Terbit *online* pada laman web jurnal : http://ejournal.upbatam.ac.id/index.php/comasiejournal

Jurnal Comasie

ISSN (Online) 2715-6265

ANALISIS PERBANDINGAN KINERJA ALGORITMA MACHINE LEARNING BERBASIS FEATURE SELECTION DALAM DETEKSI SERANGAN BOTNET

Rio[1], Koko Handoko[2]

[1]Mahasiswa Program Studi Teknik Informatika, Universitas Putera Batam
[2]Dosen Program Studi Teknik Informatika, Universitas Putera Batam
email: pb210210013@upbatam.ac.id

ABSTRACT

Internet has experienced significant development. Increasing devices connected to internet makes security against cyber attacks a critical issue, thus creates opportunities for cyber attackers, one form of those attack is botnets. In Indonesia, Botnets is the highest traffic anomalies in 2022 by BSSN. High number of attacks because detecting botnet can be challenging, difficulty of detecting attacks and low level of detection accuracy means that normal data sometimes considered an attack, so choosing method that can handle this is very important. Machine learning algorithms are able to study network data traffic and identify suspicious activity, this makes machine learning an effective method. Machine learning based on feature selection has an accuracy of above 90% in detecting DDoS attacks on datasets and machine learning algorithms are also able to detect attack data and normal data. Thus, in this research machine learning algorithms such as K-Nearest Neighbors, Support Vector Machine and Naive Bayes will be applied to dataset containing botnet and normal data to explore how machine learning algorithms can effectively detect botnet attack patterns and normal data. This research compares the performance of commonly used machine learning algorithms to find which one effective for detecting botnet attacks in existing datasets.

Keywords: *Botnet, Dataset, Feature selection, K-Nearset Neigbors, Machine Learning, Naive Bayes, Support Vector Machine*

PENDAHULUAN

Penggunaan internet mengalami perkembangan yang signifikan hal ini ditandai dengan mudahnya mendapatkan akses ke internet dimana kemajuan teknologi informasi menjadi pendukung dari perkembangan penggunaan internet yang secara langsung juga meningkatkan perkembangan teknologi jaringan data baik secara lokal maupun global. Meningkatnya jumlah perangkat seperti komputer dan ponsel yang terhubung pada internet membuat keamanan akan serangan siber menjadi isu kritis dengan banyaknya perangkat ini membuat peluang lebih banyak celah bagi penyerang siber untuk melakukan serangan terhadap perangkat tersebut yang mana salah satunya adalah serangan botnet.

Di Indonesia sendiri, *MyloBot* salah satu bentuk dari *Botnet* menjadi sumber trafik anomali tertinggi pada tahun 2022 dengan 254.260.339 jumlah kasus

Terbit *online* pada laman web jurnal : http://ejournal.upbatam.ac.id/index.php/comasiejournal

Jurnal Comasie

ISSN (Online) 2715-6265

sebagaimana tertera dalam dokumen "Lanskap Keamanan Siber Indonesia Tahun 2022" oleh BSSN. Alasan mengapa tingginya angka serangan botnet disebabkan karena deteksi serangan *botnet* bisa menjadi sebuah tantangan, sulitnya mendeteksi serangan nya dan rendahnya tingkat akurasi pendeteksian serangan menyebabkan tingginya *false detection* sehingga data normal dianggap menjadi serangan. *Botnet* ini kumpulan dari aplikasi *bot* (robot) yang dikonfigurasi untuk dapat berjalan secara otomatis dalam jaringan maka setiap komputer yang telah terinfeksi dan tergabung dalam jaringan botnet akan mengeksekusi perintah atau instruksi yang diberikan oleh *Botmaster* dengan dilakukan dari jarak jauh. *Botnet* mampu menyediakan platform yang dapat didistribusikan pada kegiatan illegal seperti *spam*, *phishing, click fraud*, pencurian kata sandi dan *Distributed Denial of Service (DdoS)* (Xing et al., 2021).

Metode yang akan digunakan dalam penelitian ini menggunakan *machine learning*. Algoritma *machine learning* mampu mempelajari data trafik jaringan dan mengidentifikasi aktivitas yang mencurigakan, hal ini membuat *machine learning* menjadi metode yang efektif. *Machine learning* dengan berbasis fitur seleksi memiliki akurasi diatas 90% dalam mendeteksi serangan DdoS pada dataset(Maslan et al., 2020), algoritma machine learning juga mampu mendeteksi serangan dan data normal (Darryl & Subali, 2021). Dengan demikian, dalam penelitian ini *Machine learning* akan diterapkan kedalam sebuah dataset yang berisi data botnet dan normal dan mengeksplorasi bagaimana algoritma *machine learning* dapat secara efektif mendeteksi yang mana pola serangan *botnet* dan yang mana data normal.

