Biner : Jurnal Ilmu Komputer, Teknik dan Multimedia
Volume 2, No. 3, Agustus 2024
ISSN 2988-3814 (media online)
Hal 243-253

Efektivitas Peran Keamanan Jaringan Dalam Melindungi Data Perusahaan Dari Ancaman Serangan Siber

Kuntadi Fais Daffa Ghazy[1], Ari Syaripudin[2*]

[1,2]Fakultas Ilmu Komputer, Program Studi Teknik Informatika, Universitas Pamulang, Kota Tangerang Selatan, Indonesia
Email: [1]daffafais9@gmail.com, [2*]dosen00671@unpam.ac.id
(* : coressponding author)

Abstrak − Penelitian ini bertujuan untuk mengevaluasi efektivitas peran keamanan jaringan dalam melindungi data perusahaan dari ancaman serangan siber. Metode yang digunakan dalam penelitian ini meliputi *Gaussian Naive Bayes*, *Bernoulli Naive Bayes*, dan *Multinomial Naive Bayes*. Data yang dianalisis merupakan log *firewall Fortigate* dari sistem *Intrusion Prevention System (IPS)* yang mencatat aktivitas ancaman terhadap jaringan Januari Nusa Network Prakarsa. Data dikumpulkan dari bulan Januari hingga Februari 2024, mencakup 43.184 entri yang berisi informasi tentang alamat IP sumber dan tujuan serangan, tingkat keparahan, *port*, negara asal, layanan yang diserang, dan jenis *malware* atau virus. Proses *preprocessing* yang dilakukan meliputi *Data Cleaning*, Label *Encoding*, dan Seleksi Fitur menggunakan ANOVA. Hasil penelitian menunjukkan bahwa model *Gaussian Naive Bayes* dengan k = 6 memberikan kinerja terbaik dengan akurasi sebesar 0.709. Fitur terbaik yang digunakan dalam model ini meliputi *"Source IP"*, *"Destination IP"*, *"Severity"*, *"Destination Port"*, *"Service"*, dan *"Name Malware/Virus"*. Akurasi ini lebih tinggi dibandingkan dengan nilai rata-rata *cross-validation* sebesar 0.707. Meskipun terdapat variasi performa antar kelas, secara keseluruhan model ini berhasil memberikan prediksi yang cukup baik. Penelitian ini menyimpulkan bahwa penggunaan algoritma *Naive Bayes* efektif dalam mendeteksi dan mencegah serangan siber, sehingga dapat meningkatkan keamanan data perusahaan.

Kata Kunci: Keamanan Jaringan, *Naive Bayes*, *IPS*, Serangan Siber, Data Perusahaan

Abstract − *This study aims to evaluate the effectiveness of network security in protecting company data from cyber threats. The methods used in this research include Gaussian Naive Bayes, Bernoulli Naive Bayes, and Multinomial Naive Bayes. The data analyzed are Fortigate firewall logs from the Intrusion Prevention System (IPS), recording threat activities against PT Nusa Network Prakarsa"s network. Data were collected from January to February 2024, encompassing 43,184 entries containing information on Source and Destination IP addresses, Severity levels, ports, origin countries, attacked Services, and types of malware or viruses. The preprocessing steps included Data Cleaning, Label Encoding, and Feature Selection. The results showed that the Gaussian Naive Bayes model with k = 6 provided the best performance with an accuracy of 0.709. The best features used in this model included "Source IP", "Destination IP", "Severity", "Destination Port", "Service", and "Name Malware/Virus". This accuracy is higher than the average cross-validation value of 0.707. Although there is variation in performance across classes, overall, the model successfully provided reasonably good predictions. This research concludes that using Naive Bayes algorithms is effective in detecting and preventing cyber attacks, thereby enhancing company data security.*

Keywords*: Network Security, Naive Bayes, IPS, Cyber Attacks, Company Data*

1. PENDAHULUAN

Dalam era digital saat ini, keamanan jaringan menjadi isu krusial bagi perusahaan untuk melindungi data dari ancaman siber yang terus meningkat. Perkembangan teknologi yang pesat menyebabkan peningkatan jumlah serangan siber, yang berdampak signifikan terhadap perusahaan, baik dari segi finansial maupun reputasi (Fatkhurohman & Pujastuti, 2019). Menurut Laporan Tahunan Ancaman Keamanan Siber Cisco 2023, rata-rata organisasi mengalami 86 serangan *phishing* dan *malware* per minggu pada tahun 2022, menunjukkan tingginya frekuensi serangan siber yang dihadapi perusahaan (CISCO, 2023). Oleh karena itu, deteksi dini dan klasifikasi serangan menjadi penting untuk menjaga integritas, kerahasiaan, dan ketersediaan sistem informasi.

Sistem Deteksi Intrusi (IDS) adalah komponen penting dalam keamanan jaringan yang bertujuan untuk mengidentifikasi dan merespons aktivitas mencurigakan atau tidak sah. Dengan memanfaatkan *Machine Learning*, IDS dapat menjadi lebih cerdas dan adaptif, menganalisis pola lalu lintas jaringan dan perilaku pengguna secara real-time untuk mengenali anomali yang

Biner : Jurnal Ilmu Komputer, Teknik dan Multimedia
Volume 2, No. 3, Agustus 2024
ISSN 2988-3814 (media online)
Hal 243-253

menunjukkan adanya upaya intrusi (Anas & Zakir, 2024). Salah satu algoritma *Machine Learning* yang sering digunakan adalah *Naive Bayes*, yang memiliki beberapa keunggulan seperti kecepatan, kesederhanaan, dan akurasi yang tinggi. *Naive Bayes* juga efektif dengan data pelatihan yang sedikit dan mudah diimplementasikan (Putro et al., 2020). Terdapat beberapa varian *Naive Bayes*, termasuk *Gaussian Naive Bayes* untuk data kontinu, *Bernoulli Naive Bayes* untuk data biner, dan *Multinomial Naive Bayes* untuk data diskrit seperti frekuensi kata dalam dokumen teks (Umar & M. Adnan Nur, 2022).

