JuSiTiK
Jurnal Sistem dan Teknologi Informasi Komunikasi
P-ISSN 2579-4116 | E-ISSN 2579-5570 | Vol. 9 No. 2 – Juli 2026

Deteksi Malware IoT Menggunakan K-Nearest Neighbor pada Dataset IoT-23

Muhammad Arif Ferdiansyah[1], Adi Wibowo[2]
[1-2]Program studi Sistem dan Teknologi Informasi, Universitas Muhammadiyah Kotabumi
[1-2]Jl. Hasan Kepala Ratu No. 1052 Sindangsari, Kotabumi, Lampung Utara
Email: muhammadarif.umko@yahoo.com[1], adi.wibowo@umko.ac.id[2]

Abstract

The development of the Internet of Things (IoT) has increased the number of interconnected devices across various sectors; however, it also introduces security challenges due to limited resources and weak protection mechanisms. These conditions make IoT devices vulnerable to cyberattacks such as malware, botnets, and Distributed Denial of Service (DDoS). Therefore, an effective and efficient intrusion detection system is required. This study aims to analyze the characteristics of malware traffic and evaluate the performance of the K-Nearest Neighbor (KNN) algorithm in detecting malicious traffic using the IoT-23 dataset based on network flow. The method employed is supervised machine learning with preprocessing stages including data cleaning, label transformation, feature encoding, and normalization using MinMaxScaler. The dataset is classified into two classes, namely benign and malicious. The experimental results show that KNN is able to achieve an accuracy of up to 97%, with optimal performance at K = 5, which provides the best balance between precision, recall, and F1-score. The ROC-AUC value of 0.97 indicates that the model has a very good capability in distinguishing between normal and malicious traffic. These results indicate that KNN based on network flow is effective and has the potential to be implemented in intrusion detection systems in IoT environments.

Abstrak

Perkembangan Internet of Things (IoT) meningkatkan jumlah perangkat terhubung di berbagai sektor, namun juga menimbulkan tantangan keamanan akibat keterbatasan sumber daya dan lemahnya mekanisme proteksi. Kondisi ini membuat perangkat IoT rentan terhadap serangan siber seperti malware, botnet, dan Distributed Denial of Service (DDoS). Oleh karena itu, diperlukan sistem deteksi intrusi yang efektif dan efisien. Penelitian ini bertujuan menganalisis karakteristik traffic malware serta mengevaluasi kinerja algoritma K-Nearest Neighbor (KNN) dalam mendeteksi traffic berbahaya menggunakan dataset IoT-23 berbasis network flow. Metode yang digunakan adalah supervised machine learning dengan tahapan preprocessing meliputi data cleaning, transformasi label, encoding fitur, dan normalisasi menggunakan MinMaxScaler. Dataset diklasifikasikan menjadi dua kelas, yaitu benign dan malicious. Hasil eksperimen menunjukkan bahwa KNN mampu mencapai akurasi hingga 97%, dengan performa optimal pada K = 5 yang memberikan keseimbangan terbaik antara precision, recall, dan F1-score. Nilai ROC-AUC sebesar 0,97 menunjukkan kemampuan model yang sangat baik dalam membedakan traffic normal dan berbahaya. Hasil ini menunjukkan bahwa KNN berbasis network flow efektif dan berpotensi diterapkan pada sistem deteksi intrusi di lingkungan IoT.

Kata Kunci: deteksi intrusi, keamanan iot, KNN, malware, network flow.

Pendahuluan

Perkembangan Internet of Things (IoT) dalam beberapa tahun terakhir menunjukkan tren peningkatan yang sangat signifikan dan berperan penting dalam mendorong transformasi digital di berbagai sektor. Konsep IoT memungkinkan berbagai perangkat fisik untuk terhubung melalui jaringan internet sehingga dapat saling berkomunikasi, bertukar informasi, serta menjalankan proses otomatisasi secara real-time. Secara umum, sistem IoT terdiri dari sensor, aktuator, serta perangkat embedded yang saling terhubung melalui teknologi komunikasi seperti Wi-Fi, ZigBee, dan LoRa. Penerapan IoT telah meluas ke berbagai bidang, antara lain smart home, layanan kesehatan, industri, dan pertanian cerdas, yang berkontribusi dalam meningkatkan efisiensi operasional dan kualitas layanan berbasis teknologi [1].

Namun, peningkatan jumlah perangkat IoT yang terhubung juga diikuti oleh bertambahnya kompleksitas sistem jaringan. Lingkungan IoT yang bersifat heterogen, dengan perbedaan kemampuan perangkat dan standar keamanan, menimbulkan berbagai tantangan khususnya dalam aspek keamanan jaringan. Banyak perangkat IoT dikembangkan dengan keterbatasan sumber daya, seperti kapasitas memori yang terbatas, kemampuan komputasi yang rendah, serta kebutuhan konsumsi energi yang

*) Corresponding Author: Adi Wibowo

efisien. Kondisi ini menyebabkan penerapan mekanisme keamanan seperti enkripsi, autentikasi, dan pembaruan sistem tidak dapat dilakukan secara optimal. Akibatnya, perangkat IoT menjadi lebih rentan terhadap berbagai jenis serangan siber, termasuk malware, botnet, dan Distributed Denial of Service (DDoS) [2],[3].

