CICES (*Cyberpreneurship Innovative and Creative Exact and Social Science*)
DOI: 10.33050/p8mgnt69 ISSN: **2356-5209**
pp. 103~112 *Online* ISSN: **2655-3058**

Implementasi ANOVA Pada Algoritma KNN++ Untuk Klasifikasi Serangan DDoS UDP FLood

Muhammad Bagus Satrio[1], Fetty Tri Anggraeny*[2], Achmad Junaidi[3]
[1,2,3]Program Studi Informatika, Fakultas Ilmu Komputer, UPN "Veteran" Jawa Timur
E-mail: [1]riosatriambs@gmail.com,
*[2]**fettyanggraeny.if@upnjatim.ac.id**,[3]achmadjunaidi.if@upnjatim.ac.id

Abstrak

Keamanan siber menjadi isu kritis di era digital, terutama serangan Distributed Denial of Service (DDoS) UDP Flood yang meningkat signifikan hingga 59,77% pada tahun 2023. Penelitian ini mengimplementasikan algoritma K-Nearest Neighbors++ (KNN++) yang diperkuat dengan seleksi fitur Analysis of Variance (ANOVA) dan penyeimbangan data Synthetic Minority Oversampling Technique (SMOTE) untuk deteksi serangan DDoS UDP Flood berbasis anomali lalu lintas jaringan. Dataset CICDDoS2019 diproses melalui tahapan preprocessing menyeluruh termasuk penghapusan 20 fitur tidak relevan, normalisasi Min-Max, dan penyeimbangan distribusi kelas. Hasil eksperimen menunjukkan performa terbaik dengan parameter K=3 mencapai akurasi 99,994%, presisi 99,998%, recall 99,996%, dan F1-score 99,997%. Penerapan SMOTE berhasil meningkatkan recall kelas minoritas sebesar 2,51% dibandingkan model tanpa SMOTE, dengan waktu komputasi 3846 detik yang tergolong efisien untuk dataset berskala besar. Temuan ini membuktikan efektivitas kombinasi KNN++, ANOVA, dan SMOTE sebagai solusi komprehensif untuk sistem deteksi intrusi berbasis anomali.

Kata Kunci*—DDoS, KNN++, ANOVA, SMOTE, Klasifikasi Jaringan*

Abstract

Cybersecurity has become a critical issue in the digital era, particularly with the significant rise of Distributed Denial of Service (DDoS) UDP Flood attacks, which increased by 59.77% in 2023. This study implements the K-Nearest Neighbors++ (KNN++) algorithm, enhanced with feature selection using Analysis of Variance (ANOVA) and data balancing through the Synthetic Minority Oversampling Technique (SMOTE), to detect UDP Flood DDoS attacks based on network traffic anomalies. The CICDDoS2019 dataset is processed through comprehensive preprocessing stages, including the removal of 20 irrelevant features, Min-Max normalization, and class distribution balancing. Experimental results show the best performance at K=3, achieving an accuracy of 99.994%, precision of 99.998%, recall of 99.996%, and an F1-score of 99.997%. The application of SMOTE successfully increased the minority class recall by 2.51% compared to the model without SMOTE, with a computation time of 3,846 seconds, which is considered efficient for large-scale datasets. These findings demonstrate the effectiveness of the combined KNN++, ANOVA, and SMOTE approach as a comprehensive solution for anomaly-based intrusion detection systems.

Keywords*— DDoS, KNN++, ANOVA, SMOTE, Network Classification*

DOI: 10.33050/p8mgnt69 ISSN: **2356-5209**
pp. 103~112 *Online* ISSN: **2655-3058**

1. PENDAHULUAN

Keamanan siber merupakan isu krusial di era digital, seiring meningkatnya ketergantungan terhadap infrastruktur jaringan di berbagai sektor. Serangan Distributed Denial of Service (DDoS) menjadi ancaman signifikan, dengan metode UDP Flood menunjukkan tren peningkatan tajam. Berdasarkan laporan Qrator Labs, metode ini menyumbang 59,77% dari seluruh serangan DDoS pada kuartal kedua 2023, naik dari 37,44% [1]. Di Indonesia, laporan SOCRadar mencatat lebih dari 43.000 serangan DDoS sepanjang 2024, dengan intensitas mencapai 693 Gbps [2].