Penelitian sebelumnya menunjukkan keberhasilan *Naive Bayes* dalam berbagai aplikasi. Misalnya, model *Multinomial Naive Bayes* mencapai akurasi 98% dalam mendeteksi ujaran kebencian berbahasa Jawa di Twitter (Yati et al., 2023) dan akurasi 86.74% dalam klasifikasi sentimen ulasan aplikasi *Amazon Shopping* (Ernianti Hasibuan & Heriyanto, 2022). Berdasarkan latar belakang ini, penelitian ini bertujuan untuk mengevaluasi efektivitas sistem deteksi intrusi berbasis *Naive Bayes* dalam melindungi data perusahaan dari serangan siber.

Penelitian ini mengidentifikasi beberapa masalah utama, termasuk kualitas dan relevansi data serangan siber, pemilihan fitur yang optimal, serta evaluasi dan optimasi kinerja model. Tantangan dalam memastikan data yang digunakan berkualitas dan relevan sangat penting untuk efektivitas model. Pemilihan fitur yang tepat juga krusial untuk meningkatkan akurasi prediksi serangan. Selain itu, penelitian ini menggunakan tiga metode *Naive Bayes* untuk memodelkan data serangan siber dan mengoptimalkan model agar akurat dan andal.

Penelitian ini merumuskan masalah sebagai berikut: bagaimana implementasi Algoritma *Naive Bayes* dalam klasifikasi serangan jaringan, bagaimana hasil pemodelan dengan Algoritma *Naive Bayes* untuk klasifikasi serangan jaringan, dan bagaimana hasil pengujian model terhadap data uji. Untuk lebih memfokuskan penelitian ini, beberapa batasan diterapkan, yaitu hanya menggunakan Algoritma *Naive Bayes* (*GaussianNB*, *BernoulliNB*, dan *MultinomialNB*) untuk klasifikasi serangan jaringan, dataset dibagi menjadi 80% data latih dan 20% data uji, seleksi fitur dilakukan menggunakan metode ANOVA, dan validasi model menggunakan *cross-validation* dengan k = 10.

Tujuan dari penelitian ini adalah mengimplementasikan Algoritma *Naive Bayes* untuk klasifikasi serangan jaringan, menentukan fitur-fitur terbaik dalam klasifikasi serangan jaringan, dan mengevaluasi performa model *Naive Bayes* pada data uji. Penelitian ini diharapkan memberikan manfaat berupa solusi efektif dalam deteksi dan klasifikasi serangan jaringan menggunakan Algoritma *Naive Bayes*, wawasan tentang fitur penting dalam klasifikasi serangan jaringan, dan model klasifikasi yang dapat digunakan oleh praktisi keamanan jaringan untuk meningkatkan keamanan sistem mereka.

Metodologi penelitian ini melibatkan beberapa tahap, yaitu pengumpulan data serangan jaringan yang relevan, pra-pemrosesan data dengan membersihkan data dan melakukan seleksi fitur menggunakan ANOVA, pemodelan dengan mengimplementasikan Algoritma *Naive Bayes* (*GaussianNB*, *BernoulliNB*, dan *MultinomialNB*) dan melakukan pemodelan dengan pembagian data latih dan data uji, validasi model menggunakan *cross-validation* dengan k = 10, pengujian model terbaik pada data uji dan evaluasi performa model menggunakan metrik akurasi, presisi, *recall*, dan *F1-score*, serta analisis hasil pemodelan dan pengujian untuk memberikan kesimpulan dan saran.

2. METODOLOGI PENELITIAN

2.1 Pengumpulan Data

Tahapan pertama adalah mengumpulkan data dari log sistem keamanan jaringan yang mencatat aktivitas penyerangan terhadap beberapa klien dalam jaringan perusahaan. Data ini mencakup informasi penting yang diperlukan untuk mendeteksi dan menganalisis serangan siber.

Data yang digunakan dalam penelitian ini adalah log dari *firewall Fortigate*, khususnya dari sistem Pencegahan Intrusi (IPS) yang mencatat aktivitas ancaman atau serangan terhadap jaringan

Biner : Jurnal Ilmu Komputer, Teknik dan Multimedia
Volume 2, No. 3, Agustus 2024
ISSN 2988-3814 (media online)
Hal 243-253

PT Nusa Network Prakarsa. Data ini dikumpulkan selama periode Januari hingga Februari 2024 untuk memberikan gambaran komprehensif mengenai aktivitas serangan dalam dua bulan tersebut.

Log ini mencatat berbagai jenis serangan yang berhasil dideteksi dan dicegah oleh *firewall*, mencakup informasi tentang alamat IP sumber serangan, alamat IP tujuan (klien yang diserang), tingkat keparahan serangan, port sumber dan tujuan, negara asal serangan, jenis layanan yang diserang, dan jenis *malware* atau virus yang digunakan. Jumlah total data yang dianalisis adalah 43.184 entri, dengan setiap entri merepresentasikan satu percobaan serangan yang terdeteksi.