Penelitian yang dilakukan oleh [4]. menunjukkan bahwa tingginya tingkat kerentanan pada ekosistem IoT dipengaruhi oleh kurangnya standarisasi keamanan serta keberagaman perangkat yang digunakan. Selain itu, masih banyak perangkat yang menggunakan kredensial bawaan (default) dan tidak dilengkapi dengan mekanisme pembaruan keamanan yang memadai, sehingga membuka peluang bagi penyerang untuk mengeksploitasi sistem. Serangan pada jaringan IoT tidak hanya berdampak pada satu perangkat, tetapi juga dapat dimanfaatkan untuk melancarkan serangan dalam skala besar. Salah satu kasus yang cukup dikenal adalah botnet Mirai yang berhasil menginfeksi banyak perangkat IoT dan digunakan untuk melakukan serangan DDoS secara masif hingga mengganggu layanan internet secara global [5].

Deteksi serangan pada jaringan IoT menjadi tantangan tersendiri karena pola lalu lintas yang dihasilkan sering kali menyerupai aktivitas normal. Hal ini menyebabkan metode konvensional berbasis signature kurang efektif dalam mengidentifikasi serangan baru atau varian malware yang belum terdaftar. Oleh karena itu, diperlukan pendekatan yang lebih adaptif dalam mendeteksi anomali pada lalu lintas jaringan IoT [6]. Pendekatan berbasis machine learning mulai banyak diterapkan karena mampu mempelajari pola komunikasi jaringan dari data historis dan melakukan identifikasi anomali secara otomatis. Beberapa penelitian menunjukkan bahwa metode ini dapat meningkatkan tingkat akurasi deteksi secara signifikan. [4] melaporkan bahwa penggunaan machine learning mampu meningkatkan performa deteksi serangan pada jaringan IoT, sementara [2] menyatakan bahwa analisis berbasis network flow efektif dalam membedakan traffic normal dan malicious.

Salah satu algoritma machine learning yang cukup populer adalah K-Nearest Neighbor (KNN), yang melakukan klasifikasi berdasarkan kedekatan jarak antar data dalam ruang fitur. Algoritma ini menentukan kelas suatu data baru berdasarkan mayoritas kelas dari sejumlah tetangga terdekat. KNN memiliki keunggulan berupa implementasi yang sederhana serta kemampuan dalam menangani pola data yang tidak linear tanpa memerlukan asumsi distribusi tertentu [7]. Penelitian oleh Verma dan Ranga juga menunjukkan bahwa KNN memiliki kinerja yang baik dalam mendeteksi anomali pada jaringan IoT berbasis network flow [8].

Dalam penelitian ini digunakan dataset IoT-23, yaitu dataset publik berbasis network flow yang telah dilengkapi dengan label. Dataset ini mencakup berbagai skenario aktivitas jaringan, baik kondisi normal maupun serangan malware, sehingga sesuai untuk pendekatan supervised learning. Penggunaan dataset ini memungkinkan analisis yang lebih representatif terhadap kondisi nyata jaringan IoT.

Berdasarkan latar belakang tersebut, penelitian ini bertujuan untuk menganalisis karakteristik lalu lintas malware serta mengevaluasi kinerja algoritma K-Nearest Neighbor (KNN) dalam mendeteksi traffic berbahaya pada jaringan IoT menggunakan dataset IoT-23. Hasil penelitian ini diharapkan dapat memberikan kontribusi dalam pengembangan sistem deteksi intrusi berbasis machine learning yang lebih adaptif dan akurat.

Meskipun berbagai penelitian sebelumnya telah mengkaji deteksi serangan pada jaringan IoT menggunakan pendekatan machine learning, sebagian besar penelitian masih berfokus pada penggunaan algoritma kompleks seperti deep learning atau ensemble learning yang memiliki kebutuhan komputasi tinggi. Hal ini menjadi kurang optimal untuk lingkungan IoT yang memiliki keterbatasan sumber daya.

Oleh karena itu, penelitian ini menawarkan pendekatan yang lebih ringan dengan menggunakan algoritma K-Nearest Neighbor (KNN) berbasis network flow pada dataset IoT-23. Kontribusi utama penelitian ini meliputi:

1) analisis karakteristik traffic malware berbasis flow pada dataset IoT-23,
2) evaluasi performa KNN pada berbagai nilai parameter K untuk mengidentifikasi konfigurasi optimal, serta

3) penyajian pendekatan deteksi yang sederhana namun tetap memiliki performa tinggi dan berpotensi diterapkan pada sistem IoT dengan keterbatasan sumber daya.

Dengan demikian, penelitian ini diharapkan dapat memberikan solusi deteksi intrusi yang efisien, ringan, dan tetap akurat untuk lingkungan IoT.

Metode Penelitian

Tahapan penelitian diawali dengan proses pengambilan dataset IoT-23 yang memuat berbagai variasi lalu lintas jaringan dari perangkat IoT, baik dalam kondisi normal (benign) maupun saat terjadi serangan (malicious). Dataset ini banyak dimanfaatkan dalam penelitian keamanan jaringan karena mampu merepresentasikan karakteristik nyata lalu lintas IoT berbasis network flow [7].