Masalah utama dalam mendeteksi serangan botnet adalah jumlah dan kompleksitas data, serta fitur yang tidak relevan yang menurunkan performa model. Penelitian ini fokus pada deteksi serangan botnet dengan menggunakan fitur seleksi untuk mengurangi kompleksitas dan meningkatkan akurasi deteksi, sambil mempertahankan keterwakilan dataset. Algoritma machine learning seperti Naive Bayes, SVM, dan KNN diterapkan dengan fitur seleksi untuk mengatasi masalah ini secara efisien dan meningkatkan akurasi model deteksi.

KAJIAN TEORI

2.1 Botnet

Menurut (Moorthy & Nathiya, 2022) Botnet adalah jaringan robot yaitu kumpulan jaringan yang terhubung satu sama lain yang ditangani oleh satu kepala.Kepala yang mengontrol semua jaringan yang terhubung ini disebut sebagai bot-herder. Jika penyerang memutuskan untuk menyerang suatu organisasi seperti Google atau Amazon, dia membutuhkan banyak kekuatan untuk membobol keamanan firewall server web tersebut. Pada dasarnya, penyerang membentuk pasukan dengan jaringan malware-nya, pertama-tama dia menyerang pengguna individu dengan metode peretasan. Dengan serangan *phishing* atau serangan *malware*, penyerang mengambil alih komando dan kendali perangkat. Dengan bantuan dari jaringan ini penyerang dapat melakukan serangan DDoS atau *brute force attack* yang akan mengganggu lalu lintas jaringan atau membuat server jaringan korban *crash.*

Terbit *online* pada laman web jurnal : http://ejournal.upbatam.ac.id/index.php/comasiejournal

Jurnal Comasie

ISSN (Online) 2715-6265

2.2 Algoritma *Machine Learning*

Menurut (Sharifani & Amini, 2023) *Machine learning* digunakan untuk tugas-tugas seperti pengenalan gambar, bahasa alami dalam pemrosesan, dan pemodelan prediktif. data dalam jumlah besar sangat penting untuk keberhasilan *Machine learning* dimana kualitas dan keragaman data yang digunakan untuk pelatihan dapat sangat mempengaruhi keakuratan dan generalisasi model. Algoritma dalam machine learning yang akan digunakan adalah

1. Algoritma *K-Nearest Neighbor*, algoritma yang paling simple pada *machine learning* yang berbasis *supervised learning*. Untuk cara kerjanya pertama-tama, tentukan jumlah K dari ketetanggaannya, kemudian gunakan rumus *Euclidean Distance* untuk menghitung jarak antara K tetangga, ambil K tetangga yang paling dekat per perhitungan ini untuk dihitung angka dari data point setiap kategori antara setiap K tetangga terdekat ini, data point baru akan ditetapkan pada kategori yang mana nilai K tetangga tertinggi. (Suyal & Goyal, 2022)
2. Algoritma *Support Vector Machine*, pertama kali diusulkan oleh Vapnik merupakan algoritma *supervised learning* yang digunakan untuk klasifikasi dan regresi. Target umum dari SVM adalah untuk menemukan hyperplane individu dengan margin tertinggi yang bisa membagi kelas secara linear. Tujuan dari SVM untuk mengidentifikasi kumpulan data di mana jumlah data pelatihan terbatas dan di mana solusi optimal tidak dapat dijamin dengan penggunaan normal sejumlah besar statistik Di ruang masukan, saat pemisahan linier menjadi lebih mudah dengan menggunakan fungsionalitas kernel, data pelatihan diproyeksikan ke ruang fitur berdimensi lebih tinggi. supaya menemukan *hyperplane* yang lebih baik dalam membagi data ke dalam kelompok-kelompoknya, SVM menggunakan fungsi kernel yang berbeda seperti radial fungsi dasar (RBF) atau kernel polinomial dan bila digunakan untuk set pelatihan yang minimal, memiliki klasifikasi yang akan memiliki kinerja yang baik. Oleh karena itu, agar dapat mengevaluasi *hyperplanes* dengan benar dan mengurangi kesalahan klasifikasi, maka algoritma SVM memerlukan fungsi kernel yang tepat (Muawanah et al., 2023).
3. Algoritma *Naive Bayes*, algoritma klasifikasi untuk masalah klasifikasi multikelas. Disebut dengan *Naive Bayes* dikarenakan menggunakan perhitungan probabilitas untuk setiap kelas akan disederhanakan agar perhitungannya mudah dilakukan. Pengklasifikasi *Naive Bayes* juga dibangun berdasarkan metode klasifikasi *Bayesian*, Ini bergantung pada teorema *Bayes* yaitu sebuah persamaan yang menggambarkan hubungan kondisional probabilitas dari kuantitas statistik (Viet et al., 2021).