Variabel kunci dalam data ini meliputi:

a. *Source IP* : Alamat *IP* dari *attacker* yang mencoba melakukan serangan.

b. *Destination IP* : Alamat *IP* dari klien yang menjadi target serangan.

c. *Severity* : Tingkat keparahan dari serangan yang terdeteksi.

d. *Source Port* : *Port* sumber yang digunakan oleh *attacker* untuk mengirim serangan.

e. *Destination Port* : *Port* tujuan yang menjadi target serangan.

f. Negara : Negara asal dari serangan yang terdeteksi.

g. *Service* : Jenis layanan yang menjadi target serangan, seperti HTTP, FTP, dll.

h. *Name Malware/Virus* : Nama *malware* atau virus yang digunakan dalam serangan.

i. *Action* : Tindakan yang diambil oleh *firewall*, yaitu *deny*, *close*, *dropped*, *ip-conn*, *server-rst*, *client-rs*, *dns*, dan *timeout*.

2.2 *Preprocessing data*

Pada tahap ini, data yang telah dikumpulkan akan diproses lebih lanjut untuk memastikan kebersihannya dan relevansi. Proses ini melibatkan penghapusan atau pengisian nilai yang hilang dan melakukan seleksi fitur menggunakan teknik seperti ANOVA atau *mutual information*. Seleksi fitur bertujuan untuk memilih fitur-fitur yang paling relevan sehingga dapat meningkatkan kinerja model prediksi.

Seleksi fitur dilakukan dengan metode ANOVA, yang menghitung nilai statistik untuk setiap fitur dalam dataset. Nilai statistik ANOVA yang tinggi menunjukkan adanya perbedaan signifikan antara kelompok, sehingga fitur-fitur tersebut dipilih untuk analisis lebih lanjut atau pembangunan model. Metode ini efektif dalam mengidentifikasi fitur-fitur paling informatif, meningkatkan efisiensi dan akurasi model yang dikembangkan (Kurniawan, 2023).

2.3 Pemodelan

Setelah data siap, tahap berikutnya adalah pemodelan. Dalam penelitian ini, tiga jenis *Naive Bayes* (*Gaussian*, *Bernoulli*, dan *Multinomial*) akan diterapkan untuk memprediksi tindakan berdasarkan data serangan. *Cross Validation* dengan k = 10 akan digunakan untuk membagi data menjadi 10 subset berbeda, di mana setiap subset digunakan sebagai data uji secara bergantian. *GridSearchCV* akan digunakan untuk menemukan kombinasi parameter terbaik yang menghasilkan model optimal.

Gaussian Naïve Bayes memiliki fitur utama yang dapat menangani data kontinu. Model ini menggunakan distribusi normal untuk proses klasifikasi, memungkinkan penyesuaian probabilitas di seluruh dataset. Dengan ini, *Gaussian Naïve Bayes* dapat menentukan distribusi probabilitas kondisional untuk setiap kelas berdasarkan nilai kontinu dari Parameter Gerakan Tanah yang diberikan (Cataldi et al., 2021).

$$P(x|c) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \quad (1)$$

Biner : Jurnal Ilmu Komputer, Teknik dan Multimedia
Volume 2, No. 3, Agustus 2024
ISSN 2988-3814 (media online)
Hal 243-253

Dimana, $P(x|c)$ adalah probabilitas fitur xx diberikan kelas c, yang dihitung menggunakan distribusi normal (*Gaussian*). $P(c)$ adalah probabilitas prior dari kelas c. $P(x)$ adalah probabilitas a priori dari fitur x (sering diabaikan karena konstan untuk semua kelas). μ adalah rata-rata (*mean*) dari fitur x dalam kelas c. σ adalah standar deviasi dari fitur x dalam kelas c.

Model *Multinomial* dalam algoritma *Naïve Bayes* sangat cocok untuk data diskrit dan umum digunakan dalam klasifikasi teks atau dokumen. Model ini memperhitungkan berapa kali setiap kata muncul dalam dokumen tertentu (Ashari et al., 2020).

$$P(x|c) = \frac{(\sum_{i-1}^{n} x_i)!}{\prod_{i-1}^{n} x_i!} \prod\nolimits_{i=1}^{n} P(x_i|c)^{x_i} \quad (2)$$

Dimana, $P(x|c)$ adalah probabilitas dokumen x dalam kelas c, dihitung berdasarkan frekuensi kata. $P(c)$ adalah probabilitas priori dari kelas c. $P(x)$ adalah probabilitas a priori dari fitur x (sering diabaikan karena konstan untuk semua kelas). x_i adalah jumlah kemunculan kata i dalam dokumen x. $P(x|c)$ adalah probabilitas kata ii muncul dalam dokumen dari kelas c.

Model *Bernoulli* dalam algoritma *Naïve Bayes* Classifier adalah jenis model klasifikasi yang cocok untuk memperhitungkan keberadaan atau ketiadaan istilah dalam data, bukan seberapa sering istilah tersebut muncul (Ashari et al., 2020).

$$P(x|c) = \prod\nolimits_{i=1}^{n} P(x_i|c)^{x_i}.(1 - P(x_i|c))^{(1-x_i)} \quad (3)$$

Dimana, $P(x|c)$ adalah probabilitas bahwa dokumen xx muncul di kelas c, dihitung berdasarkan kehadiran atau ketidakhadiran fitur. $P(c)$ adalah probabilitas prior dari kelas c. $P(x)$ adalah probabilitas prior dari fitur x (sering diabaikan karena konstan untuk semua kelas). x_i adalah fitur biner yang menunjukkan apakah kata ii hadir dalam dokumen x (1 jika hadir, 0 jika tidak). $P(x_i|c)$ adalah probabilitas kata i muncul dalam dokumen dari kelas c.