Selanjutnya dilakukan tahap preprocessing untuk meningkatkan kualitas data sebelum digunakan dalam proses pemodelan. Tahapan ini mencakup pembersihan data (data cleaning), perubahan label (label transformation), pengkodean fitur kategorikal (feature encoding), serta proses normalisasi menggunakan metode MinMaxScaler guna menyamakan rentang nilai antar fitur [8].

Setelah data melalui tahap preprocessing, dataset dibagi menjadi dua bagian, yaitu data latih sebesar 80% dan data uji sebesar 20%. Pembagian ini bertujuan untuk melatih model sekaligus menguji kemampuan generalisasi terhadap data baru yang belum pernah digunakan sebelumnya [9]. Data latih digunakan dalam proses pembentukan model klasifikasi menggunakan algoritma K-Nearest Neighbor (KNN), sedangkan data uji digunakan untuk mengukur kinerja model dalam mengklasifikasikan lalu lintas jaringan.

Tahap akhir penelitian adalah evaluasi performa model dengan menggunakan beberapa metrik klasifikasi, yaitu accuracy, precision, recall, F1-score, confusion matrix, serta ROC-AUC. Kombinasi metrik tersebut digunakan untuk memberikan gambaran yang komprehensif terhadap kemampuan model dalam mendeteksi aktivitas normal maupun serangan pada jaringan IoT [10]. Gambar 1 menunjukkan alur metodologi penelitian yang digunakan dalam studi ini.

Gambar 1. Flowchart Metodologi Penelitian

A. Dataset Penelitian

Dataset yang digunakan dalam penelitian ini adalah IoT-23, yaitu dataset berbasis network flow yang memuat berbagai fitur statistik komunikasi jaringan, seperti durasi koneksi, jumlah paket, volume data (byte), serta status koneksi. Pendekatan berbasis flow ini memungkinkan proses analisis dilakukan secara lebih efisien dibandingkan dengan metode inspeksi paket secara langsung (packet inspection). Oleh karena itu, representasi ini banyak digunakan dalam pengembangan Intrusion Detection System (IDS) berbasis machine learning [11],[12].

B. Preprocessing Data

a. Normalisasi Data

Proses normalisasi dilakukan menggunakan metode *Min-Max Scaling* dengan tujuan menyamakan rentang nilai antar fitur agar berada pada skala yang seragam. Hal ini penting untuk menghindari dominasi fitur tertentu dalam perhitungan jarak. Persamaan yang digunakan dalam proses normalisasi ditunjukkan sebagai berikut:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \quad \text{.....(1)}$$

dengan keterangan:
x = nilai asli dari fitur
x_min = nilai minimum pada fitur
x_max = nilai maksimum pada fitur
x' = nilai hasil normalisasi

Tahap normalisasi menjadi krusial karena algoritma K-Nearest Neighbor (KNN) bergantung pada perhitungan jarak, sehingga perbedaan skala data dapat mempengaruhi hasil klasifikasi.

b. Transformasi Label

Pada tahap ini, label yang awalnya terdiri dari beberapa kelas disederhanakan menjadi dua kategori (binary classification) untuk mempermudah proses klasifikasi serta meningkatkan kestabilan model [13]. Transformasi label dilakukan dengan mengelompokkan data ke dalam dua kelas utama sebagai berikut:

$$y = \begin{cases} 0, & \text{benign} \\ 1, & \text{malicious} \end{cases} \quad \text{.....(2)}$$

0, untuk data benign
1, untuk data malicious

C. Pembagian Data

Dataset pada penelitian ini dibagi menjadi dua bagian, yaitu data latih sebesar 80% dan data uji sebesar 20% dengan menggunakan metode *stratified split*. Pemilihan metode ini bertujuan untuk menjaga proporsi distribusi kelas antara data benign dan malicious tetap seimbang pada kedua subset, sehingga proses evaluasi model dapat merepresentasikan kondisi data secara lebih akurat.

Pendekatan *stratified split* menjadi penting dalam klasifikasi data jaringan karena distribusi kelas yang tidak seimbang dapat mempengaruhi hasil pembelajaran model. Jika pembagian data dilakukan secara acak tanpa mempertimbangkan komposisi kelas, maka terdapat kemungkinan terjadinya dominasi salah satu kelas pada data latih maupun data uji, yang berpotensi menurunkan kualitas evaluasi.

Dengan mempertahankan distribusi kelas yang proporsional, model dapat dilatih secara lebih optimal dan terhindar dari kecenderungan bias terhadap kelas tertentu. Selain itu, kesesuaian karakteristik antara data latih dan data uji memungkinkan proses pengujian generalisasi model dilakukan secara lebih objektif.

Melalui pendekatan ini, algoritma K-Nearest Neighbor (KNN) diharapkan mampu mengenali pola data secara efektif pada tahap pelatihan serta menghasilkan performa yang stabil ketika dihadapkan pada data baru yang belum pernah digunakan sebelumnya.

