**Journal on Education**
Volume 05, No. 03, Maret-April 2023, pp. 7925-7937
E-ISSN: 2654-5497, P-ISSN: 2655-1365
Website: http://jonedu.org/index.php/joe

# Penerapan Random Forest dan Adaboost untuk Klasifikasi Serangan DDoS

Ahmad Fauzi[1], Ema Utami[2], Anggit Dwi Hartanto[3]

[1,2,3], Jl. Ring Road Utara, Ngringin, Condongcatur, Kec. Depok, Kabupaten Sleman, Daerah Istimewa Yogyakarta
ahmad.1238@students.amikom.ac.id

***Abstract***

Among the different types of attacks in the field of Information Technology, DDOS attacks are one of the biggest threats to internet sites and pose a devastating risk to the security of computer systems, mainly due to their potential impact. Hence why research in this area is growing rapidly, with researchers focusing on new ways to address intrusion detection and prevention. Machine learning and Artificial Intelligence are some of the latest additions to the list of technologies studied to perform intrusion detection classification. This study explores the behavior and application of DDoS datasets for machine learning in the context of intrusion detection. The flow in this study, first is to collect raw DDoS datasets from reputable sources. After the data is obtained, the final data set is created for modeling. Data management involves data cleansing, data type transformation and data exchange on data collection. The selection process is accompanied by a model. Two separate algorithms, random and adaboost, are used to train a model with a dataset. The model is validated and retrained with a k-fold cross. The model was eventually evaluated using invisible data. The result is determined by various output sizes. In the experiment, DDoS datasets were used: CICDDoS_2019 The intrusion detection performance of this dataset was analyzed using two machine learning models. The dataset is divided in an 80:20 ratio for model training, validation and testing. Machine learning models are selected systematically and carefully to ensure that experiments are conducted in the right way. The results were analyzed using a set of performance metrics, including accuracy, precision, recall, f-measure, and compute time.
**Keywords:** IT Security, Ddos Attack, Machine Learning, Random Forest, Adaboost

**Abstrak**

Di antara berbagai jenis serangan di bidang Teknologi Informasi, serangan DDOS adalah salah satu ancaman terbesar bagi situs internet dan menimbulkan risiko yang menghancurkan keamanan sistem komputer, terutama karena potensi dampaknya. Oleh karena itu mengapa penelitian di bidang ini berkembang pesat, dengan para peneliti yang berfokus pada cara-cara baru untuk mengatasi deteksi dan pencegahan intrusi. Machine learning dan Artificial Intelligent adalah beberapa tambahan terbaru dalam daftar teknologi yang diteliti untuk melakukan klasifikasi deteksi intrusi. Studi ini mengeksplorasi perilaku dan penerapan dataset DDoS untuk pembelajaran mesin dalam konteks deteksi intrusi. Alur dalam penelitian ini, pertama adalah mengumpulkan dataset DDoS mentah dari sumber yang memiliki reputasi baik. Setelah data diperoleh, kumpulan data akhir dibuat untuk pemodelan. Manajemen data melibatkan pembersihan data, transformasi tipe data dan pertukaran data pada pengumpulan data. Proses seleksi disertai dengan model. Dua algoritma terpisah, random dan adaboost, digunakan untuk melatih model dengan dataset. Model divalidasi dan dilatih ulang dengan k-fold cross. Model tersebut akhirnya dievaluasi menggunakan data yang tidak terlihat. Hasilnya ditentukan oleh berbagai ukuran keluaran. Dalam percobaan, dataset DDoS digunakan: CICDDoS_2019 Performa deteksi intrusi set data ini dianalisis menggunakan dua model pembelajaran mesin. Dataset dibagi dalam rasio 80:20 untuk pelatihan model, validasi dan pengujian. Model pembelajaran mesin dipilih secara sistematis dan hati-hati untuk memastikan bahwa eksperimen dilakukan dengan cara yang tepat. Hasilnya dianalisis menggunakan sekumpulan metrik performa, termasuk akurasi, presisi, recall, f-measure, dan waktu komputasi.
**Kata Kunci:** Keamanan TI, Serangan Ddos, Pembelajaran Mesin, Hutan Acak, Adaboost



Corresponding author: Ahmad Fauzi
Email Address: 1238@students.amikom.ac.id (Jl. Ring Road Utara, Depok, Kab. Sleman, DI Yogyakarta)
Received 01 February 2023, Accepted 08 February 2023, Published 08 February 2023