Untuk mengatasi ancaman tersebut, sistem deteksi intrusi berbasis anomali menjadi solusi penting. Algoritma machine learning seperti K-Nearest Neighbors (KNN) sering digunakan karena kesederhanaan dan efektivitasnya dalam klasifikasi berbasis similaritas [3][4]. Algoritma KNN bersifat supervised dimana hasil dari query instance yang baru diklasifikasikan berdasarkan mayoritas pada kategori [5]. Namun, performa KNN sangat bergantung pada pemilihan nilai k yang optimal. Nilai k yang tidak sesuai dapat menyebabkan penurunan akurasi [6], terlebih pada dataset besar dan tidak seimbang, di mana KNN cenderung bias terhadap kelas mayoritas [7].

Sebagai solusi, berbagai pengembangan terhadap KNN telah dilakukan. KNN++ mengadopsi pendekatan distance-aware untuk meningkatkan efisiensi dan akurasi klasifikasi melalui pengoptimalan struktur data [8].

Selain pemilihan algoritma, kualitas data juga memengaruhi kinerja model. Seleksi fitur menggunakan ANOVA dapat meningkatkan efisiensi dengan mengurangi dimensi data [9], sedangkan penyeimbangan data menggunakan SMOTE (Synthetic Minority Oversampling Technique) membantu mengatasi bias akibat ketidakseimbangan kelas [10][11].

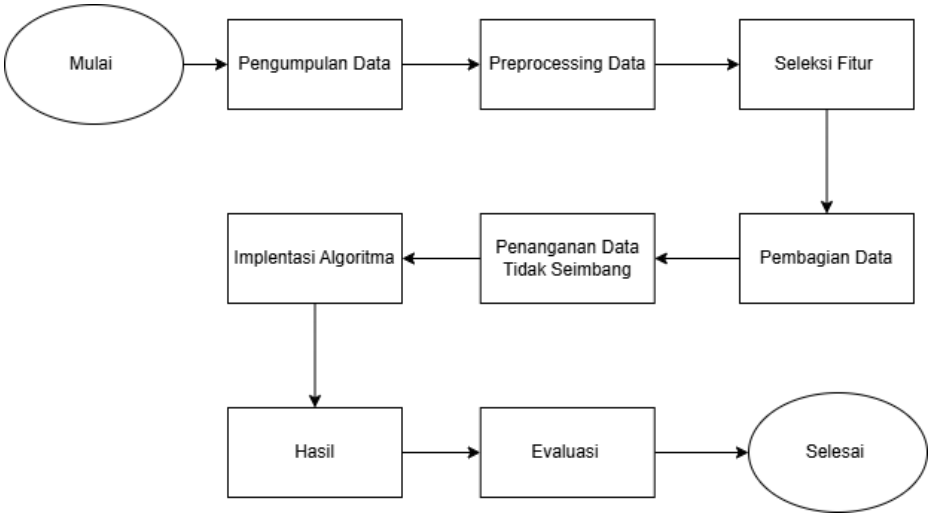
Penelitian sebelumnya oleh Lakshminarayana & Basarkod (2023) menunjukkan peningkatan performa KNN++ pada sistem IDS dengan peningkatan F1-score hingga 5,33% [8].

Tujuan penelitian ini adalah untuk mengimplementasikan algoritma KNN++ dalam mendeteksi serangan siber berbasis anomali, dengan dukungan seleksi fitur ANOVA dan teknik balancing data SMOTE, serta mengevaluasi kinerja model menggunakan metrik akurasi, presisi, recall, F1-score, dan ROC-AUC.

2. METODE PENELITIAN

Pada bagian ini akan dijelaskan setiap tahapan secara rinci yang dilalui selama proses penelitian. Berikut merupakan gambaran umum mengenai alur penelitian yang ditunjukkan, memperlihatkan urutan langkah-langkah yang diambil dalam pelaksanaan penelitian ini yang dijelaskan pada Gambar 1.

DOI: 10.33050/p8mgnt69 ISSN: **2356-5209**
pp. 103~112 *Online* ISSN: **2655-3058**



Gambar 1. Alur Penelitian

2.1 Pengumpulan Data

Pada penelitian ini, data yang digunakan untuk menganalisis serangan DDoS UDP Flood bersumber dari dataset CICDDoS 2019 (Canadian Institute for Cybersecurity Distributed Denial of Service 2019). Dataset ini dipilih karena menyediakan berbagai jenis serangan DDoS, termasuk UDP Flood, dalam lingkungan yang realistis dan terlabeli dengan baik.