D. Algoritma K-Nearest Neighbor (KNN)

K-Nearest Neighbor (KNN) merupakan salah satu algoritma klasifikasi yang bekerja berdasarkan prinsip kedekatan antar data dalam ruang fitur. Proses penentuan kelas pada data uji dilakukan dengan membandingkan jaraknya terhadap sejumlah data latih, kemudian memilih kelas mayoritas dari tetangga terdekat.

a. Jarak Euclidean

Untuk mengukur tingkat kedekatan antar data, digunakan perhitungan jarak Euclidean. Metode ini menghitung jarak antara dua titik dalam ruang fitur dengan mempertimbangkan selisih setiap atribut yang dimiliki.

$$d(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \quad \text{.....(3)}$$

di mana:

$\boldsymbol{x_i}$= fitur ke-i data uji
$\boldsymbol{y_i}$= fitur ke-i data latih
$\boldsymbol{n}$ =jumlah fitur

b. Proses Klasifikasi

KNN menentukan kelas berdasarkan mayoritas tetangga terdekat:

$$\hat{y} = \text{mode}(y_1, y_2, \dots, y_k) \quad (4)$$

di mana:
k =jumlah tetangga terdekat
$\hat{y}$=hasil prediksi

c. Variasi Parameter K

Nilai K yang digunakan dalam penelitian:

$$K = \{3,5,7,9\} \quad (5)$$

Pemilihan nilai K bertujuan untuk menganalisis pengaruh bias dan variansi terhadap performa model.

E. Evaluasi Model

Evaluasi kinerja model dilakukan dengan menggunakan beberapa metrik klasifikasi yang umum diterapkan dalam penelitian machine learning dan sistem Intrusion Detection System (IDS), yaitu accuracy, precision, recall, dan F1-score [14]. Metrik-metrik tersebut digunakan untuk memberikan gambaran yang lebih menyeluruh terhadap kemampuan model dalam melakukan klasifikasi.

Untuk menilai performa model secara lebih komprehensif, digunakan beberapa parameter evaluasi yang mencerminkan tingkat ketepatan, keakuratan, serta keseimbangan hasil klasifikasi antara kelas positif dan negatif.

a. Accuracy

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (6)$$

b. Precision

$$Precision = \frac{TP}{TP+FP} \quad (7)$$

c. Recall

$$Recall = \frac{TP}{TP+FN} \quad (8)$$

d. F1-Score

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (9)$$

b. ROC dan AUC

Kurva ROC (*Receiver Operating Characteristic*) digunakan untuk menganalisis kinerja model dalam membedakan dua kelas dengan melihat hubungan antara *True Positive Rate* (TPR) dan *False Positive Rate* (FPR) pada berbagai nilai ambang batas (threshold). TPR atau *sensitivity* menunjukkan proporsi data positif yang berhasil diklasifikasikan dengan benar, sedangkan FPR menunjukkan tingkat kesalahan model dalam mengklasifikasikan data negatif sebagai positif. Perhitungan kedua metrik tersebut dinyatakan sebagai berikut:

$$TPR = \frac{TP}{TP+FN} \quad (10)$$

$$FPR = \frac{FP}{FP+TN} \quad (11)$$

Selain itu, nilai AUC (*Area Under Curve*) digunakan sebagai indikator untuk mengukur kemampuan model dalam membedakan antara kelas positif dan negatif secara keseluruhan. Semakin tinggi nilai AUC, maka semakin baik kemampuan model dalam melakukan klasifikasi.

F. Validitas Eksperimen

Untuk menjaga validitas dan konsistensi hasil penelitian, beberapa langkah pengendalian eksperimen diterapkan secara sistematis. Pertama, penggunaan random state yang tetap dilakukan pada proses pembagian dataset guna memastikan bahwa hasil eksperimen dapat direproduksi dengan kondisi yang sama.

Kedua, Dalam penelitian ini, seluruh skenario pengujian menggunakan dataset yang sama sehingga perbedaan hasil yang diperoleh sepenuhnya disebabkan oleh variasi parameter model, bukan karena perubahan pada data yang digunakan. Dengan demikian, pengaruh konfigurasi model dapat dianalisis secara lebih objektif.

Selain itu, proses evaluasi dilakukan menggunakan data uji (*testing data*) yang tidak dilibatkan dalam tahap pelatihan. Pendekatan ini bertujuan untuk memastikan bahwa hasil evaluasi benar-benar mencerminkan kemampuan model dalam melakukan generalisasi terhadap data baru yang belum pernah dikenali sebelumnya.

Selanjutnya, perhatian juga diberikan pada keseimbangan distribusi kelas antara data benign dan malicious untuk menghindari bias model terhadap kelas tertentu. Distribusi kelas yang relatif seimbang memungkinkan model untuk belajar secara proporsional dan menghasilkan performa klasifikasi yang lebih akurat. Dengan menerapkan langkah-langkah tersebut, penelitian ini berupaya memastikan bahwa hasil yang diperoleh bersifat objektif, dapat dipercaya, serta memiliki tingkat validitas yang tinggi dalam mengukur performa algoritma K-Nearest Neighbor pada deteksi traffic malware dalam jaringan IoT.