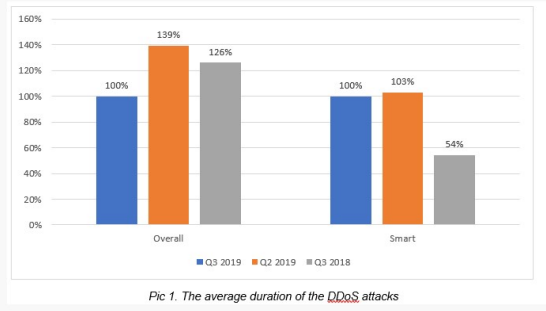
## PENDAHULUAN

Keamanan di bidang teknologi informasi merupakan aspek penting yang harus menjadi

prioritas setiap pengguna. Dengan terkoneksinya jaringan global di seluruh dunia menjadikan semakin luasnya jangkauan komunikasi antar pengguna di seluruh dunia, maka dalam keadaan tersebut bisa muncul ancaman-ancaman pada kondisi sumber daya di jaringan komputer pengguna. Contoh ancaman yang dikenal cukup sering dilakukan untuk menyerang jaringan komputer adalah DDoS (Distributed Denial of Service). Tujuan dari serangan DDoS adalah untuk menjatuhkan layanan target menggunakan berbagai sumber yang didistribusikan. Contoh tipikal dari serangan seperti itu adalah serangan berbasis flooding di mana korban kewalahan dengan jumlah besar lalu lintas jaringan yang diterima. Ide serangan DDoS berkisar pada fakta bahwa sejumlah besar sumber yang didistribusikan di beberapa lokasi digunakan untuk menargetkan korban.

Serangan DDoS menjadi salah satu serangan yang membahayakan, bahayanya serangan ini sudah terbukti pada pekan kedua Agustus 2011 di hongkong diberitakan oleh *cyber threat.id*. DDoS attack melumpuhkan server HKExnews.hk. serangan itu terjadi pada tanggal 12-13 Agustus 2011. Sejumlah perdagangan besar di pasar keuangan terbesar ketiga di Asia itu terpaksa ditangguhkan, termasuk HSBC, Cathay Pacific Airways dan HKEx sendiri yang memiliki nilai pasar gabungan sebesar HK$ 1,5 triliun.

Kemudian pada tahun 2019 menurut statistik yang dikumpulkan Kaspersky DDoS Protection pada tanggal 11 November 2019, Serangan DDoS naik 30% pada triwulan 3 2019 dibanding dengan triwulan sebelumnya, dan juga naik 32% jika dibandingkan dengan triwulan 3 2018. Perubahan drastis ini tak lepas dari lonjakan aktivitas DDoS di awal tahun ajaran, sebagian besar serangan DDoS (53%) terdeteksi pada bulan September 2019, Kaspersky mengungkapkan bahwa 60% dari serangan yang dicegah selama ini dilakukan terhadap sekolah dan situs jurnal elektronik.




Gambar 1. Durasi rata-rata serangan DDoS

Untuk mendeteksi dan mencegah potensi serangan, terdapat sebuah metode yang disebut dengan IDS (Intruder Detection System). IDS adalah sistem perangkat keras dan perangkat lunak yang memantau peristiwa yang terjadi pada komputer dan jaringan komputer untuk menganalisis masalah keamanan. IDS memiliki dua metode identifikasi, yaitu Rule Based (Signature Based) dan Behavior Based. Identifikasi Berbasis Signature Based dilakukan dengan menggunakan aturan administrator untuk menyeimbangkan lalu lintas jaringan dan menyimpannya dalam database. Jenis identifikasi ini memerlukan pembaruan aturan IDS. Tidak seperti Behavior Based, yang mendeteksi

serangan dengan membandingkan pola dalam kumpulan data menggunakan metode klasifikasi.

Umumnya, Behavior Based proses kerjanya menyamakan tren data atau peristiwa terkini, kemudian mengklasifikasikannya ke dalam metode dan membuat model. Dalam model yang dikembangkan, data pengujian diperiksa untuk menghasilkan output untuk melihat apakah trafik yang ada dapat dikategorikan sebagai intrusi atau tidak. Oleh karena itu, dalam hal ini diperlukan suatu metode yang digunakan untuk mencapai akurasi yang akurat dalam proses klasifikasi.

Pada penelitian sebelumnya, sudah dilakukan oleh Penelitian tersebut bermaksud untuk melakukan pengklasifikasian serangan DDoS menggunakan metode Naïve Bayes dan Super Vector Machine (SVM) untuk mengetahui tingkat akurasi dari masing-masing metode dengan menggunakan dataset ICSX 2012. Proses pengujian memanfaatkan pengambilan sampel dengan teknik random sampling dimana persentase data training 60% data testing 40% sehingga menghasilkan persentase model akurasi serta output berupa confusion matrix dan kurva ROC (Receiver Operating Characteristic). Dari hasil menganalisis perbandingan metode yang dihasilkan dari proses klasifikasi berdasarkan nilai akurasi confusion matrix, precision, recall, dan f1 score. Naive Bayes, SVM Linear, SVM Polynomial dan SVM Sigmoid menghasilkan persentase akurasi berturut-turut sebesar 85%, 99%, 99%, dan 99%. Persentase akurasi tertinggi diperoleh SVM Polynomial, sedangkan Naive Bayes menghasilkan persentase akurasi terendah.