2.2 Preprocessing Data

Preprocessing data meliputi beberapa tahapan. Pertama, penghapusan fitur tidak relevan dilakukan dengan menghilangkan atribut bertipe string, nilai konstan, atau yang tidak berkontribusi pada prediksi. Selanjutnya, pembersihan data mencakup penanganan *missing values*, eliminasi nilai negatif, dan penghapusan duplikat. Untuk klasifikasi, label kategorikal ("Benign" dan "DrDoS_UDP") diubah menjadi numerik melalui *label encoding* (0 dan 1). Terakhir, normalisasi dengan *Min-Max Scaling* diaplikasikan untuk menstandarisasi rentang fitur numerik ke interval [0, 1], memastikan keseragaman skala sebelum pemodelan.

2.3 Seleksi Fitur ANOVA

Analysis of Variance (ANOVA) merupakan teknik seleksi fitur yang digunakan untuk memilih fitur yang paling relevan dalam model klasifkasi. Dalam ANOVA, nilai F-Score dihitung untuk setiap fitur, yang mengukur sejauh mana variasi antar kelompok data lebih besar dibandingkan dengan variasi dalam kelompok tersebut. Penghitungan terkait ANOVA dijelaskan sebagai berikut : [12]

1. Derajat Kebebasan Antar Kelas (dfb)

$$dfb = g - 1 \quad (1)$$

2. Derajat Kebebasan Dalam Kelas (dfw)

$$dfw = N - g \quad (2)$$

3. Variabilitas Antar Kelas (SSB)

DOI: 10.33050/p8mgnt69 ISSN: **2356-5209**
pp. 103~112 *Online* ISSN: **2655-3058**

$$SSB = \sum_{i=1}^{g} n_i (X_i - X)^2 \quad (3)$$

4. Variabilitas Dalam Kelas (SSW)

$$SSW = \sum_{i=1}^{g} \sum_{j=1}^{n_i} \left(X_{ij} - X_i\right)^2 \quad (4)$$

5. Penghitungan Nilai F

$$F = \frac{(SSB/df_b)}{(SSW/df_w)} \quad (5)$$

Keterangan :

dfb = Derajat Kebebasan Antar Kelas

dfw = Derajat Kebebasan Dalam Kelas

SSB = Variabilitas Antar Kelas

SSW = Variabilitas Dalam Kelas

n_i = Ukuran sampel kelompok ke-i

x_i = Rata-rata kelompok ke-I

X = Rata rata keseluruhan

g = Jumlah kelas

N = Jumlah total observasi

X_{ij} = Nilai observasi ke-j pada kelas ke-i.

F = Statistik nilai F

2. 4 Pembagian Data

Setelah melalui seleksi fitur, dataset dibagi menjadi dua bagian, yaitu data pelatihan (data training) dan data pengujian (data testing). Hal ini dilakukan untuk memastikan model yang dibangun dapat memprediksi data yang belum pernah dilihat sebelumnya. Dalam penelitian ini, dataset dibagi dengan proporsi 80% untuk data pelatihan dan 20% untuk data pengujian

2.5 SMOTE (Synthetic Minority Oversampling Technique)

SMOTE (Synthetic Minority Oversampling Technique) adalah metode oversampling yang dirancang untuk mengatasi masalah ketidakseimbangan kelas dalam dataset. Teknik ini bekerja dengan menghasilkan sampel sintetik untuk kelas minoritas melalui interpolasi antara data minoritas yang ada dan tetangganya. SMOTE memungkinkan representasi kelas minoritas

DOI: 10.33050/p8mgnt69 ISSN: **2356-5209**
pp. 103~112 *Online* ISSN: **2655-3058**

menjadi lebih seimbang, sehingga model pembelajaran tidak bias terhadap kelas mayoritas. Selain itu, metode ini dapat meningkatkan performa prediksi pada dataset yang tidak seimbang Penghitungan terkait *Synthetic Minority Oversampling Technique* dijelaskan sebagai berikut *:* [13]

$$X_{new} = X + (X_{ch} - X) \text{ x } random\ (0,1) \quad (6)$$

Keterangan :