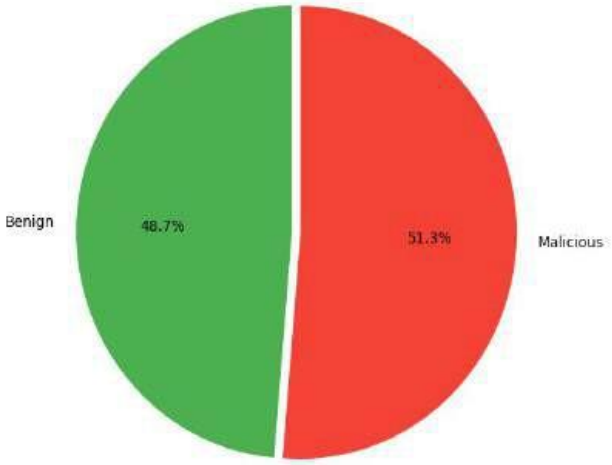
Hasil dan Pembahasan

A. Deskripsi Dataset dan Hasil Preprocessing

IoT-23, yaitu dataset publik yang berisi berbagai skenario lalu lintas jaringan dari perangkat Internet of Things (IoT), yang mencakup aktivitas normal (*benign*) maupun aktivitas serangan malware (*malicious*). Dataset ini berbasis *network flow* yang merepresentasikan komunikasi jaringan dalam bentuk statistik seperti durasi koneksi, jumlah paket, dan volume data.

Setelah proses preprocessing dilakukan, label dataset dikonversi menjadi dua kelas utama, yaitu:

- Benign (0) : traffic jaringan normal
- Malicious (1) : traffic jaringan yang mengandung aktivitas serangan



Gambar 2. Distribusi Kelas Dataset IoT-23

Gambar 2 menunjukkan distribusi kelas setelah proses transformasi label dilakukan. Berdasarkan hasil tersebut, dataset terdiri dari:

- 3205 data benign
- 3379 data malicious

Distribusi ini menunjukkan bahwa jumlah data malicious sedikit lebih besar dibandingkan benign, dengan persentase sekitar 51,3% malicious dan 48,7% benign. Distribusi yang relatif seimbang ini sangat penting dalam pelatihan model machine learning karena dapat mengurangi bias model terhadap

salah satu kelas dan meningkatkan validitas hasil klasifikasi.

B. Implementasi Algoritma K-Nearest Neighbor

Pada penelitian ini, proses klasifikasi lalu lintas jaringan dilakukan menggunakan algoritma K-Nearest Neighbor (KNN). Metode ini menentukan kelas suatu data dengan mempertimbangkan kedekatannya terhadap sejumlah data terdekat yang terdapat pada data latih. Pengukuran kedekatan antar data dilakukan menggunakan Euclidean Distance. Dengan pendekatan ini, setiap data uji akan diklasifikasikan berdasarkan kelas mayoritas dari tetangga terdekat yang memiliki jarak paling kecil dalam ruang fitur.

Untuk mengetahui pengaruh parameter terhadap kinerja model, dilakukan pengujian dengan beberapa nilai K, yaitu K = 3, K = 5, dan K = 7. Variasi nilai K digunakan untuk mengamati keseimbangan antara sensitivitas terhadap pola lokal dan kemampuan generalisasi model. Nilai K yang lebih kecil cenderung lebih responsif terhadap pola spesifik pada data, sedangkan nilai K yang lebih besar memberikan hasil yang lebih stabil terhadap variasi data. Pemilihan algoritma KNN dalam penelitian ini didasarkan pada kesesuaiannya dengan karakteristik data jaringan IoT. KNN memiliki keunggulan dari sisi implementasi yang sederhana karena tidak memerlukan proses pelatihan model yang kompleks. Proses klasifikasi dilakukan secara langsung dengan membandingkan data uji terhadap data latih yang tersedia.

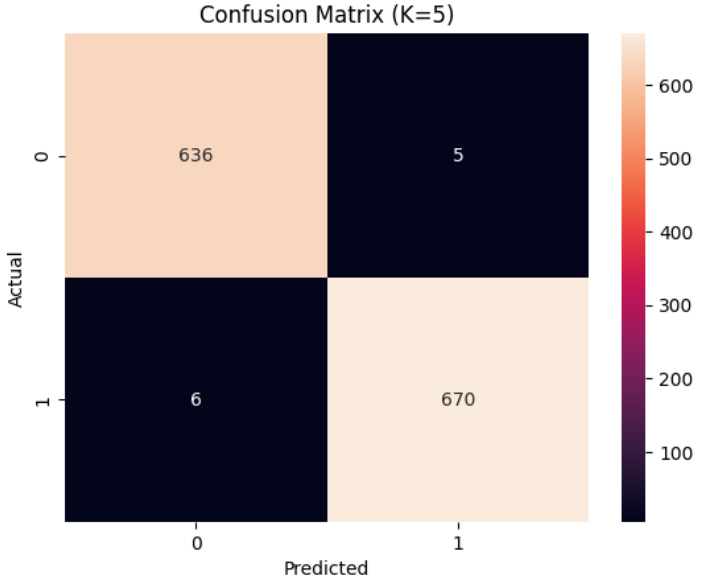
Selain itu, KNN termasuk dalam metode non-parametrik yang tidak bergantung pada asumsi distribusi data tertentu, sehingga mampu menangani pola data yang kompleks dan tidak linear. Fleksibilitas dalam menentukan nilai K juga menjadi kelebihan tersendiri karena memungkinkan penyesuaian performa model sesuai kebutuhan. Dengan karakteristik tersebut, KNN dinilai efektif untuk diterapkan dalam sistem deteksi intrusi berbasis machine learning, khususnya dalam membedakan antara lalu lintas jaringan normal (benign) dan aktivitas berbahaya (malicious).