2.3 HyperParameter

Hyperparameter adalah nilai-nilai yang ditentukan sebelum proses pelatihan model algoritma dimulai. *Hyperparameter* mengontrol cara kerja algoritma selama pelatihan. Ada beberapa *hyperparameter* yang digunakan pada penelitian ini, yaitu :

1. K tetangga KNN, *Hyperparameter* K atau tetangga terdekat pada KNN

Terbit *online* pada laman web jurnal : http://ejournal.upbatam.ac.id/index.php/comasiejournal

Jurnal Comasie

ISSN (Online) 2715-6265

dilakukan untuk mencari kelompok k objek dalam data training yang paling dekat (mirip) dengan objek pada data baru atau disebut data *testing*. *Crossvalidation* atau sering disebut estimasi rotasi, adalah teknik validasi model yang digunakan untuk menilai seberapa baik hasil analisis statistik dapat digeneralisasi ke kumpulan data yang independen. Teknik ini terutama sering diterapkan untuk mengevaluasi model prediktif dan memperkirakan tingkat akurasi model ketika digunakan dalam praktik nyata. Salah satu metode yang populer dalam *cross-validation* adalah *k-fold cross-validation*, di mana dataset dibagi menjadi *K* bagian atau lipatan dengan ukuran yang sama. Setiap lipatan bergantian digunakan sebagai data uji, sementara itu sisanya digunakan sebagai data pelatihan. Teknik ini membantu mengurangi bias pada data dan memberikan evaluasi yang lebih andal terhadap performa deteksi model. Misalnya, penggunaan *5-fold cross-validation* membagi dataset menjadi lima lipatan untuk meminimalkan bias dan menghasilkan estimasi akurasi yang lebih konsisten (Azis et al., 2020).

2. Kernel dan C SVM, Fungsi kernel dalam SVM biasanya digunakan untuk menyelesaikan kasus non linier karena kebanyakan kasus dalam dunia nyata jarang terdapat kasus yang bersifat linier. Fungsi kernel ini akan dimasukkan kedalam algoritma SVM, yaitu kernel linier. Untuk mendapatkan nilai *support vector* atau biasa disebut *alpha*, ditentukan parameter C. Parameter C sendiri berguna untuk mengontrol antara memaksimalkan sebuah margin dan meminimalkan kesalahan pelatihan. Nilai parameter C yang bisa untuk digunakan adalah 0.000001, 0.0001, 0.01, 1, 100, 10000, 1000000 (Pratiwi & Setyawan, 2021).
3. *Gaussian Naive Bayes,* algoritma *hyperparameter* ini digunakan untuk klasifikasi paling sederhana dan paling terkenal yang menggunakan teorema Bayes. Oleh karena itu, beberapa variabel kontinu dikonversi menjadi nilai diskrit sehingga set pelatihan berisi nilai kontinu yang diklasifikasikan menurut kategori dan menghitung rata-rata serta deviasi setiap klasifikasi. Keuntungan dari *GaussianNB* adalah dapat diterapkan pada klasifikasi yang praktis, membutuhkan sedikit data pelatihan, dan dapat dilatih secara efektif dalam pengawasan. Sedangkan itu yang menjadi kelemahannya adalah berupa atributnya ini diasumsikan independen (Agustina & Saraswati, 2024). Tidak seperti algoritma lain, *Naive Bayes* hanya memiliki jumlah hyperparameter yang sedikit, salah satunya *gaussianNB*.