## METODE

### *Dataset*

Untuk uji coba ini, CICDDoS_2019 berisi kumpulan data serangan DDoS awal dan baru-baru ini yang terlihat seperti data nyata (PCAP). Ini juga berisi CICFlowMeter-V32dan analisis lalu lintas jaringan. Untuk membuat profil perilaku manusia dan membangun lalu lintas latar naturalistik yang baik, sistem B-Profile digunakan. Dataset ini didasarkan pada protokol HTTP, HTTPS, FTP, SSH dan email dan memiliki perilaku abstrak dari 25 pengguna. Sejumlah lampiran DDoS refleksif modern disertakan dalam kumpulan data, termasuk Port Map, NetBIOS, LDAP, MSSQL, UDP, UDP-Lag, SYN, NTP, DNS dan SNMP.

Tabel 1. Spesifikasi OS dan IP Mesin untuk CICDDoS 2019. Diadaptasi dari DDoS Evaluation Set

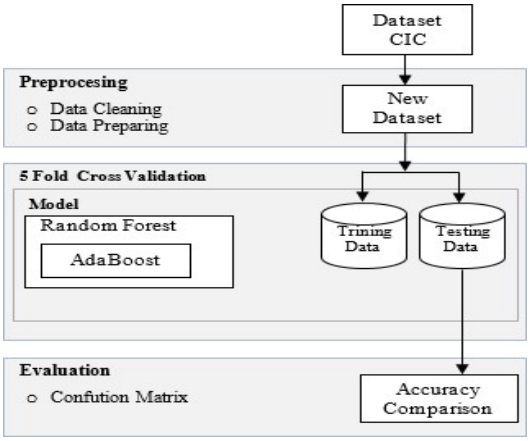
| Machine | OS | IPs |
|---|---|---|
| Server | Ubuntu 16.04 (Web Server) | 192.168.50.1 (first day) 192.168.50.4 (second day) |
| Firewall | Fortinet | 205.174.165.81 |
| PCs (first day) | Win 7<br>Win Vista<br>Win 8.1<br>Win 10 | 192.168.50.8<br>192.168.50.5<br>192.168.50.6<br>192.168.50.7 |
| PCs (second day) | Win 7<br>Win Vista<br>Win 8.1<br>Win 10 | 192.168.50.9<br>192.168.50.6<br>192.168.50.7<br>192.168.50.8 |

### *Implementasi dan Desain Eksperimen*

Bagian ini menjelaskan bagaimana solusi yang disarankan direncanakan dan dilaksanakan. Python digunakan untuk solusinya. Pertama, penjelasan diberikan mengenai solusi dan langkah-langkah implementasi didefinisikan secara singkat. Bagian kedua menjelaskan metode penyusunan data, pemberian informasi pesan data serta transformasi dan pembagian. Bagian ketiga menjelaskan pemodelan proses persiapan, validasi, dan pengujian dengan gambaran umum terperinci. Bagian 4 diakhiri dengan uraian proses penilaian dan ikhtisar temuan tinjauan keluaran kumpulan data DDoS.

**Gambaran Umum**

Diagram alir dari metode pembelajaran yang disurvei sebagai bagian dari solusi yang disarankan dalam penelitian ini disajikan pada Gambar 2. Yang pertama adalah mengumpulkan dataset DDoS mentah dari sumber yang memiliki reputasi baik. Setelah data diperoleh, kumpulan data akhir dibuat untuk pemodelan. Manajemen data melibatkan pembersihan data, transformasi tipe data dan pertukaran data pada pengumpulan data. Proses seleksi disertai dengan model. Dua algoritma terpisah, random forest dan adaboost, digunakan untuk melatih model dengan dataset. Model divalidasi dan dilatih ulang dengan k-fold cross. Model tersebut akhirnya dievaluasi menggunakan data yang tidak terlihat. Hasilnya ditentukan oleh berbagai ukuran keluaran.