C. Evaluasi Model

Evaluasi kinerja model dilakukan dengan memanfaatkan beberapa metrik klasifikasi, yaitu accuracy, precision, recall, dan F1-score. Metrik-metrik tersebut digunakan untuk menilai tingkat ketepatan serta keseimbangan hasil prediksi yang dihasilkan oleh model. Selain itu, analisis performa juga didukung dengan penggunaan confusion matrix dan kurva ROC (*Receiver Operating Characteristic*) untuk memberikan gambaran evaluasi yang lebih menyeluruh.

a. Confusion Matrix

Confusion matrix digunakan sebagai alat untuk menganalisis hasil klasifikasi dengan membandingkan antara nilai prediksi dan kondisi sebenarnya pada data uji. Melalui matriks ini, dapat diketahui jumlah prediksi yang benar maupun yang salah, sehingga memberikan informasi yang lebih rinci terkait kinerja model dalam mengklasifikasikan data.



Gambar 3. Confusion Matrix Model KNN

Berdasarkan Gambar 3, confusion matrix memperlihatkan empat kategori utama dalam hasil klasifikasi, yaitu:

- True Positive (TP): data malicious yang berhasil diprediksi dengan benar
- True Negative (TN): data benign yang terklasifikasi dengan tepat
- False Positive (FP): data benign yang keliru diklasifikasikan sebagai malicious
- False Negative (FN): data malicious yang tidak terdeteksi dan diklasifikasikan sebagai benign

Hasil tersebut menunjukkan bahwa mayoritas data berhasil diklasifikasikan secara tepat oleh model KNN, dengan jumlah kesalahan prediksi yang relatif rendah. Hal ini mengindikasikan bahwa model memiliki kemampuan yang baik dalam membedakan antara lalu lintas jaringan normal dan aktivitas serangan.

b. **Performa Model KNN**

Hasil pengujian model berdasarkan metrik klasifikasi disajikan pada Tabel 1, yang menampilkan nilai accuracy, precision, recall, dan F1-score untuk setiap variasi parameter K yang digunakan..

Tabel 1. Hasil Evaluasi Model KNN

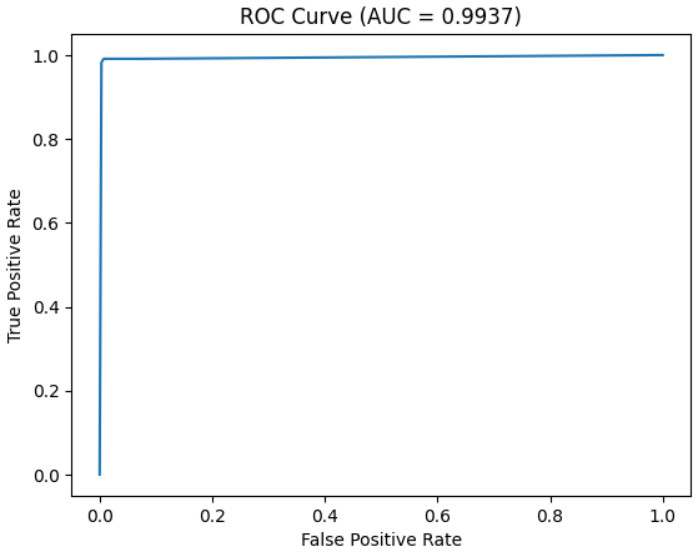
K	Acc	Prec	Rec	F1
3	0.96	0.95	0.94	0.95
5	0.97	0.96	0.96	0.96
7	0.96	0.95	0.95	0.95

Berdasarkan hasil tersebut, konfigurasi dengan K = 5 menunjukkan kinerja terbaik dengan nilai akurasi tertinggi sebesar 97%. Hal ini mengindikasikan bahwa pemilihan nilai K yang tepat berpengaruh signifikan terhadap kemampuan model dalam melakukan klasifikasi secara optimal.

Tingginya nilai performa yang diperoleh juga menunjukkan bahwa fitur-fitur yang dihasilkan dari representasi network flow mampu menggambarkan pola lalu lintas jaringan dengan baik, sehingga memudahkan model dalam membedakan antara aktivitas normal dan serangan.

c. **Analisis ROC Curve**

Selain menggunakan confusion matrix, kinerja model juga dianalisis melalui kurva *Receiver Operating Characteristic* (ROC).



Gambar 4. ROC Curve Model KNN

Kurva ROC digunakan untuk menggambarkan hubungan antara *True Positive Rate* (TPR) dan *False*

Positive Rate (FPR) pada berbagai nilai ambang batas (*threshold*) klasifikasi. Melalui kurva ini, dapat diamati kemampuan model dalam membedakan antara kelas positif dan negatif pada berbagai kondisi pengambilan keputusan.

Berdasarkan hasil pengujian, diperoleh nilai ROC-AUC sebesar 0,97 yang menunjukkan bahwa model memiliki kemampuan yang sangat baik dalam mengklasifikasikan data antara kategori benign dan malicious. Nilai AUC yang mendekati 1 mengindikasikan bahwa model memiliki tingkat pemisahan kelas yang tinggi serta mampu mempertahankan performa yang konsisten pada berbagai nilai threshold.