Cross Validation adalah metode penting dalam statistik yang digunakan untuk mengevaluasi kinerja model atau algoritma secara menyeluruh. Teknik ini melibatkan pembagian data menjadi beberapa subset yang disebut *folds*. Misalnya, dalam 5-*fold Cross Validation*, data dibagi menjadi lima bagian. Setiap bagian ini secara bergantian digunakan sebagai data uji, sementara bagian lainnya digunakan untuk melatih model. Proses ini diulang sebanyak jumlah *fold* yang ada, sehingga model dilatih dan diuji beberapa kali. Hasil evaluasi dari setiap *fold* dirata-ratakan untuk mendapatkan gambaran yang lebih akurat mengenai efektivitas model secara keseluruhan. Teknik ini sangat efektif dalam mengurangi bias penilaian dan memberikan estimasi yang lebih terpercaya tentang kemampuan model dalam memproses data baru (Fuadah et al., 2022).

2.4 Pengujian

Tahap ini melibatkan pengujian model yang telah dibangun dengan menggunakan 20% data uji yang telah disisihkan sebelumnya. Pengujian bertujuan untuk mengevaluasi performa model pada data baru. Evaluasi dilakukan menggunakan *confusion matrix* untuk menghitung metrik akurasi, presisi, *recall*, dan *F1-score*.

Confusion matrix adalah alat yang meringkas kinerja model klasifikasi dalam bentuk matriks n x n, di mana n adalah jumlah kelas yang berbeda. Matriks ini membandingkan prediksi klasifikasi dengan hasil aktual, memberikan wawasan mendalam tentang kinerja model dan jenis kesalahan yang terjadi. Contohnya, TP (*True Positive*) dan TN (*True Negative*) mewakili prediksi yang benar, sedangkan FP (*False Positive*) dan FN (*False Negative*) menunjukkan kesalahan prediksi. *Confusion matrix* ini mendasari perhitungan metrik evaluasi seperti akurasi, presisi, *recall*, dan *F1-score*, yang secara keseluruhan menggambarkan performa model. Dengan menggunakan *confusion matrix*, peneliti dan praktisi dapat mengidentifikasi area yang perlu ditingkatkan, memungkinkan penyesuaian dan optimasi model untuk mencapai hasil yang lebih akurat dan andal (Ernianti Hasibuan & Heriyanto, 2022).

Biner : Jurnal Ilmu Komputer, Teknik dan Multimedia
Volume 2, No. 3, Agustus 2024
ISSN 2988-3814 (media online)
Hal 243-253

$$Akurasi = \frac{TP + TN}{TP + TN + FP + FN} \quad (4)$$

$$Presisi = \frac{TP}{TP + FP} \quad (5)$$

$$Recall = \frac{TP}{TP + FN} \quad (6)$$

$$F1 - Score = 2x\frac{Presisi\ x\ Recall}{Presisi + Recall} \quad (7)$$

2.5 Hasil

Hasil dari pengujian akan dievaluasi dengan menggunakan metrik akurasi, presisi, *recall*, dan *F1-score*. Penelitian ini akan membandingkan hasil akurasi dari data pelatihan dengan akurasi pada data uji untuk menilai kemampuan generalisasi model. Analisis ini akan membantu menentukan apakah model dapat diandalkan untuk aplikasi praktis dalam mendeteksi dan mengklasifikasikan serangan siber.

3. ANALISA DAN PEMBAHASAN

Pada bagian ini berisi analisa, hasil serta pembahasan dari topik penelitian, yang bisa di buat terlebih dahulu metodologi penelitian. Bagian ini juga merepresentasikan penjelasan yang berupa penjelasan, gambar, tabel dan lainnya.

3.1 *Preprocessing*

Proses *preprocessing data* dilakukan untuk mempersiapkan data sebelum tahap pemodelan, yang meliputi *Data Cleaning*, penghapusan data kosong, dan *encoding*.

a. *Data Cleaning*

Data Cleaning adalah langkah penting dalam *preprocessing* untuk memastikan data bersih, konsisten, dan bebas dari kesalahan yang dapat mengganggu analisis. Langkah pertama adalah mengidentifikasi kolom-kolom bertipe data objek, yang biasanya berisi data teks atau kategorikal yang perlu dibersihkan. Transformasi data dilakukan dengan mengubah semua huruf dalam kolom objek menjadi huruf kecil untuk konsistensi, menghapus spasi di awal dan akhir string, mengganti spasi ganda di antara kata-kata dengan satu spasi tunggal, serta menghapus tanda kutip dalam data teks.

b. *Data encoding*

Data encoding diperlukan karena data kategorikal sering tidak dapat digunakan langsung dalam algoritma *Machine Learning* dan perlu diubah menjadi format numerik. Langkah pertama dalam proses ini adalah mengidentifikasi kolom objek selain kolom *"Action"*, yang merupakan target variabel. Selanjutnya, dilakukan implementasi *LabelEncoder* dari *scikit-learn* untuk memberikan label numerik unik pada setiap nilai dalam kolom kategorikal. Misalnya, setiap alamat *IP* unik di kolom *"Source IP"* diubah menjadi nilai numerik. Nilai hasil *encoding* ditambahkan dengan 1 untuk menghindari nilai nol dalam data. Proses-proses ini memastikan bahwa data siap digunakan dalam analisis dan pemodelan *Machine Learning* dengan meminimalkan kesalahan dan ketidakkonsistenan.

