untuk kasus di mana efisiensi komputasi lebih diprioritaskan, model baseline SVM sudah cukup memadai karena perbedaan performa pada metrik lainnya tidak signifikan.

3.2.2. Perbandingan Performa Antar Model Yang Berbeda

3.2.2.1. Perbandingan Performa Antar Baseline

Pada bagian ini dilakukan pembahasan mengenai perbandingan performa antar model *random forest* dan SVM pada kondisi *baseline*, yaitu tanpa penerapan metode seleksi fitur apapun. Analisis perbandingan dilakukan dengan menggunakan uji statistik *Mann-Whitney U-Test* terhadap lima metrik evaluasi utama, yaitu *accuracy, precision, recall, F1-Score,* dan *kappa*. Uji statistik ini dilakukan agar dapat diperoleh gambaran perbedaan performa yang lebih objektif di antara model *baseline*. Hasil dari pengujian *Mann-Whitney* dapat dilihat pada Tabel 5.

Tabel 5. Uji *Mann-Whitney* pada kedua *baseline*

Statistik Mann-Whitney		Baseline RF vs Baseline SVM
Accuracy	*U-Statistic*	100
	P-Value	0.000172
	Signifikan	Ya
Precision	*U-Statistic*	100
	P-Value	0.000132
	Signifikan	Ya
Recall	*U-Statistic*	96
	P-Value	0.000498
	Signifikan	Ya
F1-Score	*U-Statistic*	100
	P-Value	0.000177
	Signifikan	Ya
Kappa	*U-Statistic*	100
	P-Value	0.000178
	Signifikan	Ya

Hasil uji statistik *Mann-Whitney* pada Tabel 5, menunjukkan bahwa seluruh metrik evaluasi, yaitu *accuracy, precision, recall, F1-Score* dan *kappa*, memiliki nilai *p-value* yang lebih kecil dari 0.05, sehingga perbedaan kinerja kedua model dinyatakan signifikan secara statistik. Nilai *U-Statistic* yang konsisten tinggi pada hampir seluruh metrik mengindikasikan adanya perbedaan distribusi performa yang konsisten antara kedua model. Temuan ini menguatkan bahwa, dalam konteks data dan skenario pengujian yang digunakan, perbedaan performa antara *baseline random forest* dan *baseline* SVM bukanlah hasil dari variasi acak, melainkan mencerminkan perbedaan nyata dalam kemampuan kedua model dalam mendeteksi serangan DDoS pada jaringan *cloud*.

3.2.2.1. Perbandingan Performa Antar Seleksi Fitur

Pada bagian ini, juga akan dilakukan pembahasan mengenai perbandingan performa setelah kedua model diterapkan metode seleksi fitur, baik menggunakan CFS maupun *rough set* (RS). Pembahasan ini bertujuan untuk mengetahui apakah terdapat perbedaan performa yang signifikan antar model yang sama-sama menggunakan seleksi fitur, serta mengidentifikasi model dengan kombinasi algoritma dan metode seleksi fitur yang paling optimal dalam mendeteksi serangan DDoS. Berikut hasil dari beberapa uji statistik yang dilakukan:

1. RF CFS vs SVM CFS, dengan hasil seluruh metrik menunjukkan perbedaan signifikan ($p < 0.05$) pada *accuracy*, *precision*, *recall*, *F1-Score*, dan *kappa*.
2. RF CFS vs SVM RS, dengan hasil seluruh metrik menunjukkan perbedaan signifikan ($p < 0.05$) pada *accuracy*, *precision*, *recall*, *F1-Score*, dan *kappa*.
3. RF RS vs SVM CFS, dengan hasil seluruh metrik menunjukkan perbedaan signifikan ($p < 0.05$) pada *accuracy*, *precision*, *recall*, *F1-Score*, dan *kappa*.
4. RF RS vs SVM RS, dengan hasil seluruh metrik menunjukkan perbedaan signifikan ($p < 0.05$) pada *accuracy*, *precision*, *recall*, *F1-Score*, dan *kappa*.

Berdasarkan hasil uji *cross-comparison* antara *random forest* dan SVM dengan metode seleksi fitur CFS dan *rough set*, seluruh kombinasi perbandingan menunjukkan adanya perbedaan signifikan pada kelima metrik evaluasi, yaitu *Accuracy*, *Precision*, *Recall*, *F1-Score*, dan *Kappa* ($p < 0.05$). Pada perbandingan RF CFS vs SVM CFS dan RF CFS vs SVM RS, hasil uji mengindikasikan bahwa kinerja RF secara konsisten berbeda secara statistik dibandingkan SVM, baik ketika keduanya menggunakan metode seleksi fitur yang sama maupun berbeda. Hal serupa juga terlihat pada RF RS vs SVM CFS dan RF RS vs SVM RS, di mana perbedaan signifikan tetap terjadi pada semua metrik evaluasi. Temuan ini menunjukkan bahwa perbedaan kinerja antara RF dan SVM bukan hanya dipengaruhi oleh metode seleksi fitur yang digunakan, tetapi juga oleh karakteristik algoritma itu sendiri, sehingga model dengan metode seleksi fitur yang sama belum tentu memiliki performa yang setara.