Gambar 2 Alur Kerja Supervised Learning

**Preprocessing**

1. Data Cleaning dan Preparing

Penanganan data yang hilang dalam pembelajaran mesin itu penting karena dapat menyebabkan perkiraan yang salah dalam model apa pun. Untuk mengetahui apakah dataset tidak memiliki nilai. Ini diimplementasikan dengan metode pengisian perpustakaan pandas seperti yang ditunjukkan di bawah ini. dataset_ddos.isnull()

2. Data Tidak Terdefinisi

Penghapusan nilai nol dapat mengakibatkan data tidak terdefinisi. Bidang kosong tanpa sel di sebelah kirinya menjadi NaN setelah propagasi, karena tidak ada sel yang memberikan nilai. Akibatnya, nilai-nilai ini diterjemahkan menjadi 0. Maka jika ada data yang tidak memiliki nilai akan

diterjemahkan dengan metode filna. Ini semua dilakukan dengan menggunakan metode fillna [8]. dataset_ddos.filna(0, inplace=True)

3. Transformasi

Format data yang dikumpulkan mungkin tidak sesuai untuk pemodelan. Dalam kasus seperti itu, data dan tipe data perlu diubah sehingga data tersebut dapat dimasukkan ke dalam model. Karenanya, beberapa fitur data diubah menjadi numerik atau float, karena model tidak bekerja dengan baik dengan string, atau tidak berfungsi sama sekali.

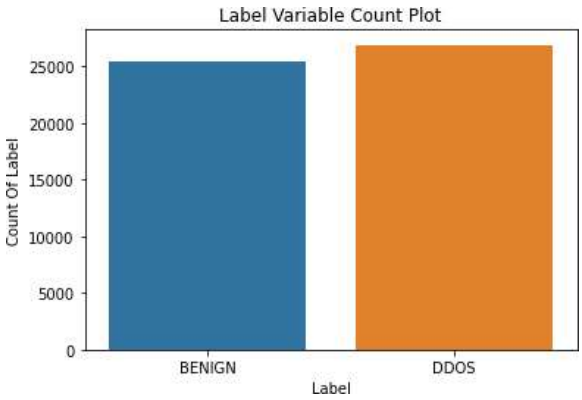
Label Kelas. Setiap instance set data mewakili cuplikan lalu lintas jaringan pada titik waktu tertentu. Instance ini diberi label menurut sifat lalu lintas, apakah lalu lintas itu jinak atau jahat untuk keperluan klasifikasi. Klasifikasi adalah biner, di mana lalu lintas jinak diberi label sebagai BENIGN, dan lalu lintas berbahaya diberi label sebagai DDoS. Tabel 5 merangkum sistem klasifikasi.

Table 2. Pelabelan Untuk Klasifikasi

| Label | Scenario |
|---|---|
| BENIGN | Trafik Jinak |
| DDoS | Trafik Berbahaya |

**New Dataset**

Pada dataset CICDDoS2019 yang diambil, terdapat 25.426 (48,6%) record yang diklasifikasikan sebagai traffic normal dan 26892 (51.4%) diklasifikasikan sebagai traffic berbahaya.




Gambar 3.Klasifikasi traffic

**Splitting Dataset**

Pada dasarnya, dua kumpulan data diperlukan dalam model, satu untuk pelatihan dan yang lainnya untuk pengujian. Data digunakan untuk melatih model sedangkan data uji digunakan untuk menilai generalisasi model yaitu keluaran model dengan data yang digali. Data tersebut digunakan untuk tujuan pelatihan. Divisi pelatihan / praktek dapat membuahkan hasil yang baik, namun terdapat banyak kelemahan dalam pendekatan ini. Karena pembagian adalah fenomena acak, ia dapat membentuk ketidakseimbangan antara set latihan dan tes, di mana ada beberapa contoh dalam satu kelas dalam set pelatihan.

Kumpulan data dibagi menjadi dua subset untuk memecahkan masalah ini; persiapan, validasi dan pengujian. Pemisahan ini dilakukan untuk persiapan, validasi, dan pengujian dengan perbandingan masing-masing 80:20. Untuk pemisahan, seperti yang ditunjukkan dalam fragmen kode berikut, metode helper split test train dari library scikit-learn digunakan. Pelatihan dilakukan dalam dua tahap yaitu pelatihan dan validasi set untuk metode ini. Selanjutnya, model dilatih di set pelatihan. Set validasi kemudian digunakan untuk menguji performa model pada data yang modelnya belum dilatih. Validasi metode k-fold telah dilakukan untuk keperluan analisis ini:

```
# Splite data
from sklearn.model_selection import train_test_split
X_train, X_test , y_train, y_test = train_test_split (X, y, test_size=0.2, random_state=0)
```

```
# Clasificasi
from sklearn.ensemble import RandomForestClassifier
rfc = RandomForestClassifier(criterion = "gini",
                             max_depth = 8,
                             min_samples_split = 10,
                             random_state = 5)
```

Gambar 4. Validasi Metode K-Fold

**Pemilihan Model**

Fase klasifikasi terdiri dari dua aspek;(1) konstruksi model pembelajaran, dan (2) pembuatan label yang diprediksi. Tugas-tugas ini diimplementasikan menggunakan sklearn, pustaka Python untuk penambangan data, analisis data, dan pembelajaran mesin.