X_{new} = Sampel sintetis baru
X = Titik asli data minoritas
X_{ch} = Tetangga terdekat dari titik X

2.6 KNN++ (K-Nearest Neighbors ++)

KNN++ merupakan pengembangan dari algoritma K-Nearest Neighbor (KNN) tradisional yang dirancang untuk meningkatkan efisiensi dan akurasi klasifikasi, khususnya dalam deteksi intrusi jaringan. Algoritma ini memperkenalkan pendekatan distance-aware dengan memanfaatkan analisis statistik seperti varians untuk mengoptimalkan proses pencarian tetangga terdekat. Penghitungan KNN++ dijelaskan sebagai berikut : [14] [8]

1. Penghitungan Jarak Euclidean

$$d(x,y) = \sqrt{\sum_{i=1}^{m}(x_i - y_i)^2} \quad (7)$$

Keterangan :

d(x,y) = Jarak antara dua titik x dan y
x_i = Nilai fitur ke-i dari titik x
y_i = Nilai fitur ke-i dari titik y
m = Jumlah fitur dalam data

2. Menghitung Rata Rata Setiap Fitur pada data latih

$$\overrightarrow{\mu} = \frac{1}{D}\left(\sum_{j=1}^{D} y_{ij}\right) \quad (8)$$

Keterangan :

i = Fitur ke – i
j = Data ke - J
D = Total jumlah data pada fitur J

3. Menghitung Rata Rata Selisih Kuadrat Setiap Fitur pada data latih

$$\sigma_i^2 = \frac{\sum_{j=1}^{D}(y_{ij} - \mu_j)^2}{(D-1)} \quad (9)$$

Keterangan :

Yij = Data Ke J pada Fitur i
μ_j = Rata Rata Fitur J
D = Total jumlah data pada fitur

4. Menghitung k tetangga terdekat dengan jarak terkecil

DOI: 10.33050/p8mgnt69 ISSN: **2356-5209**
pp. 103~112 *Online* ISSN: **2655-3058**

$$D(X', y_t) = \sqrt{\sum_{u=1}^{n} (X' - y_{tu})^2} \quad (10)$$

Keterangan :
X' = Data Uji
Y_{tu} = Data latih ke- t pada fitur U

2.7 Evaluasi Model

Evaluasi model dilakukan menggunakan *confusion matrix*, *Confusion matrix* menyajikan informasi dalam bentuk tabel yang menunjukkan jumlah prediksi yang benar dan salah dari setiap kelas, mencakup *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN). Berdasarkan nilai-nilai ini, matriks evaluasi yang dihitung meliputi akurasi, presisi, recall, F1-score dan ROC-AUC. Akurasi mengukur proporsi prediksi yang benar terhadap seluruh prediksi yang dilakukan. Presisi menunjukkan seberapa banyak prediksi positif yang benar-benar relevan. Recall menilai kemampuan model dalam menemukan semua sampel positif yang sebenarnya. Sementara itu, F1-score merupakan rata-rata harmonis dari presisi dan recall,

3. HASIL DAN PEMBAHASAN

Penelitian ini bertujuan untuk mengimplementasikan algoritma K-Nearest Neighbor++ (KNN++) dalam klasifikasi serangan Distributed Denial of Service (DDoS) jenis UDP Flood secara berbasis anomali. Fokus utama penelitian ini adalah pada integrasi metode seleksi fitur Analysis of Variance (ANOVA) untuk meningkatkan efektivitas pemilihan fitur yang relevan, serta penggunaan Synthetic Minority Over-sampling Technique (SMOTE) sebagai teknik penyeimbang data untuk mengatasi ketidakseimbangan kelas dalam dataset. Model klasifikasi yang dihasilkan akan dievaluasi menggunakan beberapa metrik kinerja, yaitu akurasi, presisi, recall, dan F1-score. Dataset yang digunakan telah melalui tahapan preprocessing, termasuk pembersihan data, normalisasi, dan encoding, sebelum diterapkan ke dalam proses pelatihan dan pengujian model. Hasil analisis akan disajikan dalam bentuk tabel dan grafik untuk menunjukkan performa KNN++ dalam mendeteksi serangan DDoS berdasarkan kombinasi fitur yang telah diseleksi serta data yang telah seimbang. Penelitian ini diharapkan dapat memberikan kontribusi dalam pengembangan sistem deteksi dini serangan siber, khususnya pada lalu lintas jaringan yang menunjukkan pola anomali. Dengan menggabungkan metode seleksi fitur dan balancing data, model yang diusulkan berpotensi meningkatkan akurasi deteksi dibanding pendekatan konvensional. Selain itu, pemanfaatan KNN++ sebagai algoritma dasar memberikan pendekatan klasifikasi yang lebih efisien terhadap distribusi data yang kompleks. Implementasi model dilakukan pada dataset CIC-DDoS 2019 yang telah dikenal luas dalam penelitian keamanan jaringan.