2.4 *Feature Selection*

Pengenalan pola adalah salah satu aplikasi pembelajaran mesin yang paling penting dalam berbagai ilmu. Menurut (Rostami et al., 2022), Tujuan dari pemodelan dan klasifikasi data untuk memprediksi berdasarkan data latih dan fitur yang tersedia. Kumpulan data besar dengan ruang fitur berdimensi tinggi dan jumlah sampel yang relatif lebih kecil merupakan masalah penting bagi tugas *machine learning*, ketika terdapat sejumlah fitur yang tidak relevan dan mubazir di antara kumpulan fitur awal, maka pengurangan dimensi adalah salah satu teknik penting untuk menghilangkan fitur-fitur ini. Teknik ini efisien untuk tingkatkan kinerja akurasi, menurunkan

Terbit *online* pada laman web jurnal : http://ejournal.upbatam.ac.id/index.php/comasiejournal

Jurnal Comasie

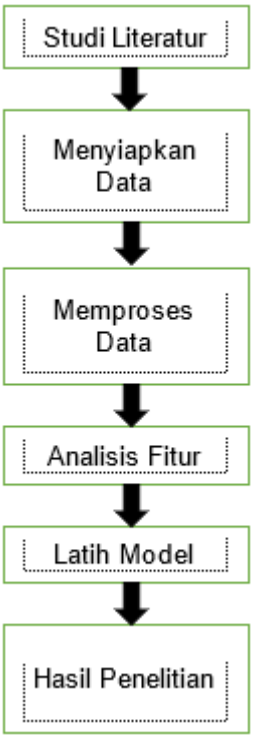
ISSN (Online) 2715-6265

kompleksitas, membangun model yang lebih umum, dan juga mengurangi penyimpanan yang diperlukan.

METODE PENELITIAN

3.1 Desain Penelitian

Desain penelitian menyediakan kerangka dan alur kerja mencakup sepanjang proses penelitian. Penulis membagi penelitian menjadi beberapa tahap sebagai berikut :



Gambar 1. Desain Penelitian
(Sumber: Data Penelitian, 2025)

Dari gambar desain penelitian diatas, penjelasan setiap kotak adalah sebagai berikut :

1. Studi Literatur

Tahap pertama adalah studi literatur, yang mencakup identifikasi masalah dan penentuan judul penelitian. Peneliti menganalisis tantangan yang ada dalam mendeteksi serangan botnet serta menentukan dataset, algoritma, dan metode yang dapat meningkatkan akurasi deteksi..

2. Menyiapkan Data

Tahap selanjutnya adalah menyiapkan data, di mana peneliti mengumpulkan dataset yang relevan untuk melatih model machine learning dalam mendeteksi serangan selama periode penelitian dari September 2024 hingga Januari 2025.

3. Memproses Data

Tahap ketiga adalah pemrosesan data, di mana dataset dibagi menjadi data latih dan data uji. Data latih menjalani preprocessing, termasuk data cleaning untuk menghapus duplikat dan missing value, serta normalisasi untuk menyeragamkan rentang nilai fitur agar siap melatih model deteksi.

4. Analisis Fitur

Tahap keempat adalah menganalisis fitur dengan memilih fitur relevan dari data latih yang telah dibersihkan dengan menggunakan teknik seleksi fitur. Fitur terbaik disimpan dalam dataset baru untuk digunakan dalam penelitian.

5. Latih Model

Tahap kelima melatih model dengan dataset baru yang berisi data latih dengan fitur pilihan. Algoritma machine learning dilatih sesuai parameter masing-masing untuk membangun sebuah model deteksi serangan dan data normal pada data uji.

6. Hasil Penelitian

Tahap keenam mengevaluasi hasil model dengan menguji data latih yang telah dinormalisasi sesuai dengan data latih. Evaluasi dilakukan dengan menggunakan *confusion matrix* serta metrik akurasi, presisi, *recall*, dan *F1-Score* untuk membandingkan performa tiap model

Terbit *online* pada laman web jurnal : http://ejournal.upbatam.ac.id/index.php/comasiejournal

Jurnal Comasie

ISSN (Online) 2715-6265

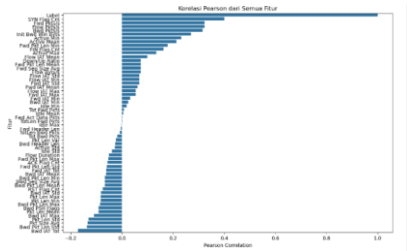
dalam mendeteksi serangan dan data normal.