Hasil penelitian menunjukkan bahwa KNN mampu mencapai performa tinggi meskipun menggunakan pendekatan yang relatif sederhana. Hal ini menunjukkan bahwa fitur berbasis network flow memiliki kemampuan representasi yang kuat dalam membedakan traffic normal dan malicious.

Jika dibandingkan dengan penelitian sebelumnya, performa yang diperoleh dalam penelitian ini sebanding bahkan mendekati metode berbasis ensemble dan deep learning, namun dengan kompleksitas yang lebih rendah. Hal ini menjadi keunggulan utama dari pendekatan yang diusulkan.

Selain itu, stabilitas performa pada variasi nilai K menunjukkan bahwa model memiliki tingkat generalisasi yang baik dan tidak terlalu sensitif terhadap perubahan parameter.

D. Pembahasan

Hasil penelitian menunjukkan bahwa algoritma K-Nearest Neighbor (KNN) memiliki kemampuan yang baik dalam mendeteksi aktivitas malware pada jaringan IoT dengan tingkat akurasi yang tinggi. Temuan ini mengindikasikan bahwa fitur-fitur statistik yang dihasilkan dari representasi network flow pada dataset IoT-23 mampu menggambarkan pola lalu lintas jaringan secara efektif.

Dengan karakteristik tersebut, model dapat membedakan secara jelas antara traffic normal (benign) dan traffic yang mengandung serangan (malicious), sehingga mendukung penggunaan pendekatan berbasis machine learning dalam sistem deteksi intrusi pada lingkungan IoT.

Pendekatan machine learning dalam sistem deteksi intrusi memberikan keunggulan dibandingkan metode konvensional berbasis signature, karena:

- Mampu mempelajari pola komunikasi jaringan
- Dapat mendeteksi serangan baru (unknown attack)
- Tidak bergantung pada database signature

Namun demikian, algoritma KNN memiliki keterbatasan, yaitu:

- Kompleksitas komputasi tinggi pada data besar
- Membutuhkan perhitungan jarak terhadap seluruh data latih
- Kurang optimal untuk implementasi real-time skala besar

Secara keseluruhan, temuan dalam penelitian ini menunjukkan bahwa algoritma K-Nearest Neighbor (KNN) memiliki potensi yang cukup tinggi untuk diterapkan dalam sistem Intrusion Detection System (IDS) berbasis machine learning pada lingkungan Internet of Things (IoT). Kemampuan KNN dalam mengklasifikasikan lalu lintas jaringan secara akurat menjadikannya sebagai salah satu metode yang efektif dalam mendukung deteksi aktivitas mencurigakan.

Meskipun demikian, penerapan KNN juga perlu mempertimbangkan aspek efisiensi komputasi, terutama ketika dihadapkan pada volume data yang besar. Oleh karena itu, diperlukan pengembangan lebih lanjut, baik dari sisi optimasi algoritma maupun integrasi dengan metode lain, agar kinerja sistem dapat ditingkatkan secara lebih optimal dan adaptif terhadap berbagai jenis serangan siber.

Nilai K = 5 memberikan keseimbangan terbaik antara bias dan variansi. Nilai K yang terlalu kecil cenderung overfitting, sedangkan nilai yang terlalu besar dapat mengurangi sensitivitas terhadap pola lokal.

Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa terdapat perbedaan yang jelas antara karakteristik lalu lintas malware dan lalu lintas normal pada jaringan Internet of Things (IoT). Perbedaan ini dapat diidentifikasi melalui fitur statistik berbasis *network flow*,

seperti durasi koneksi, jumlah paket, dan volume data. Representasi berbasis *flow* terbukti efektif dalam menggambarkan pola komunikasi jaringan dan dapat dimanfaatkan sebagai dasar dalam proses klasifikasi.

Penerapan algoritma K-Nearest Neighbor (KNN) menunjukkan kinerja yang sangat baik dalam mengklasifikasikan lalu lintas jaringan ke dalam dua kategori, yaitu *benign* dan *malicious*. Hasil evaluasi menunjukkan bahwa model mampu mencapai akurasi hingga 97%, dengan nilai precision, recall, dan F1-score yang konsisten tinggi. Analisis *confusion matrix* juga menunjukkan bahwa tingkat kesalahan klasifikasi relatif rendah. Selain itu, evaluasi menggunakan kurva ROC menghasilkan nilai AUC sebesar 0,97, yang mengindikasikan kemampuan model yang sangat baik dalam membedakan kedua kelas.

Hasil pengujian terhadap variasi parameter K menunjukkan bahwa pemilihan nilai K berpengaruh terhadap performa model. Nilai K = 5 memberikan keseimbangan terbaik antara sensitivitas terhadap pola lokal dan kemampuan generalisasi, sehingga menghasilkan performa yang lebih stabil dibandingkan nilai K lainnya.

Secara keseluruhan, penelitian ini menunjukkan bahwa algoritma K-Nearest Neighbor (KNN) merupakan metode yang efektif untuk diterapkan dalam sistem Intrusion Detection System (IDS) berbasis machine learning pada lingkungan IoT. Pendekatan ini mampu memberikan performa yang tinggi dengan kompleksitas yang relatif rendah, sehingga sesuai untuk lingkungan dengan keterbatasan sumber daya.