Pada studi ini menampilkan pengujian dan pelatihan random forest dan adaboost. Model yang digunakan disini adalah Ran Random Forest merupakan metode pembelajaran mesin yang dapat digunakan untuk klasifikasi dan pembuatan prediksi. Dalam hal mendeteksi serangan DDoS, Random Forest dapat digunakan untuk mengklasifikasikan trafik jaringan sebagai normal atau serangan dengan menggunakan fitur-fitur yang relevan seperti jumlah paket, ukuran paket, dan waktu kedatangan.
Pohon individu dibangun menggunakan algoritma yang disajikan pada Tabel 3. Seperti disebutkan sebelumnya, algoritma ensemble memberikan akurasi yang lebih tinggi karena kombinasi beberapa model.

Tabel 3. Kode Pseudo Untuk Algoritma Hutan Acak

| **Algorithm Random Forest** |
|---|
| Memerlukan IDT (penginduksi pohon keputusan), T (jumlah iterasi), S (set pelatihan), μ (ukuran subsampel), N (jumlah atribut yang digunakan di setiap node) |
| |
| start |
| t ← 1 |
| repeat |
| _____ |
| St ← Sample μ instances from S with replacement. Build classifier Mt using IDT(N) on St |
| t++ |

| |
|---|
| until t > T |
| end |

**Training**

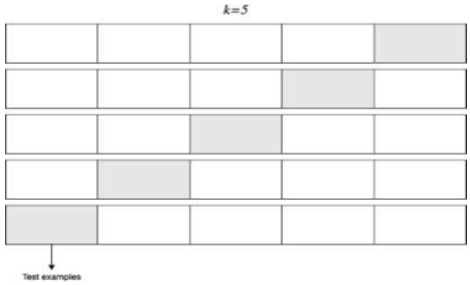
Selama proses pelatihan, algoritma yang dipilih dilengkapi dengan data pelatihan untuk dipelajari hingga akhirnya membuat model pembelajaran mesin. Karenanya, set pelatihan digunakan, Pada tahap ini dalam proses, sumber data masukan perlu disediakan dan harus berisi atribut target (label kelas). Proses pelatihan melibatkan menemukan pola dalam set pelatihan yang memetakan fitur masukan dengan atribut target. Berdasarkan pola yang diamati, dihasilkan model.

Dalam penelitian ini digunakan dataset CICDDoS2019 sebagai sumber data masukan, dimana atribut targetnya adalah jenis trafik jaringan yaitu attack atau normal. dua algoritma dilatih dengan dataset tersebut. Pelatihan dilakukan menggunakan beberapa metode dari pustaka scikit-learn. Tabel 4 memberikan rincian metode yang digunakan untuk setiap algoritma.

Tabel 4. Metode Dan Pengklasifikasi Dari Pustaka Python Scikit-Learn Digunakan Untuk Membangun Model.

| Model | Scikit-learn Methods & Classifiers |
|---|---|
| Random Forest | sklearn.ensemble.RandomForestClassifier [62 |
| Adaboost | sklearn.ensemble.AdaBoostClassifier [..] |

Model divalidasi dengan k-fold cross-validation setelah tahap pelatihan. Untuk menentukan generalisasi model, digunakan validasi silang. Pendekatan ini bertujuan untuk meminimalkan kesalahan overfitting ketika model terlalu cocok dengan data sampel yang berbeda. Validasi silang dilakukan dalam iterasi dan data dibagi menjadi k subset, disebut lipatan, per iterasi. Seperti yang ditunjukkan pada Gambar 11, model diperiksa pada lipatan k-1, dan lipatan lainnya disimpan. Ini diulangi sampai semua plat menjadi plat uji. Pengukuran penilaian diringkas dengan mengukur nilai rata-rata setelah metode selesai.




Gambar 5. K-Fold Cross Validation With 5 fold

Dalam studi ini, pendekatan k-fold bertingkat digunakan menggunakan dataset validasi (20% dari set global). K-fold bertingkat adalah variasi validasi k-fold cross yang memastikan bahwa distribusi kelas sama di semua lipatan. Ini diimplementasikan menggunakan metode StratifiedKFold

dari pustaka scikit-learn, dengan k = 5. Di bawah ini adalah potongan kode dari stratified k-fold, dimana n_splits menentukan jumlah lipatan.

```
for tr_in,val_in in StratifiedKFold(shuffle
True,n_splits=5).split(X_val,y_val):

{{model}}.fit(X_val.iloc[tr_in],y_val.iloc[tr_in])

accuracy.append(knn.score(X_val.iloc[val_in],y_val.iloc[val_in]))
```

Gambar 6. Potongan Kode

1. Testing

Pada tahap terakhir dari tahap pemodelan, model diuji dengan data yang tidak terlihat. Data ghaib yang digunakan pada tahap ini adalah hasil test set dari data split (20%). Pengujian dilakukan untuk menilai bagaimana model merepresentasikan data dan seberapa baik performanya di masa mendatang. Studi ini memastikan bahwa setiap penyesuaian model dilakukan sebelum pengujian, sehingga data pengujian hanya digunakan satu kali. Berbagai metrik performa dihasilkan untuk dapat menganalisis performa dataset DDoS, seperti akurasi, presisi, recall, dan F-measure. Ini dijelaskan di bagian selanjutnya.