3.2 Tahap *Pre Processing*

Peneliti menggunakan *dataset* CSV yang diambil dari github, ini merupakan hasil olahan dari *CTU-13 Dataset,* sebuah *dataset* jaringan *botnet* yang ditangkap di CTU University, Czech Republic pada tahun 2011. *Dataset* ini dihasilkan dengan memproses file *Pcap* dari *Dataset CTU-13* menggunakan *CICFlowMeter Tool*. Untuk membuat kumpulan data CSV ini, semua file *Pcap* dari *Dataset CTU-13* diimpor dengan menggunakan Alat *CICFlowMeter*. Alat ini mengubah File *Pcap* yang diberikan menjadi File CSV berdasarkan Aliran Paket, kemudian semua sampel lalu lintas serangan digabung dalam *Attack_Traffic.csv* berdasarkan ip Serangan. Demikian pula, semua sampel lalu lintas normal digabung ke dalam *Normal_Traffic.csv* dengan memakai semua ip selain ip Serangan. Setelah mendapatkan *dataset* peneliti menggunakan *JupyterLab* dan metode *data.duplicate* dan *data.isnull* untuk mencari data duplikat dan *missing values,* peneliti juga melakukan normalisasi data dengan menggunakan *standardscaler* untuk menyamakan *value* tiap fitur dalam suatu rentang yang sama.

3.3 Analisis Fitur

Untuk menganalisis fitur peneliti menggunakan pustaka *pandas,matplotlib, scikit-learn* dan *numpy*, pertama-tama memisahkan fitur dan target (label) kemudian menghitung korelasi 1 dan -1 dengan *korelasi pearson*, hasil yang ditampilkan adalah fitur dengan korelasi tertinggi sampai terendah. Hasil yang didapatkan bisa dilihat pada gambar dibawah :



Gambar 2. Fitur *Dataset* dengan *Korelasi Pearson*
(Sumber: Data Penelitian, 2025)

Berdasarkan grafik diatas maka peneliti menggunakan 15 fitur dengan hasil korelasi tertinggi untuk melatih model algoritma *machine learning*.

3.4 Model Algoritma

Pada tahap ini peneliti menggunakan 3 algoritma *machine learning* yaitu *K-Nearest Neighbours, Support Vector Machine,* dan *Naïve Bayes*, ketiga algoritma ini adalah algoritma yang cukup umum digunakan dalam *machine learning* untuk masalah klasifikasi dan juga pendeteksian. Hasil pelatihan ketiga algoritma adalah sebagai berikut

1. KNN dilatih dengan menggunakan *5-fold cross-validation* hal ini karena karakteristik dari algoritma ini. Untuk menentukan nilai k terbaik, dilakukan pengujian terhadap nilai k=3 hingga k=7, dan hasilnya k=4 memberikan akurasi tertinggi dengan rata-rata 77.8%.
2. SVM dilatih menggunakan *kernel linear* dengan *GridSearch* dan *cross-validation* untuk menentukan yang mana parameter C terbaik. Dari pengujian pada C = 0.1, 0.001, 0.0001, 1, 100, dan 1000, nilai C = 1, 100, dan 1000 memberikan nilai yang sama dan akurasi tertinggi. Oleh karena itu, peneliti memilih C =

Terbit *online* pada laman web jurnal : http://ejournal.upbatam.ac.id/index.php/comasiejournal

Jurnal Comasie

ISSN (Online) 2715-6265

1 dengan akurasi 84% sebagai acuan model.

3. Naïve Bayes dilatih dengan memakai *GaussianNB* dengan menggunakan parameter *var_smoothing*, lalu diuji melalui GridSearch dan Cross Validation pada nilai 1e-13 hingga 1e-5. Karena hasil kurang optimal, peneliti menambahkan prediksi probabilitas dengan threshold 0.5 untuk meningkatkan recall kelas 1. Model akhirnya mencapai akurasi 75%.

HASIL DAN PEMBAHASAN

4.1 Hasil Penelitian

Untuk menguji akurasi model latih dalam mendeteksi serangan *botnet* dengan data uji maka digunakan beberapa metode seperti *Confusion Matrix* yang mana bisa menggambarkan hasil prediksi model yang sebenarnya serta membantu untuk lebih memahami bagaimana model dalam mendeteksi atau klasifikasi kelas, metode lainnya yang peneliti gunakan untuk menganalisis kinerja tiap algoritma adalah :

1. Akurasi, menghitung prediksi yang betul dari total data.
2. Presisi, Mengukur seberapa tepat model dalam mengklasifikasi data positif.
3. *Recall*, mengukur seberapa baik model dalam menemukan semua data positif.
4. *F1-Score*, rata-rata antara presisi dan *recall*, memberikan penilaian yang lebih seimbang terhadap pengaruh ketidakseimbangan antara presisi dan *recall*.