3.1 Hasil Pengujian

Setelah model K-Nearest Neighbor++ (KNN++) diimplementasikan dengan dukungan seleksi fitur menggunakan ANOVA dan teknik penyeimbangan data SMOTE, proses pengujian dilakukan untuk mengevaluasi kinerja model dalam mengklasifikasikan serangan Distributed Denial of Service (DDoS) jenis UDP Flood. Evaluasi ini melibatkan perbandingan antara hasil klasifikasi model dengan label data aktual, menggunakan metrik evaluasi yang terdiri dari akurasi, presisi, recall, F1-score dan ROC-AUC. Penggunaan keempat metrik ini memberikan gambaran menyeluruh terhadap efektivitas model, di mana nilai akurasi yang tinggi

DOI: 10.33050/p8mgnt69 ISSN: **2356-5209**
pp. 103~112 *Online* ISSN: **2655-3058**

menunjukkan ketepatan keseluruhan, sedangkan nilai presisi dan recall yang tinggi mencerminkan kemampuan model dalam mengidentifikasi serangan secara tepat dan menyeluruh. F1-score sebagai harmonisasi dari presisi dan recall memberikan ukuran keseimbangan performa model, terutama saat menangani data yang tidak seimbang. Hasil evaluasi untuk setiap skenario pengujian baik sebelum maupun sesudah penerapan SMOTE—disajikan dalam bentuk tabel yang menunjukkan performa klasifikasi pada berbagai kombinasi fitur dan kondisi dataset. Tabel-tabel tersebut menjadi dasar untuk analisis lebih lanjut terhadap pengaruh seleksi fitur dan balancing data terhadap kemampuan model dalam mendeteksi serangan DDoS secara lebih akurat dan efisien. Pada Tabel 1 merupakan hasil evaluasi model tanpa menggunakan balancing data *SMOTE*

Tabel 1. Evaluasi Tanpa SMOTE

Metode	**Parameter**	**Akurasi**	**Presisi**	**Recall**	**F1-Score**	**Nilai ROC-AUC**	**Waktu Komputasi (detik)**
KNN++	K=3	99.994	99.997	99.997	99.997	99.54	1946
	K=5	99.992	99.994	99.997	99.996	99.08	1966
	K=7	99.993	99.994	99.998	99.996	99.08	1975
	K=9	99.988	99.989	99.998	99.994	98.17	1985
	K=11	99.988	99.989	99.998	99.994	98.17	1992
	K=13	99.988	99.988	100	99.994	97.94	1994

Berdasarkan hasil yang ditampilkan pada Tabel 1: Evaluasi Tanpa SMOTE, dapat disimpulkan bahwa algoritma KNN++ mampu memberikan hasil klasifikasi yang cukup baik dalam mendeteksi serangan DDoS jenis UDP Flood. Secara umum, nilai akurasi, presisi, recall, dan F1-score berada pada kisaran tinggi, yakni di atas 99.99%, dengan sedikit variasi antar parameter K. Di antara seluruh skenario, nilai K=3 menunjukkan performa yang paling seimbang, dengan akurasi sebesar 99.994%, presisi dan recall sebesar 99.997%, serta F1-score sebesar 99.997%. Selain memiliki nilai metrik yang tinggi, model dengan K=3 juga menghasilkan waktu komputasi paling singkat, yaitu 1946 detik, dibandingkan dengan parameter K lainnya. Nilai ROC-AUC pada K=3 juga menjadi yang tertinggi, yaitu 99.54, yang mengindikasikan kemampuan terbaik model dalam membedakan antara kelas serangan dan bukan serangan. Sebaliknya, pada nilai K yang lebih besar seperti K=13, meskipun recall mencapai 100%, ROC-AUC menurun hingga 97.94, yang menandakan penurunan kemampuan generalisasi model. Hal ini menunjukkan bahwa pemilihan parameter K yang tepat menjadi faktor penting dalam menjaga keseimbangan antara performa klasifikasi dan efisiensi komputasi. Oleh karena itu, hasil ini akan menjadi acuan untuk membandingkan performa model setelah diterapkannya teknik SMOTE guna melihat sejauh mana peningkatan dapat dicapai dengan penyeimbangan data. Selanjutnya pada Tabel 2 merupakan hasil evaluasi model dengan seleksi fitur anova yang dikombinasikan dengan SMOTE