2. Evaluasi

Bagian penting dalam memahami kinerja model adalah menghasilkan metrik kinerja. Dalam studi ini, berbagai metrik dihasilkan. Ini dijelaskan di bawah.

Accuracy. Salah satu cara untuk mendeskripsikan kinerja model klasifikasi adalah jumlah instance yang diklasifikasikan dengan benar dan salah. Nilai-nilai ini biasanya direpresentasikan dalam matriks kebingungan. Matriks kebingungan adalah visualisasi tabulasi dari kinerja algoritme pembelajaran yang diawasi. Baris mewakili jumlah instance di kelas yang sebenarnya, sedangkan kolom mewakili jumlah instance di kelas prediktif. Tabel 5 menggambarkan matriks kebingungan untuk masalah klasifikasi biner.

Table 5. Contoh *Confusion matrix* untuk mengklasifikasi biner

| | | Predicted Class | |
|---|---|---|---|
| | | *Class 0* | *Class 1* |
| Actual Class | *Class 0* | 180 | 15 |
| | *Class 1* | 20 | 90 |

3. Confusion matrix.

Memberikan informasi yang cukup untuk menentukan kinerja pengklasifikasi yang berdiri sendiri. Namun, akan lebih mudah dan lebih jelas untuk menggambar elemen-elemen matriks menjadi satu nilai. Dalam penelitian ini matriks diringkas dengan menggunakan metrik akurasi yang dihitung sebagai berikut:

$$Accuracy = \frac{Correcly\ clasified\ instances}{Total\ Instances} x\ 100\% \quad (1)$$

Equation 1. Accuracy Rasio

4. Precision

Akurasi seringkali tidak cukup untuk menilai kinerja model pembelajaran. Meskipun akurasi memberikan indikasi apakah model sedang dilatih dengan benar, akurasi tidak memberikan informasi tentang informasi mendetail tentang aplikasi tertentu. Akibatnya, metrik kinerja lainnya digunakan, seperti presisi. Presisi didefinisikan sebagai tingkat dari positif yang diklasifikasikan dengan benar, atau positif benar. Ada banyak skenario ketika positif palsu mungkin berdampak. Dalam kasus penelitian ini, memiliki tingkat positif palsu yang tinggi berarti bahwa lalu lintas akan diidentifikasi sebagai berbahaya, padahal sebenarnya bukan. Di luar dunia akademis, hal ini dapat mengakibatkan pemborosan waktu dan biaya. Presisi dihitung sebagai berikut:

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives} \quad (2)$$

Equation 2. Precision ratio

5. Recall

Metrik kinerja lainnya adalah perolehan. Perolehan kembali adalah ukuran berapa banyak positif aktual yang ditemukan atau ditarik kembali. Ini juga merupakan metrik yang sangat penting, karena memiliki positif yang tidak terdeteksi, atau negatif palsu, mungkin memiliki konsekuensi serius di beberapa area. Misalnya, model yang tidak mengingat semua kasus serangan DDoS berarti bahwa lalu lintas jaringan yang berbahaya tidak akan diketahui, meningkatkan potensi kerusakan pada sistem dan penggunanya.

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Positives} \quad (3)$$

Equation 3. Recall ratio

6. F-measure.

F-measure adalah metrik yang memberikan skor akurasi keseluruhan untuk model dengan menggabungkan presisi dan perolehan. Skor F-measure yang baik berarti bahwa model memiliki positif palsu rendah dan negatif palsu rendah, dan oleh karena itu, model mengidentifikasi ancaman dengan benar sambil memiliki alarm palsu minimal.

$$F - measure = 2x \frac{Precision\ x\ Recall}{Precision + Recall} \quad (4)$$

Equation 4. F-measure rasio

7. Computation time

Metrik kinerja terakhir yang digunakan dalam penelitian ini adalah waktu komputasi. Ini tidak terkait langsung dengan klasifikasi, melainkan, ini menjelaskan waktu pelatihan yang dibutuhkan oleh model. Metrik ini memberikan indikasi efisiensi model. Waktu komputasi yang tercatat didasarkan pada Linux dengan RAM 4GB dan prosesor i3.

## HASIL DAN DISKUSI

Tabel 6 menyajikan evaluasi metrik model machine learning berdasarkan dataset CICDDoS, termasuk akurasi, presisi, recall, f-measure, dan waktu komputasi. Semua model pembelajaran mesin dilatih, divalidasi, dan diuji menggunakan splitting data 60:20:20 dari kumpulan data global. Tujuan dari evaluasi ini adalah untuk menganalisis kinerja dataset DdoS dalam hal kapasitasnya untuk mendeteksi intrusi (melalui serangan DDoS).