3.2 Pemodelan *Naive Bayes*

Pada subbab ini, akan dijelaskan proses pembuatan model menggunakan data yang telah dikumpulkan dan dilakukan *preprocessing*. Pemodelan dilakukan dengan menerapkan tiga jenis algoritma *Naive Bayes*: *Gaussian Naive Bayes*, *Bernoulli Naive Bayes*, dan *Multinomial Naive Bayes*. Setiap model akan dilatih dan divalidasi menggunakan data yang telah dibagi menjadi data pelatihan dan data uji. Proses pelatihan mencakup pemilihan parameter optimal menggunakan *GridSearchCV* dan penerapan teknik *cross-validation* untuk memastikan model tidak *overfitting*

Biner : Jurnal Ilmu Komputer, Teknik dan Multimedia
Volume 2, No. 3, Agustus 2024
ISSN 2988-3814 (media online)
Hal 243-253

serta dapat menggeneralisasi dengan baik pada data baru. Hasil dari setiap model akan dicatat untuk dibandingkan pada tahap evaluasi. Langkah awal dalam proses pemodelan melibatkan pemisahan data menjadi dua set utama: data latih dan data uji. Dalam penelitian ini, data latih digunakan sebanyak 80% dari total dataset, sedangkan sisa 20% digunakan sebagai data uji. Berikut hasil pemodelan yang diperoleh.

Tabel 1. Hasil Pemodelan

Indeks	*model*	*best_params*	*accuracy*	*selected_features*	*best_model*
12	*GaussianNB*	*{}*	*0.709579*	*Index([“Source IP”, “Destination IP”, “Severity”, “Destination Port”, “Service”, “Name Malware/Virus”], dtype=“object”)*	*GaussianNB()*
9	*GaussianNB*	*{}*	*0.654879*	*Index([“Source IP”, “Destination IP”, “Severity”, “Service”, “Name Malware/Virus”], dtype=“object”)*	*GaussianNB()*
6	*GaussianNB*	*{}*	*0.630454*	*Index([“Source IP”, “Severity”, “Service”, “Name Malware/Virus”], dtype=“object”)*	*GaussianNB()*
0	*GaussianNB*	*{}*	*0.624221*	*Index([“Service”, “Name Malware/Virus”], dtype=“object”)*	*GaussianNB()*
3	*GaussianNB*	*{}*	*0.61926*	*Index([“Severity”, “Service”, “Name Malware/Virus”], dtype=“object”)*	*GaussianNB()*
5	*MultinomialNB*	*{“alpha”: 1e-10}*	*0.48836*	*Index([“Severity”, “Service”, “Name Malware/Virus”], dtype=“object”)*	*MultinomialNB(alpha=1e-10)*
11	*MultinomialNB*	*{“alpha”: 1e-10}*	*0.385193*	*Index([“Source IP”, “Destination IP”, “Severity”, “Service”, “Name Malware/Virus”], dtype=“object”)*	*MultinomialNB(alpha=1e-10)*

Biner : Jurnal Ilmu Komputer, Teknik dan Multimedia
Volume 2, No. 3, Agustus 2024
ISSN 2988-3814 (media online)
Hal 243-253

Indeks	*model*	*best_params*	*accuracy*	*selected_features*	*best_model*
8	*MultinomialNB*	*{"alpha": 1e-10}*	*0.375525*	*Index(["Source IP", "Severity", "Service", "Name Malware/Virus"], dtype="object")*	*MultinomialNB(alpha=1e-10)*
2	*MultinomialNB*	*{"alpha": 1e-10}*	*0.328203*	*Index(["Service", "Name Malware/Virus"], dtype="object")*	*MultinomialNB(alpha=1e-10)*
1	*BernoulliNB*	*{"alpha": 1e-10}*	*0.294492*	*Index(["Service", "Name Malware/Virus"], dtype="object")*	*BernoulliNB(alpha=1e-10)*
4	*BernoulliNB*	*{"alpha": 1e-10}*	*0.294492*	*Index(["Severity", "Service", "Name Malware/Virus"], dtype="object")*	*BernoulliNB(alpha=1e-10)*
7	*BernoulliNB*	*{"alpha": 1e-10}*	*0.294492*	*Index(["Source IP", "Severity", "Service", "Name Malware/Virus"], dtype="object")*	*BernoulliNB(alpha=1e-10)*
10	*BernoulliNB*	*{"alpha": 1e-10}*	*0.294492*	*Index(["Source IP", "Destination IP", "Severity", "Service", "Name Malware/Virus"], dtype="object")*	*BernoulliNB(alpha=1e-10)*
13	*BernoulliNB*	*{"alpha": 1e-10}*	*0.294492*	*Index(["Source IP", "Destination IP", "Severity", "Destination Port", "Service", "Name Malware/Virus"], dtype="object")*	*BernoulliNB(alpha=1e-10)*
14	*MultinomialNB*	*{"alpha": 1e-10}*	*0.234067*	*Index(["Source IP", "Destination IP", "Severity", "Destination Port", "Service", "Name Malware/Virus"], dtype="object")*	*MultinomialNB(alpha=1e-10)*

Biner : Jurnal Ilmu Komputer, Teknik dan Multimedia
Volume 2, No. 3, Agustus 2024
ISSN 2988-3814 (media online)
Hal 243-253

Hasil dari pemodelan menunjukkan beberapa temuan signifikan terkait kinerja dan pemilihan model terbaik. Pada penggunaan *Gaussian Naive Bayes* (*GaussianNB*), model ini menghasilkan akurasi tertinggi sebesar 70,96% dengan penggunaan enam fitur utama, yaitu *"Source IP"*, *"Destination IP"*, *"Severity"*, *"Destination Port"*, *"Service"*, dan *"Name Malware/Virus"*. Evaluasi menggunakan teknik *cross-validation* menunjukkan konsistensi yang baik dengan nilai rata-rata akurasi sekitar 70,74% dan standar deviasi 0,47%, menandakan stabilitas model dalam berbagai subset data.