4. KESIMPULAN

Berdasarkan hasil eksperimen dan analisis yang telah dilakukan, penelitian ini menunjukkan bahwa algoritma *random forest* dan SVM mampu mendeteksi serangan DDoS di jaringan *cloud* dengan performa yang sangat tinggi. Pada skenario terbaiknya, RF dengan metode *rough set* mencapai akurasi 0.9999, *precision* 1.0000, *recall* 0.9999, *F1-Score* 0.9999, dan kappa 0.9998. Sementara itu, SVM dengan metode CFS mencapai akurasi 0.9982, *precision* 0.9973, *recall* 0.9991, *F1-Score* 0.9982, dan kappa 0.9964. Hasil uji statistik *Mann-Whitney U* menunjukkan bahwa perbedaan performa antar algoritma signifikan pada seluruh metrik evaluasi (*p-value* < 0.05), termasuk pada *baseline* (RF vs SVM) yang masing-masing metrik memiliki p-value antara 0.000132 hingga 0.000498. Namun, perbedaan antar skenario dalam algoritma yang sama umumnya tidak signifikan secara statistik (*p-value* > 0.05), kecuali pada metrik Recall di SVM, di mana *baseline* vs CFS ($p = 0.0223$) dan CFS vs rough set ($p = 0.0015$) menunjukkan perbedaan signifikan. Kelebihan penelitian ini terletak pada konsistensi performa tinggi kedua algoritma di berbagai skenario serta kemampuan RF mempertahankan akurasi optimal dengan atau tanpa seleksi fitur. Kekurangannya adalah selisih performa antar skenario yang relatif kecil, sehingga manfaat seleksi fitur tidak selalu signifikan secara statistik. Berdasarkan temuan ini, RF dengan metode *rough set* direkomendasikan sebagai kombinasi terbaik untuk deteksi serangan DDoS di lingkungan *cloud*, sedangkan SVM dapat menjadi alternatif andal ketika sensitivitas deteksi (*recall*) menjadi prioritas utama.

DAFTAR PUSTAKA

[1] P. Bintoro, Ratnasari, E. Wihardjo, I. P. Putri dan A. Asari, Pengantar Machine Learning, Kota Solok, Sumatera Barat: PT MAFY MEDIA LITERASI INDONESIA, September 2024.
[2] T. Aytaç, M. A. Aydın dan A. H. Zaim, "Detection DDOS Attacks Using Machine Learning Methods," *Electrica*, vol. 20, pp. 159-167, 2020.
[3] S. Budiman, A. Sunyoto dan A. Nasiri, "Analisa Performa Penggunaan Feature Selection untuk Mendeteksi Intrusion Detection Systems dengan Algoritma Random Forest Classifier," *SISTEMASI*, vol. 10, p. 753, 2021.
[4] M. Kuhn dan K. Johnson, Applied Predictive Modeling, New York: Springer, 2013.
[5] D. G. Altman, Practical Statistics for Medical Research, London: Chapman and Hall/CRC, 1991.
[6] R. Amrish, "DDoS Detection using Machine Learning Techniques," *Journal of IoT in Social, Mobile, Analytics, and Cloud*, vol. 4, no. 1, pp. 24-32, 2022.
[7] Attou, Hanaa dan Azidine, "Cloud-Based Intrusion Detection Approach Using Machine Learning Techniques," *Big Data Mining and Analytics*, vol. 6, pp. 311-320, 2023.

[8] Beer, F. Buhler dan Ulrich, "Feature selection for flow-based intrusion detection using Rough Set Theory," *IEEE Explore*, pp. 617-624, 2017.
[9] Whitney dan Mann, "On a Test of Whether One of Two Random Variables is Stochastically Larger than the Other," *Annals of Mathematical Statistics,* vol. 18, pp. 50-60, 1974.
[10] F. Chollet, Deep Learning with Python, Manning Publications, 2021.
[11] V. Jayaswal, "Performance Metrics: Confusion matrix, Precision, Recall, and F1 Score," Towards Data Science, 14 September 2020. [Online]. Available: https://towardsdatascience.com/performance-metrics-confusion-matrix-precision-recall-and-f1-score-a8fe076a2262/. [Diakses 19 Agustus 2025].
[12] M. Fadli dan R. A. Saputra, "KLASIFIKASI DAN EVALUASI PERFORMA MODEL RANDOM FOREST UNTUK PREDIKSI STROKE," *Jurnal Teknik,* vol. 12, no. 2, pp. 72-80, 2023.
[13] Amorim, A. Fernandez dan Delgado, "Do we need hundreds of classifiers to solve real world classification problems?," *Journal of Machine Learning Research,* vol. 15, pp. 3133-3181, 2014.
[14] J. V. Ade dan A. V. Deorankar, "Ensemble Learning Methods for DDoS Attack Detection," *International Journal of Science and Engineering Applications,* vol. 13, no. 05, pp. 40-45, 2024.
[15] R. Gunawan, E. S. Handika dan E. Ismanto, "Pendekatan Machine Learning Dengan Menggunakan Algoritma Xgboost (ExtremeGradient Boosting) Untuk Peningkatan Kinerja Klasifikasi Serangan Syn," *Jurnal Computer Science and Information Technology (CoSciTech),* vol. 3 No 3, pp. 453-463, Desember 2022.