Output deteksi dari model KNN, SVM dan *Naive Bayes* menggunakan *Python* bisa dilihat pada gambar dibawah.

1. KNN

```
=== Evaluation Results ===
+----------------------+--------+
|        Metric        | Value  |
+----------------------+--------+
|       Accuracy       | 0.7910 |
| Precision (Rata-rata) | 0.7922 |
|  Recall (Rata-rata)  | 0.7910 |
| F1-Score (Rata-rata) | 0.7876 |
+----------------------+--------+

Confusion Matrix:
+---+--------------+--------------+
|   | Terdeteksi 0 | Terdeteksi 1 |
+---+--------------+--------------+
| 0 |     9363     |     1289     |
| 1 |     2565     |     5226     |
+---+--------------+--------------+

Classification Report:
+---+--------------+-----------+--------+----------+---------+
|   | Class/Metric | Precision | Recall | F1-Score | Support |
+---+--------------+-----------+--------+----------+---------+
| 0 |      0       |  0.7850   | 0.8790 |  0.8293  |  10652  |
| 1 |      1       |  0.8021   | 0.6708 |  0.7306  |  7791   |
| 2 |   Accuracy   |  0.7910   | 0.7910 |  0.7910  |    0    |
| 3 |  macro avg   |  0.7936   | 0.7749 |  0.7800  |  18443  |
| 4 | weighted avg |  0.7922   | 0.7910 |  0.7876  |  18443  |
+---+--------------+-----------+--------+----------+---------+
```

Gambar 3. Output evaluasi KNN (Sumber: Data Penelitian, 2025)

2. SVM

```
=== Evaluation Results ===
+----------------------+--------+
|        Metric        | Value  |
+----------------------+--------+
|       Accuracy       | 0.8446 |
| Precision (Rata-rata) | 0.8443 |
|  Recall (Rata-rata)  | 0.8446 |
| F1-Score (Rata-rata) | 0.8438 |
+----------------------+--------+

Confusion Matrix:
+---+--------------+--------------+
|   | Terdeteksi 0 | Terdeteksi 1 |
+---+--------------+--------------+
| 0 |     9475     |     1177     |
| 1 |     1689     |     6102     |
+---+--------------+--------------+

Classification Report:
+---+--------------+-----------+--------+----------+---------+
|   | Class/Metric | Precision | Recall | F1-Score | Support |
+---+--------------+-----------+--------+----------+---------+
| 0 |      0       |  0.8487   | 0.8895 |  0.8686  |  10652  |
| 1 |      1       |  0.8383   | 0.7832 |  0.8098  |  7791   |
| 2 |   Accuracy   |  0.8446   | 0.8446 |  0.8446  |    0    |
| 3 |  macro avg   |  0.8435   | 0.8364 |  0.8392  |  18443  |
| 4 | weighted avg |  0.8443   | 0.8446 |  0.8438  |  18443  |
+---+--------------+-----------+--------+----------+---------+
```

Gambar 4. Output evaluasi SVM (Sumber: Data Penelitian, 2025)

3. *Naïve Bayes*

Terbit *online* pada laman web jurnal : http://ejournal.upbatam.ac.id/index.php/comasiejournal

Jurnal Comasie

ISSN (Online) 2715-6265

```
=== Evaluation Results ===
+----------------------+--------+
|        Metric        | Value  |
+----------------------+--------+
|       Accuracy       | 0.7551 |
| Precision (Rata-rata) | 0.7588 |
|  Recall (Rata-rata)   | 0.7551 |
| F1-Score (Rata-rata)  | 0.7479 |
+----------------------+--------+

Confusion Matrix:
+---+--------------+--------------+
|   | Terdeteksi 0 | Terdeteksi 1 |
+---+--------------+--------------+
| 0 |     9358     |     1294     |
| 1 |     3223     |     4568     |
+---+--------------+--------------+

Classification Report:
+---+--------------+-----------+--------+----------+---------+
|   | Class/Metric | Precision | Recall | F1-Score | Support |
+---+--------------+-----------+--------+----------+---------+
| 0 |      0       |  0.7438   | 0.8785 |  0.8056  |  10652  |
| 1 |      1       |  0.7793   | 0.5863 |  0.6692  |  7791   |
| 2 |   Accuracy   |  0.7551   | 0.7551 |  0.7551  |    0    |
| 3 |  macro avg   |  0.7615   | 0.7324 |  0.7374  |  18443  |
| 4 | weighted avg |  0.7588   | 0.7551 |  0.7479  |  18443  |
+---+--------------+-----------+--------+----------+---------+
```