Tabel 2. Evaluasi Dengan SMOTE

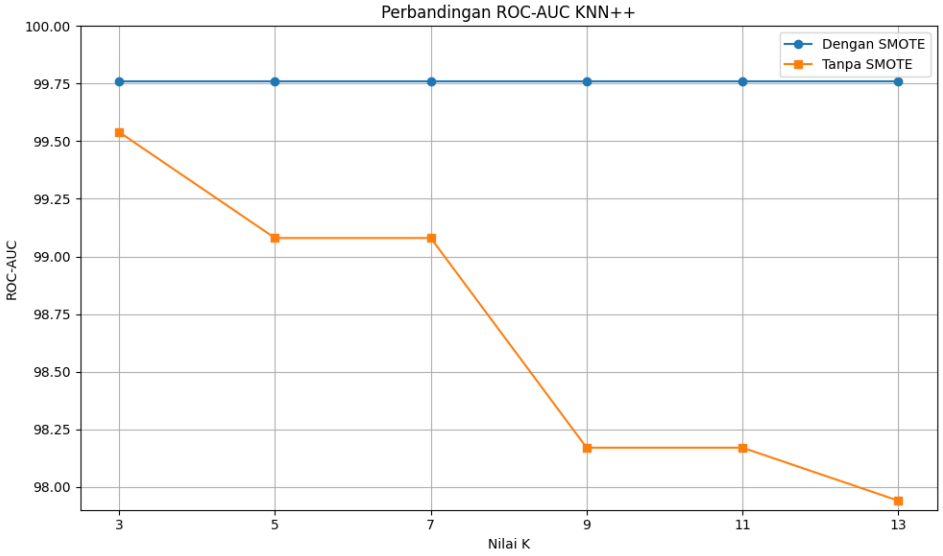
Metode	**Parameter**	**Akurasi**	**Presisi**	**Recall**	**F1-Score**	**Nilai ROC-AUC**	**Waktu Komputasi (detik)**

DOI: 10.33050/p8mgnt69 ISSN: **2356-5209**
pp. 103~112 *Online* ISSN: **2655-3058**

Metode	Parameter	Akurasi	Presisi	Recall	F1-Score	Nilai ROC-AUC	Waktu Komputasi (detik)
KNN++	K=3	99.994	99.998	99.996	99.997	99.76	3846
	K=5	99.994	99.998	99.996	99.997	99.76	3858
	K=7	99.988	99.998	99.989	99.994	99.76	3876
	K=9	99.988	99.998	99.989	99.994	99.76	3896
	K=11	99.987	99.998	99.988	99.993	99.76	3889
	K=13	99.983	99.987	99.984	99.991	99.76	3891

Berdasarkan hasil yang ditampilkan pada Tabel 2: Evaluasi Dengan SMOTE, dapat dilihat bahwa penerapan teknik penyeimbangan data SMOTE memberikan dampak positif terhadap kestabilan performa model KNN++ dalam mendeteksi serangan DDoS jenis UDP Flood. Seluruh metrik evaluasi, seperti akurasi, presisi, recall, dan F1-score, tetap berada pada rentang yang sangat tinggi, yaitu antara 99.983% hingga 99.994%, dengan perbedaan yang sangat kecil antar variasi nilai K. Nilai ROC-AUC juga menunjukkan peningkatan konsistensi, berada di angka 99.76 untuk sebagian besar nilai K, yang mengindikasikan kemampuan model yang stabil dalam membedakan antara kelas serangan dan bukan serangan setelah distribusi data diseimbangkan.