Sebaliknya, model *Multinomial Naive Bayes* (*MultinomialNB*) dan *Bernoulli Naive Bayes* (*BernoulliNB*) menunjukkan performa yang kurang memuaskan dibandingkan *GaussianNB*. *MultinomialNB*, yang menggunakan tiga fitur terbaik (*"Severity"*, *"Service"*, *dan "Name Malware/Virus"*), memperoleh akurasi sekitar 48,84%. Sementara itu, *BernoulliNB* dengan dua fitur terbaik (*"Service" dan "Name Malware/Virus"*) hanya mencapai akurasi sekitar 29,45%. Kedua model ini menunjukkan hasil yang lebih rendah, bahkan setelah dilakukan optimasi parameter dengan *GridSearchCV*.

Kesimpulannya, *GaussianNB* dengan enam fitur terbaik terbukti sebagai model yang paling optimal dalam penelitian ini, dengan akurasi yang lebih tinggi dan konsisten dibandingkan dengan model lainnya. Fitur yang dipilih mencakup elemen-elemen penting untuk klasifikasi, seperti alamat IP, tingkat keparahan serangan, dan layanan yang terlibat. Untuk penelitian di masa depan, disarankan untuk mengeksplorasi fitur tambahan atau metode pengolahan data yang lebih canggih guna meningkatkan kinerja model lebih lanjut.

4. IMPLEMENTASI

4.1 Evaluasi Model

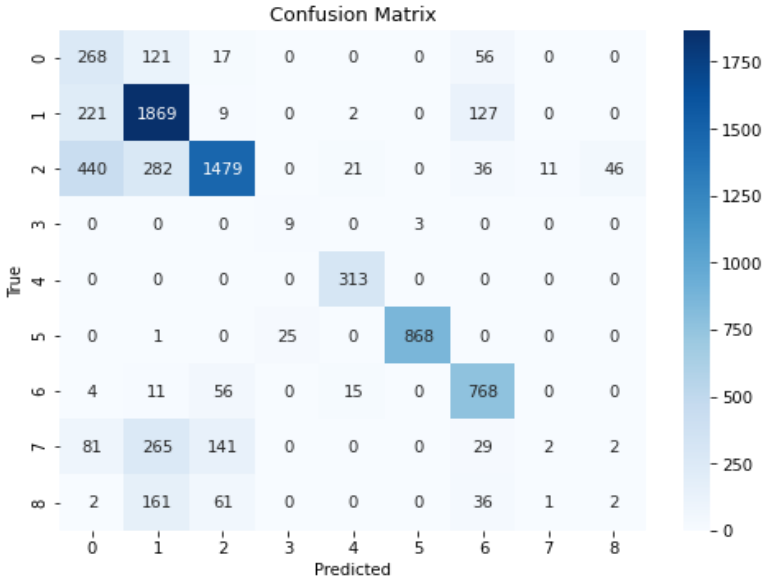
Pada subbab ini, kinerja model yang telah dibuat akan dievaluasi secara mendalam. Evaluasi dilakukan menggunakan data uji yang sebelumnya telah dipisahkan dari data pelatihan. Metrik evaluasi yang digunakan meliputi akurasi, presisi, *recall*, dan *F1-score*, yang dihitung berdasarkan *confusion matrix*. *Confusion matrix* akan memberikan gambaran mengenai performa model dalam mengklasifikasikan data serangan, termasuk jenis kesalahan yang dibuat seperti *True Positive*, *True Negative*, *False Positive*, dan *False Negative*. Analisis hasil evaluasi ini akan membantu dalam memahami seberapa baik model mampu mendeteksi dan mengklasifikasikan serangan jaringan, serta memberikan wawasan untuk perbaikan lebih lanjut jika diperlukan.

Pengujian dimulai dengan memilih model terbaik yang dihasilkan dari fase pemodelan sebelumnya. Model terbaik dipilih berdasarkan evaluasi menyeluruh menggunakan teknik *cross-validation* dan penyesuaian *hyperparameter* dengan *param_grid*. Setelah model terbaik ditentukan, langkah berikutnya adalah mengujinya dengan data uji yang tidak digunakan selama pelatihan.

Proses pengujian dimulai dengan menghitung kinerja model menggunakan *confusion matrix*, yang merupakan tabel yang membandingkan hasil prediksi model dengan nilai aktual dari data uji. *Confusion matrix* memungkinkan visualisasi jumlah prediksi yang benar dan salah serta jenis kesalahan yang terjadi.

Selanjutnya, kinerja model diukur dengan beberapa metrik evaluasi, yaitu akurasi, presisi, *recall*, dan *F1-score*. Akurasi menunjukkan seberapa tepat model dalam memprediksi kelas data uji. Presisi mengukur proporsi prediksi positif yang benar dibandingkan dengan semua prediksi positif yang dilakukan model. *Recall*, atau sensitivitas, mengukur proporsi kelas positif yang berhasil diprediksi model dari semua *instance* yang benar-benar positif. *F1-score*, sebagai rata-rata harmonis dari presisi dan *recall*, memberikan gambaran menyeluruh mengenai kemampuan model dalam memprediksi kedua kelas dengan seimbang. Berikut hasil *confusion matrix* yang diperoleh pada pengujian.