Namun, penelitian ini masih memiliki keterbatasan karena hanya menggunakan satu dataset, yaitu IoT-23. Oleh karena itu, pengujian pada dataset lain diperlukan untuk memastikan kemampuan generalisasi model.

Untuk pengembangan selanjutnya, penelitian dapat diarahkan pada penerapan klasifikasi multi-kelas untuk mengidentifikasi jenis serangan secara lebih spesifik. Selain itu, perlu dilakukan perbandingan dengan algoritma lain seperti Random Forest, Support Vector Machine, maupun pendekatan deep learning. Implementasi dalam sistem real-time serta optimasi kompleksitas komputasi juga menjadi langkah penting untuk meningkatkan kinerja pada skala jaringan IoT yang lebih besar.

Daftar Pustaka

[1] W. Adi and F. Kurniawan, “Implementasi keamanan ruangan berbasis IoT dengan sensor PIR, Telegram, dan peringatan suara,” *Journal of Applied Computer Science and Technology*, vol. 5, no. 2, pp. in press, Dec. 2024, doi: 10.52158/jacost.v5i2.860.

[2] H. Rahman and T. Sutabri, “Analisis serangan DDoS menggunakan machine learning pada arsitektur software-defined network,” *JSAI: Journal Scientific and Applied Informatics*, vol. 7, no. 3, 2024.

[3] A. Widodo, A. F. Rochim, and B. Warsito, “Enhancing detection of low-rate DoS attacks using CatBoost-RFE and multilayer perceptron,” *International Journal of Intelligent Engineering and Systems*, vol. 18, no. 4, pp. 701–715, 2025, doi: 10.22266/ijies2025.0531.46.

[4] Y. G. Kim, B. Mendoza, O. Kwon, and J. Yoon, “Task-specific feature selection and detection algorithms for IoT-based networks,” *Journal of Computer and Communications*, vol. 10, no. 10, pp. 59–73, 2022, doi: 10.4236/jcc.2022.1010005.

[5] Y. Meidan *et al.*, “N-BaIoT: Network-based detection of IoT botnet attacks using deep autoencoders,” *IEEE Pervasive Computing*, 2018, doi: 10.1109/MPRV.2018.03367731.

[6] A. Mehrban and P. Ahadian, “Malware detection in IoT systems using machine learning techniques,” *International Journal of Wireless & Mobile Networks*, vol. 15, no. 6, pp. 13–23, Dec. 2023, doi: 10.5121/ijwmn.2023.15602.

[7] A. Rosyida, B. Kanata, C. Ramadhani, and J. T. Elektro, “Optimasi penyaluran bantuan langsung tunai menggunakan algoritma K-Nearest Neighbor di Desa Penedagador,” *Jurnal Pendidikan dan*

Teknologi Indonesia (JPTI), vol. 5, no. 2, pp. 403–418, 2025, doi: 10.52436/1.jpti.742.

[8] A. Verma and V. Ranga, "Machine learning based intrusion detection systems for IoT applications," *Wireless Personal Communications*, 2023, doi: 10.1007/s11277-019-06986-8.

[9] S. Susanto, M. A. S. Arifin, and H. O. L. Wijaya, "IoT botnet detection using autoencoders and decision trees," *Jurnal Sisfokom (Sistem Informasi dan Komputer)*, vol. 12, no. 3, pp. 329–334, Nov. 2023, doi: 10.32736/sisfokom.v12i3.1693.

[10] A. Ichwani, R. Indra Kesuma, A. Setiawan, E. Wicaksono, and R. Hanifah, "Preventing data leakage in classification via integrated machine learning pipelines," *Jurnal Teknik Informatika (JUTIF)*, vol. 7, no. 1, 2026, doi: 10.52436/1.jutif.2026.7.2.5490.

[11] R. W. Pratama and P. T. Hutabarat, "Comparison of harmony search and gravitational search algorithms on decision tree for predicting focus levels," *Jurnal Ilmiah Sistem Informasi*, vol. 5, no. 1, pp. 412–424, Jan. 2026, doi: 10.51903/pyk6nr81.

[12] M. Sanwasih *et al.*, "Deteksi intrusi jaringan berbasis machine learning menggunakan model boosting," *Journal of Information System Research*, vol. 7, no. 2, pp. 489–498, 2026, doi: 10.47065/josh.v7i2.8775.

[13] Z. Munawar *et al.*, *Fundamental Internet of Things (IoT): Memahami Teori dan Penerapannya*. Kaizen Media Publishing, 2023.

[14] A. Sandriana and F. Maulana, "Klasifikasi serangan malware terhadap lalu lintas jaringan IoT menggunakan algoritma K-Nearest Neighbour (K-NN)," *E-JOINT*, vol. 3, no. 1, 2022.

[15] R. P. Ray and A. Sembiring, "Analisis pengaruh fungsi aktivasi CNN terhadap performa klasifikasi hewan," *INCODING: Journal of Informatics and Computer Science Engineering*, vol. 5, no. 2, pp. 184–196, May 2025, doi: 10.34007/incoding.v5i2.847.

[16] M. Chanafy and H. Sulistiani, "Analisis perbandingan kinerja algoritma machine learning untuk klasifikasi kesehatan mental mahasiswa," *SMATIKA JURNAL*, vol. 16, no. 1, pp. 19–29, Mar. 2026, doi: 10.32664