Sama seperti pada pengujian tanpa SMOTE, nilai K=3 kembali menunjukkan performa yang unggul dengan akurasi dan F1-score tertinggi (99.994% dan 99.997%), serta waktu komputasi yang lebih rendah dibandingkan dengan nilai K lainnya (3846 detik). Hal ini memperkuat temuan sebelumnya bahwa nilai K yang lebih kecil tidak hanya menghasilkan kinerja klasifikasi yang baik, tetapi juga lebih efisien secara komputasi. Meski demikian, waktu komputasi secara keseluruhan meningkat secara signifikan dibandingkan pengujian tanpa SMOTE, akibat dari tambahan proses sintesis data minoritas. Secara umum, penerapan SMOTE berhasil meningkatkan stabilitas dan generalisasi model, meskipun dengan konsekuensi meningkatnya beban komputasi. Pada Gambar 2 merupakan visualisasi nilai ROC-AUC dengan SMOTE dan tanpa menggunakan SMOTE.



Gambar 2. Visualisasi ROC-AUC

DOI: 10.33050/p8mgnt69 ISSN: **2356-5209**
pp. 103~112 *Online* ISSN: **2655-3058**

Grafik di atas menunjukkan perbandingan nilai ROC-AUC algoritma KNN++ dengan dan tanpa penerapan metode SMOTE pada berbagai nilai K. ROC-AUC merupakan metrik evaluasi yang menggambarkan kemampuan model dalam membedakan antara kelas positif dan negatif, di mana nilai yang mendekati 100% menunjukkan performa yang sangat baik. Berdasarkan visualisasi, terlihat bahwa penerapan SMOTE menghasilkan performa yang sangat stabil dengan nilai ROC-AUC konstan di sekitar 99,76% pada seluruh variasi nilai K (3, 5, 7, 9, 11, dan 13). Hal ini menunjukkan bahwa SMOTE mampu mengatasi ketidakseimbangan kelas pada data pelatihan, sehingga model KNN++ tetap efektif dalam melakukan klasifikasi.

Sebaliknya, model KNN++ tanpa SMOTE menunjukkan penurunan performa yang cukup signifikan seiring dengan meningkatnya nilai K. Nilai ROC-AUC menurun dari sekitar 99,54% pada K=3 menjadi 97,94% pada K=13. Penurunan ini mengindikasikan bahwa tanpa penanganan ketidakseimbangan data, model menjadi kurang stabil dan lebih sensitif terhadap perubahan parameter K. Dengan demikian, dapat disimpulkan bahwa penggunaan SMOTE tidak hanya meningkatkan performa KNN++ secara keseluruhan, tetapi juga membantu menjaga kestabilan model terhadap variasi parameter.

5. KESIMPULAN

Berdasarkan hasil evaluasi yang telah dilakukan, implementasi algoritma KNN++ dengan seleksi fitur ANOVA menunjukkan performa klasifikasi yang sangat tinggi dalam mendeteksi serangan DDoS jenis UDP Flood, baik tanpa maupun dengan penerapan teknik SMOTE. Penggunaan SMOTE terbukti meningkatkan stabilitas model terhadap variasi parameter K dan menghasilkan nilai ROC-AUC yang lebih konsisten, meskipun berdampak pada peningkatan waktu komputasi. Model dengan nilai K=3 memberikan hasil paling optimal pada kedua skenario, dengan kombinasi performa tinggi dan efisiensi waktu terbaik. Kelebihan dari pendekatan ini terletak pada akurasi tinggi dan ketahanan terhadap ketidakseimbangan data, sementara kekurangannya adalah kebutuhan komputasi yang cukup tinggi, terutama setelah penyeimbangan data. Penelitian ini membuktikan bahwa kombinasi KNN++, ANOVA, dan SMOTE dapat menjadi pendekatan efektif dalam sistem deteksi serangan berbasis anomali.