Biner : Jurnal Ilmu Komputer, Teknik dan Multimedia
Volume 2, No. 3, Agustus 2024
ISSN 2988-3814 (media online)
Hal 243-253



Gambar 1. *Confusion Matrix* Hasil Pengujian

Hasil dari *confusion matrix* memberikan wawasan mendalam tentang performa model klasifikasi pada setiap kelas *output* yang diuji. Untuk kelas *"deny"*, model menunjukkan hasil yang sangat baik dengan 268 prediksi yang tepat. Hal ini mencerminkan akurasi, presisi, *recall*, dan *F1-score* yang memuaskan. Presisi menunjukkan bahwa semua prediksi *"deny"* benar-benar termasuk dalam kelas tersebut, sedangkan *recall* menunjukkan bahwa model dapat mengidentifikasi sebagian besar kasus *"deny"* yang sebenarnya ada. *F1-score*, yang menggabungkan presisi dan *recall*, memberikan gambaran menyeluruh tentang kinerja model untuk kelas ini.

Pada kelas *"close"*, model juga menunjukkan performa yang solid dengan 1869 prediksi yang benar. Tingkat presisi dan *recall* yang tinggi menunjukkan bahwa model dapat secara akurat mengklasifikasikan sebagian besar kasus *"close"*, baik dalam hal prediksi yang benar maupun kemampuan untuk mengenali sebagian besar kasus *"close"* yang nyata.

Untuk kelas *"dropped"*, model berhasil memprediksi 1479 kasus dengan akurat, menunjukkan keseimbangan yang baik antara presisi dan *recall*. *F1-score* untuk kelas ini memberikan gambaran keseluruhan mengenai seberapa efektif model dalam mengklasifikasikan kasus *"dropped"*, dengan mempertimbangkan baik presisi maupun *recall*. Detail pengukuran akurasi, presisi, *recall*, dan *F1-score* untuk setiap kelas ditampilkan pada gambar di bawah ini.

Hasil evaluasi model terbaik pada data uji:

	precision	recall	f1-score	support
client-rst	0.26	0.58	0.36	462
close	0.69	0.84	0.76	2228
deny	0.84	0.64	0.73	2315
detected	0.26	0.75	0.39	12
dns	0.89	1.00	0.94	313
dropped	1.00	0.97	0.98	894
ip-conn	0.73	0.90	0.81	854
server-rst	0.14	0.00	0.01	520
timeout	0.04	0.01	0.01	263
accuracy			0.71	7861
macro avg	0.54	0.63	0.55	7861
weighted avg	0.70	0.71	0.69	7861

Gambar 2. *Classification Repport* Hasil Pengujian

Biner : Jurnal Ilmu Komputer, Teknik dan Multimedia
Volume 2, No. 3, Agustus 2024
ISSN 2988-3814 (media online)
Hal 243-253

Evaluasi model terbaik pada data uji menunjukkan kinerja yang bervariasi di setiap kelas *output*. Kelas *"dropped"* menunjukkan performa sangat baik dengan *precision*, *recall*, dan *F1-score* masing-masing sebesar 1.00, 0.97, dan 0.98. Ini menandakan bahwa model hampir sempurna dalam mengklasifikasikan kasus yang benar-benar masuk dalam kategori *"dropped"*. Kelas *"dns"* juga menunjukkan hasil positif dengan *precision* mencapai 0.89, yang berarti sebagian besar prediksi sebagai *"dns"* oleh model benar-benar sesuai dengan kelas tersebut.

Namun, ada beberapa kelas seperti *"server-rst"* dan *"timeout"* yang menunjukkan kinerja jauh lebih rendah. Kelas *"server-rst"* memiliki *precision* yang sangat rendah pada 0.14, menunjukkan bahwa banyak prediksi yang dikategorikan sebagai *"server-rst"* oleh model ternyata salah. Kelas *"timeout"* juga memiliki *precision* dan *recall* yang sangat rendah, yaitu 0.04 dan 0.01, menunjukkan bahwa model hampir tidak efektif dalam mengidentifikasi atau mengklasifikasikan kasus yang benar-benar berada dalam kategori *"timeout"*.

Secara keseluruhan, model mencapai akurasi sebesar 0.71, menunjukkan bahwa sekitar 71% dari semua prediksi model adalah benar. Meskipun hasil ini cukup baik, peningkatan performa model harus fokus pada kelas-kelas dengan *precision* dan *recall* rendah seperti *"server-rst"* dan *"timeout"*. Analisis ini memberikan wawasan penting untuk pengembangan lebih lanjut dari model klasifikasi agar dapat mencapai hasil yang lebih baik dalam situasi nyata dan aplikasi praktis.

4.2 Hasil Analisis

Dari keseluruhan proses yang dilakukan, beberapa poin penting dapat disoroti dari hasil pemodelan dan pengujian model terbaik. *Gaussian Naive Bayes* dengan k = 6 dipilih sebagai model terbaik, dengan akurasi mencapai 0.709. Fitur-fitur utama yang dipilih untuk model ini meliputi *"Source IP"*, *"Destination IP"*, "Severity", "Destination Port", "Service", dan "Name *Malware*/Virus".

Nilai akurasi ini diperoleh setelah model diuji pada data uji, dengan hasil akurasi mencapai 0.71. Meskipun terdapat variasi dalam performa antar kelas, secara keseluruhan model memberikan prediksi yang cukup baik, bahkan sedikit lebih tinggi dari rata-rata akurasi *cross-validation* yang dicapai selama proses pemodelan (0.707).