Dari sudut pandang model, model ansambel random forest berkinerja baik secara keseluruhan, mencapai akurasi mencapai 99 %. Selain itu, random forest juga mencapai presisi dan recall sama-sama mencapai 99%. Begitu ensemble boosting menggunakan adaboost-decision tree menghasilkan akurasi yang baik dan waktu komputasi pun lumayan cepat di bawah 10 detik

Terkait waktu komputasi, model random forest memerlukan waktu lebih lama untuk dilatih dengan CICDoS2019 sebagai sumber datanya, yaitu 84 detik. Kemungkinan besar, ini karena volume catatan kumpulan data, dengan total 1.046.845 baris. Namun dengan adaboost model membutuhkan waktu kurang dari sepuluh detik.

Tabel 6. Performance Matrix

| | **Random Forest** | **AdaBoost - Decision tree** |
|---|---|---|
| *Accuracy* | 0.99 | 0.99 |
| *Precision* | 0.99 | 0.99 |
| *Recall* | 0.99 | 0.99 |
| *F-measure* | 0.99 | 0.99 |
| *Computation TIme* | 84.2 seconds | 4.53 seconds |

Tingkat akurasi yang dicapai oleh model yang dilatih dengan dataset CICDDoS2019 [47]. Dari pengamatan awal, terlihat jelas bahwa model model berkinerja terbaik adalah random forest, mencapai akurasi 99%, dengan presisi 99% dan perolehan 99%. Model ini juga membutuhkan waktu paling lama untuk berlatih, dengan waktu komputasi 84,2 detik. Sedangkan model adaboost membutuhkan waktu kurang dari 10 detik untuk berlatih.

Studi ini mengeksplorasi perilaku dan penerapan dataset DDoS untuk pembelajaran mesin dalam konteks deteksi intrusi. Deteksi intrusi telah menjadi titik sakit dan subjek penelitian ekstensif karena kerentanan yang meningkat. Selama beberapa tahun terakhir, Internet telah tumbuh secara eksponensial dengan ribuan aplikasi berbasis komputer dibuat setiap hari. Dengan cepat, internet telah menjadi komponen penting untuk generasi saat ini, dan dengan pertumbuhannya yang agresif,

lingkungan jaringan yang aman menjadi penting. Di antara berbagai jenis serangan, serangan DDOS adalah salah satu ancaman terbesar bagi situs internet dan menimbulkan risiko yang menghancurkan keamanan sistem komputer, terutama karena potensi dampaknya. Oleh karena itu mengapa penelitian di bidang ini berkembang pesat, dengan para peneliti yang berfokus pada cara-cara baru untuk mengatasi deteksi dan pencegahan intrusi. Pembelajaran mesin dan kecerdasan buatan adalah beberapa tambahan terbaru dalam daftar teknologi yang diteliti untuk deteksi intrusi. Namun, banyak pemangku kepentingan industri dan peneliti masih kesulitan mendapatkan dataset berkualitas baik untuk mengevaluasi dan menilai model pembelajaran mesin deteksi mereka. Masalah inilah yang menjadi motivasi utama penelitian ini, dan menjadi dasar pertanyaan penelitian.

Pekerjaan ini dimulai dengan meninjau literatur di domain ini. Pertama, ulasan ini menyajikan garis besar tentang bagaimana peneliti lain mengeksplorasi dan menangani masalah deteksi intrusi dengan penerapan pembelajaran mesin. Ini memberi pemahaman yang lebih baik tentang bagaimana algoritma yang berbeda diterapkan dalam memecahkan masalah intrusi. Selain itu, ini juga memberikan wawasan tentang algoritma mana yang biasa digunakan untuk mengatasi masalah dalam domain ini dan bagaimana hasilnya diinterpretasikan dan dianalisis. Kedua, tinjauan pustaka menggali lebih dalam tentang karakteristik dan masalah kumpulan data saat ini. Berbagai pekerjaan dianalisis untuk mengeksplorasi seluk-beluk kumpulan data ini dan bagaimana validitasnya dipengaruhi dalam konteks metodologi pembelajaran mesin. Berbagai masalah terungkap sehubungan dengan kumpulan data yang ada, termasuk masalah privasi, ketersediaan dokumentasi, aksesibilitas, dan keselarasan dengan tujuan penelitian. Ini diikuti dengan review pekerjaan sebelumnya terkait dengan survei dan perbandingan dataset.

Studi ini menyajikan solusi untuk analisis efektivitas dataset DDoS yang ada untuk mendeteksi intrusi, menggunakan algoritma random forest dan adaboost pada metode supervised learning.