6. SARAN

Berdasarkan hasil penelitian dan evaluasi yang telah dilakukan, terdapat beberapa saran yang dapat dipertimbangkan untuk pengembangan penelitian selanjutnya. Pertama, mengingat waktu komputasi yang cenderung tinggi terutama setelah penerapan SMOTE, disarankan untuk mengeksplorasi metode balancing data yang lebih efisien. Kedua, untuk meningkatkan kemampuan generalisasi model, penelitian selanjutnya dapat mempertimbangkan penggunaan algoritma ensembel seperti Random Forest atau Gradient Boosting yang dikombinasikan dengan seleksi fitur ANOVA. Selain itu, evaluasi dapat diperluas dengan menggunakan data serangan yang lebih beragam atau real-time untuk menguji ketahanan model terhadap variasi pola serangan. Terakhir, pendekatan seleksi fitur yang adaptif atau berbasis pembelajaran seperti Recursive Feature Elimination (RFE) juga dapat dijadikan alternatif untuk dibandingkan dengan ANOVA dalam meningkatkan efektivitas klasifikasi.

DOI: 10.33050/p8mgnt69 ISSN: **2356-5209**
pp. 103~112 *Online* ISSN: **2655-3058**

DAFTAR PUSTAKA

[1] Qrator Labs, "Q3 2024 DDoS, Bots and BGP Incidents Statistics and Overview," Mar. 2024.
[2] SOC Radar, "Indonesia Threat Landscape Report," 2024.
[3] R. Taguelmimt and R. Beghdad, "DS-kNN: An intrusion detection system based on a distance sum-based K-nearest neighbors," *International Journal of Information Security and Privacy*, vol. 15, no. 2, pp. 131–144, Apr. 2021, doi: 10.4018/IJISP.2021040107.
[4] R. Wazirali, "An Improved Intrusion Detection System Based on KNN Hyperparameter Tuning and Cross-Validation," *Arab J Sci Eng*, vol. 45, no. 12, pp. 10859–10873, Dec. 2020, doi: 10.1007/s13369-020-04907-7.
[5] V. Immanuel Sunarko *et al.*, "Implementasi K-Fold Dalam Prediksi Hasil Produksi Agrikultur Pada Algoritma K-Nearest Neighbor (KNN) 10 Implementasi K-Fold Dalam Prediksi Hasil Produksi Agrikultur Pada Algoritma K-Nearest Neighbor (KNN)." [Online]. Available: https://satudata.pertanian.go.id/
[6] S. Zhang, X. Li, M. Zong, X. Zhu, and R. Wang, "Efficient kNN classification with different numbers of nearest neighbors," *IEEE Trans Neural Netw Learn Syst*, vol. 29, no. 5, pp. 1774–1785, May 2018, doi: 10.1109/TNNLS.2017.2673241.
[7] Y. Song, X. Kong, and C. Zhang, "A Large-Scale k -Nearest Neighbor Classification Algorithm Based on Neighbor Relationship Preservation," *Wirel Commun Mob Comput*, vol. 2022, 2022, doi: 10.1155/2022/7409171.
[8] S. K. Lakshminarayana and P. I. Basarkod, "Unification of K-Nearest Neighbor (KNN) with Distance Aware Algorithm for Intrusion Detection in Evolving Networks Like IoT," *Wirel Pers Commun*, vol. 132, no. 3, pp. 2255–2281, Oct. 2023, doi: 10.1007/s11277-023-10722-8.
[9] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," *Computers and Electrical Engineering*, vol. 40, no. 1, pp. 16–28, Jan. 2014, doi: 10.1016/j.compeleceng.2013.11.024.
[10] N. Japkowicz and S. Stephen, "The class imbalance problem: A systematic study 1," IOS Press, 2002.
[11] N. V Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," 2002.
[12] S. Walling and S. Lodh, "Performance Evaluation of Supervised Machine Learning Based Intrusion Detection with Univariate Feature Selection on NSL KDD Dataset," Feb. 03, 2023. doi: 10.21203/rs.3.rs-2537820/v1.
[13] S. S. Shafin, S. A. Prottoy, S. Abbas, S. Bin Hakim, A. Chowdhury, and M. M. Rashid, "Distributed Denial of Service Attack Detection Using Machine Learning and Class Oversampling," in *Communications in Computer and Information Science*, Springer Science and Business Media Deutschland GmbH, 2021, pp. 247–259. doi: 10.1007/978-3-030-82269-9_19.
[14] S. Suriya and J. Joanish Muthu, "Type 2 Diabetes Prediction using K-Nearest Neighbor Algorithm," *Journal of Trends in Computer Science and Smart Technology*, vol. 5, no. 2, pp. 190–205, Jun. 2023, doi: 10.36548/jtcsst.2023.2.007.