Gambar 5. Output evaluasi Naïve Bayes
(Sumber: Data Penelitian, 2025)

a. Pembahasan

Tahap ini akan membahas hasil dari evaluasi setiap model algoritma. Untuk mendapatkan gambaran atas performa dan kinerja tiap modal dalam mendeteksi serangan dan seberapa efektifnya model-model ini dalam mendeteksi *dataset*, peneliti menggabungkan hasil output ketiga algoritma kedalam tabel. Untuk parameter *confusion matrix*, metrik kinerjanya mencakup :

1. *True Negative* (TN) : mencakup jumlah prediksi benar dari kelas 0,
2. *True Positive* (TP) : mencakup jumlah prediksi benar dari kelas 1,
3. *False Postive* (FP) : mencakup jumlah prediksi salah dari kelas 0,
4. *False Negative* (FN) : mencakup jumlah prediksi salah dari kelas 1.

Tabel 1. Hasil *Confusion Matrix* ketiga algoritma

Algoritma	*True Negative*	*False Positive*	*False Negative*	*True Positive*
KNN	9363	1289	2565	5226
SVM	9475	1177	1689	6102
NB	9358	1294	3223	4568

(Sumber: Data Penelitian, 2025).

Analisis dari hasil evaluasi confusion matrik dari ketiga algoritma adalah,

1. KNN memiliki jumlah *False positif* yang mencapai 1289 data, artinya sebanyak 1289 data normal salah diartikan menjadi data serangan ini berarti model ini cukup banyak memberikan alarm palsu, selain itu 2565 data serangan tidak terdeteksi dan salah diklasifikasikan menjadi data normal. Secara keseluruhan performanya lebih baik pada kelas (0) daripada kelas 1 (serangan).
2. SVM memiliki jumlah *false positif* yang mencapai 1177 data, model ini memiliki *false alarm* yang lebih sedikit dari model KNN, SVM juga memiliki *false negatif* yang jauh lebih sedikit dari model KNN dengan 1689 data. Secara keseluruhan SVM lebih baik dibanding KNN dalam mendeteksi kelas serangan dan normal.
3. NB memiliki *false positif* yang paling tinggi yaitu 1294 data, kinerja model ini juga sangat buruk dalam mendeteksi serangan dengan *false negatif* mencapai 3223 data. Secara

Terbit *online* pada laman web jurnal : http://ejournal.upbatam.ac.id/index.php/comasiejournal

Jurnal Comasie

ISSN (Online) 2715-6265

keseluruhan model ini tidak cukup sensitif dalam mendeteksi serangan pada *dataset* ini.

Untuk performa setiap algoritma, peneliti menggunakan metrik akurasi, presisi, *recall* dan *f-1 score* untuk menentukan efektifitas tiap algoritma, hasilnya adalah :

1. KNN bekerja lebih baik dalam mendeteksi kelas normal dibanding kelas serangan. *Recall* untuk serangan cukup rendah yaitu 67% dibandingkan kelas normal yang mencapai 87.90%, artinya cukup banyak serangan yang terlewatkan dalam deteksi sedangkan model ini baik dalam mendeteksi data normal.
2. SVM memiliki performa yang lebih baik dibandingkan KNN, model ini mempunyai keseimbangan antara deteksi data normal dan serangan dengan *recall* data normal 88.95% dan data serangan yang mencapai 78.32%, dengan ini SVM lebih cocok digunakan dalam mendeteksi serangan dibandingkan KNN.
3. NB memiliki presisi yang cukup untuk kelas serangan berupa 77.93% namun *recall* nya jauh lebih rendah dengan 58.63%, ini artinya model ini lebih banyak mendeteksi serangan dengan benar namun cenderung gagal mengenali beberapa contoh kelas 1 lainnya.