Analisis lebih lanjut tentang hubungan antara akurasi model dan hasil pengujian menunjukkan bahwa model mampu mempertahankan performa yang stabil ketika diuji dengan data baru. Namun, terdapat beberapa kelas seperti *"server-rst"* dan *"timeout"* yang menunjukkan performa yang perlu ditingkatkan, dengan nilai *precision* dan *recall* yang rendah. Hal ini mengindikasikan bahwa model mengalami kesulitan dalam mengklasifikasikan kasus-kasus yang sebenarnya termasuk dalam kelas-kelas tersebut.

Secara keseluruhan, hasil analisis ini memberikan pemahaman yang jelas tentang kekuatan dan kelemahan model dalam menghadapi dataset yang digunakan. Dengan mempertimbangkan nilai akurasi, seleksi fitur, serta hasil pengujian yang diperoleh, langkah selanjutnya bisa difokuskan pada peningkatan performa model terutama untuk kelas-kelas dengan performa prediksi yang masih rendah. Dengan demikian, model dapat dioptimalkan lebih lanjut untuk memberikan prediksi yang lebih akurat dan dapat diandalkan dalam aplikasi praktisnya. Hal ini akan membantu meningkatkan keefektifan dan efisiensi sistem yang bergantung pada model prediktif tersebut.

5. KESIMPULAN

Berdasarkan penerapan Algoritma *Naive Bayes* untuk klasifikasi serangan jaringan, hasil pemodelan menunjukkan bahwa model ini memiliki kemampuan yang baik dalam mengenali dan mengklasifikasikan berbagai jenis serangan dengan tingkat akurasi yang cukup memuaskan. Pemilihan dan seleksi fitur yang tepat berperan besar dalam meningkatkan kinerja model, dengan *Gaussian Naive Bayes* muncul sebagai model paling unggul, mencapai akurasi sebesar 0.709 pada data uji.

Selama pengujian model, ditemukan bahwa meskipun ada beberapa kategori serangan yang memerlukan peningkatan performa, secara keseluruhan model *Naive Bayes* menunjukkan hasil yang

memuaskan dengan akurasi total sebesar 0.71. Hasil ini memberikan keyakinan bahwa model tersebut dapat digunakan secara efektif dalam mendeteksi serangan jaringan di lingkungan praktis, memberikan perlindungan yang andal terhadap ancaman keamanan jaringan.

REFERENCES

Anas, I., & Zakir, S. (2024). Artificial Intelligence: Solusi Pembelajaran Era Digital 5.0. *Jurnal Sains Komputer & Informatika (J-SAKTI*, *8*(1), 35–46.

Ashari, H., Arifianto, D., Azizah, H., & Faruq, A. (2020). Kinerja Algoritma *Multinomial Naïve Bayes* (Mnb), Multivariate *Bernoulli* Dan Rocchio Algorithm Dalam Klasifikasi Konten Berita Hoax Berbahasa Indonesia Dengan Jupyter Notebook. *Jurnal Aplikasi Sistem Informasi Dan Elektronika*, *2*(2), 52–65.

Cataldi, L., Tiberi, L., & Costa, G. (2021). Estimation of MCS intensity for Italy from high quality accelerometric data, using GMICEs and *Gaussian Naïve Bayes* Classifiers. *Bulletin of Earthquake Engineering*, *19*(6), 2325–2342. https://doi.org/10.1007/s10518-021-01064-6

CISCO. (2023). *Security Outcomes Report - Achieving Security Resilience*. Cisco. https://www.cisco.com/c/en/us/products/security/cybersecurity-reports.html

Ernianti Hasibuan, & Heriyanto, E. A. (2022). Analisis Sentimen Pada Ulasan Aplikasi Amazon Shopping Di Google Play Store Menggunakan *Naive Bayes* Classifier. *Jurnal Teknik Dan Science*, *1*(3), 13–24. https://doi.org/10.56127/jts.v1i3.434

Fatkhurohman, A., & Pujastuti, E. (2019). Penerapan Algoritma *Naïve Bayes* Classifier Untuk Meningkatkan Keamanan Data Dari Website Phising. *Respati*, *14*(1). https://doi.org/10.35842/jtir.v14i1.279

Fuadah, Y. N., Ubaidullah, I. D., Ibrahim, N., Taliningsing, F. F., Sy, N. K., & Pramuditho, M. A. (2022). Optimasi Convolutional Neural Network dan K-*Fold Cross Validation* pada Sistem Klasifikasi Glaukoma. *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, *10*(3), 728. https://doi.org/10.26760/elkomika.v10i3.728

Kurniawan, F. (2023). *Analisis Pengaruh Seleksi Fitur Anova Terhadap Performa Model Klasifikasi Gaussian Naïve Bayes Pada Dataset Pima Indians Diabetes*.

Putro, H. F., Vulandari, R. T., & Saptomo, W. L. Y. (2020). Penerapan Metode *Naive Bayes* Untuk Klasifikasi Pelanggan. *Jurnal Teknologi Informasi Dan Komunikasi (TIKomSiN)*, *8*(2). https://doi.org/10.30646/tikomsin.v8i2.500

Umar, N., & M. Adnan Nur. (2022). Application of *Naïve Bayes* Algorithm Variations On Indonesian General Analysis Dataset for Sentiment Analysis. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, *6*(4), 585–590. https://doi.org/10.29207/resti.v6i4.4179

Yati, J. D., Pamungkas, E. W., Kom, S., & Kom, M. (2023). Hate Speech Detection On Social Media Content In Javanese Language With *Naive Bayes* Algorithm. *Eprints.Ums.Ac.Id*. https://eprints.ums.ac.id/id/eprint/112433%0Ahttps://eprints.ums.ac.id/112433/5/NaskahPublikasi.pdf