Dalam percobaan, dataset DDoS digunakan: CIC DDoS 2019 Performa deteksi intrusi set data ini dianalisis menggunakan dua model pembelajaran mesin. Dataset dibagi dalam rasio 60:20:20 untuk pelatihan model, validasi dan pengujian. Model pembelajaran mesin dipilih secara sistematis dan hati-hati untuk memastikan bahwa eksperimen dilakukan dengan cara yang tepat. Kedua model tersebut yaitu random forest dan adaboost. Hasilnya dianalisis menggunakan sekumpulan metrik performa, termasuk akurasi, presisi, recall, f-measure, dan waktu komputasi.

## KESIMPULAN

Meskipun tidak adanya kumpulan data merupakan titik fokus utama studi ini dilakukan, hal ini juga dapat dilihat sebagai batasan tersendiri mengingat fakta bahwa hasil yang berpotensi lebih akurat akan diperoleh dari perbandingan antar kumpulan data.

Meskipun area IDS banyak diteliti, ada banyak aspek yang harus diselidiki lebih lanjut, terutama di area pembelajaran mesin. Secara khusus, pekerjaan di masa depan dapat fokus pada

penyediaan aplikasi atau layanan yang dengannya setiap dataset baru dapat dengan cepat dianalisis dan dimasukkan ke dalam tolok ukur dengan algoritma yang dipilih oleh peneliti dengan cara yang sama seperti yang dilakukan penelitian ini.

Keluar analisis dari database yang dipilih. Aplikasi tersebut dapat menjawab pertanyaan 'kumpulan data mana yang berkinerja lebih baik dan dengan algoritma mana?'. Ini akan sangat membantu para peneliti yang mencari dataset berkinerja baik dan juga menginginkan pendekatan yang konsisten terhadap hasil, dengan menggunakan set data dan algoritma berkinerja terbaik.

Sehubungan dengan kemungkinan pekerjaan di masa depan, area menarik yang dapat dieksplorasi adalah bagaimana data spesifik IDS dapat direpresentasikan sebagai jaringan syaraf tiruan dan selanjutnya dianalisis dengan pembelajaran mendalam, setelah kumpulan data dapat direpresentasikan dalam bentuk data non-terstruktur. Hal di atas juga dapat dilihat sebagai dua masalah terpisah, yang dapat dikembangkan oleh studi selanjutnya.

## REFERENSI

N. Z. Bawany, J. A. Shamsi, and K. Salah, “DDoS Attack Detection and Mitigation Using SDN: Methods, Practices, and Solutions,” Arabian Journal for Science and Engineering. 2017, doi: 10.1007/s13369-017-2414-5.

M. A. Aydin, A. H. Zaim, and K. G. Ceylan, “A hybrid intrusion detection system design for computer network security,” Comput. Electr. Eng., 2009, doi: 10.1016/j.compeleceng.2008.12.005.

M. F. Fibrianda and A. Bhawiyuga, “Analisis Perbandingan Akurasi Deteksi Serangan Pada Jaringan Komputer Dengan Metode Naïve Bayes Dan Support Vector Machine (SVM),” J. Pengemb. Teknol. Inf. dan Ilmu Komput., vol. 2, no. 9, pp. 3112–3123, 2018.

P. Probst, M. N. Wright, and A. L. Boulesteix, “Hyperparameters and tuning strategies for random forest,” Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery. 2019, doi: 10.1002/widm.1301.

S. Raschka and V. Mirjalili, Python Machine learning. Machine learning and Deep Learning with Python, scikit-learn, and tensorflow. 2017.

K. J. Grimm, G. L. Mazza, and P. Davoudzadeh, “Model Selection in Finite Mixture Models: A k-Fold Cross-Validation Approach,” Struct. Equ. Model., 2017, doi: 10.1080/10705511.2016.1250638.

I. Ullah, B. Raza, A. K. Malik, M. Imran, S. U. Islam, and S. W. Kim, “A Churn Prediction Model Using Random Forest: Analysis of Machine Learning Techniques for Churn Prediction and Factor Identification in Telecom Sector,” IEEE Access, 2019, doi: 10.1109/ACCESS.2019.2914999.

Pandas.pydaya.org, “Pandas.Dataframe.Fillna,” Pandas 1.0.3 Documentation [online].
Available : https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.fillna.html.

Scikit-learn, “StratifiedKFold,” Scikit-learn 0.22.2 Documentation, 2019

University of New Brunswick, “DDoS Evaluation Dataset (CICDDoS2019),” unb.ca, 2019. [Online]. Available: https://www.unb.ca/cic/datasets/ddos-2019.html

A. H. Lashkari, Y. Zang, G. Owhuo, M. S. I. Mamun, and G. D. Gil, “CICFlowMeter,” Github. 2017