SIMPULAN

Berdasarkan hasil penelitian ini SVM menunjukkan performa terbaik dalam mendeteksi serangan *botnet* dengan akurasi 84% dan F1-Score 80%, diikuti oleh KNN dengan akurasi 79% dan *F1-Score* 73%. NB memiliki performa terendah dengan akurasi 75% dan F1-Score 66%. KNN lebih baik dalam mendeteksi data normal tetapi kurang efektif dalam mendeteksi serangan. SVM memiliki keseimbangan terbaik antara deteksi data normal dan serangan, sedangkan NB kurang sensitif terhadap serangan dengan banyak yang tidak terdeteksi.

DAFTAR PUSTAKA

Agustina, V., & Saraswati, S. (2024). *Implementasi Metode Naïve Bayes Classifier Dalam Analisis Sentimen Opini Publik Twitter Tentang G20 Indonesia*.

Azis, H., Purnawansyah, P., Fattah, F., & Putri, I. P. (2020). Performa Klasifikasi K-NN dan Cross Validation Pada Data Pasien Pengidap Penyakit Jantung. *ILKOM Jurnal Ilmiah*, *12*(2), 81–86. https://doi.org/10.33096/ilkom.v12i2.507.81-86

Darryl, P., & Subali, M. (2021). Perbandingan Algoritma SVM dan Algoritma KNN Dalam Menghasilkan Klasifikasi DDoS dan Benign. *Jurnal Ilmiah KOMPUTASI*.

Maslan, A., Mohamad, K. M. Bin, & Mohd Foozy, F. B. (2020). Feature selection for DDoS detection using classification machine learning techniques. *IAES International Journal of Artificial Intelligence*, *9*(1), 137–145. https://doi.org/10.11591/ijai.v9.i1.pp137-145

Moorthy, R. S. S., & Nathiya, N. (2022). Botnet Detection Using Artificial Intelligence. *Procedia Computer Science*, *218*, 1405–1413. https://doi.org/10.1016/j.procs.2023.01.119

Muawanah, S., Muzayanah, U., Pandin, M. G. R., Alam, M. D. S., & Trisnaningtyas, J. P. N. (2023).

Terbit *online* pada laman web jurnal : http://ejournal.upbatam.ac.id/index.php/comasiejournal

Jurnal Comasie

ISSN (Online) 2715-6265

Stress and Coping Strategies of Madrasah's Teachers on Applying Distance Learning During COVID-19 Pandemic in Indonesia. *Qubahan Academic Journal*, *3*(4), 206–218. https://doi.org/10.48161/Issn.2709-8206

Pratiwi, N., & Setyawan, Y. (2021). ANALISIS AKURASI DARI PERBEDAAN FUNGSI KERNEL DAN COST PADA SUPPORT VECTOR MACHINE STUDI KASUS KLASIFIKASI CURAH HUJAN DI JAKARTA. *Journal of Fundamental Mathematics and Applications (JFMA)*, *4*(2), 203–212. https://doi.org/10.14710/jfma.v4i2.11691

Rostami, M., Berahmand, K., & Forouzandeh, S. (2022). *Review of Swarm Intelligence-based Feature Selection Methods*.

Sharifani, K., & Amini, M. (2023). Machine Learning and Deep Learning: A Review of Methods and Applications. In *World Information Technology and Engineering Journal* (Vol. 10). https://ssrn.com/abstract=4458723

Suyal, M., & Goyal, P. (2022). A Review on Analysis of K-Nearest Neighbor Classification Machine Learning Algorithms based on Supervised Learning. In *International Journal of Engineering Trends and Technology* (Vol. 70, Issue 7, pp. 43–48). Seventh Sense Research Group. https://doi.org/10.14445/22315381/IJETT-V70I7P205

Viet, T. N., Minh, H. Le, Hieu, L. C., & Anh, T. H. (2021). The naïve bayes algorithm for learning data analytics. *Indian Journal of Computer Science and Engineering*, *12*(4), 1038–1043. https://doi.org/10.21817/indjcse/2021/v12i4/211204191

Xing, Y., Shu, H., Zhao, H., Li, D., & Guo, L. (2021). Survey on Botnet Detection Techniques: Classification, Methods, and Evaluation. In *Mathematical Problems in Engineering* (Vol. 2021). Hindawi Limited. https://doi.org/10.1155/2021/6640499

	Penulis pertama, Rio yang merupakan mahasiswa Prodi Sistem Infomasi Universitas Putera Batam. Mahasiswa yang aktif dalam bidang informatika
	Penulis kedua, Koko Handoko, S. Kom., yang merupakan Dosen pembimbing Prodi Teknik Informatika Universitas Putera Batam. Penulis aktif sebagai tenaga kerja dan peneliti