Jurnal Elektronik Ilmu Komputer Udayana p-ISSN: 2301-5373
Volume 13, No 3. February 2025 e-ISSN: 2654-5101

Klasifikasi Serangan Distributed Denial of Service (DDoS) Mempergunakan Support Vector Machine dengan Correlation-Based Feature Selection

I Gusti Ngurah Made Dika Varuna[a1], I Gusti Agung Gede Arya Kadyanan [a2], Anak Agung Istri Ngurah Eka Karyawati [a3], Made Agung Raharja[a4]

[a]Informatics Engineering, Faculty of Math and Science, University of Udayana
South Kuta, Badung, Bali, Indonesia
[1]ngurahdika22@gmail.com
[2] gungde@unud.ac.id
[3]eka.karyawati@unud.ac.id
[4]made.agung@unud.ac.id

Abstract

Distributed Denial of Service (DDoS) attacks provide several drawbacks for a firm and may lead to substantial losses. This project will use "Correlation-Based Feature Selection" (CFS) for the classification of DDoS assaults. The author will use the "CSE-CIC-IDS2018" dataset in this investigation. The feature selection of the dataset using CFS yields a sequential ranking of 68 characteristics, ordered from highest to lowest value. Only 31 characteristics are used here, since their values exceed 0.1. The system achieves high accuracy across different kernels, beginning with the linear kernel with an accuracy of 1.0, followed by the polynomial kernel at 0.99, and the RBF kernel also at 0.99. This investigation reveals that the accuracy achieved without CFS is equivalent to that acquired with CFS; however, the ROC AUC score for the polynomial kernel is 0.98, while the linear and RBF kernels get a score of 1.0. In this instance, CFS does not provide a notable enhancement in the efficacy of the SVM model. Nonetheless, CFS is advantageous for streamlining the model and decreasing the amount of used features without compromising performance..

Keyword: *Distributed Denial of Service (DDoS), Correlation-based Feature Selection, Support Vector Machine, Classification, Kernel, CSE-CIC-IDS2018, Feature Selection*

1. Pendahuluan

Kemajuan teknologi saat ini terjadi dengan kecepatan yang luar biasa, yang pada akhirnya menciptakan "ketergantungan masyarakat" didalam penggunaan sistem komputer. Komputer ini dipergunakan didalam berbagai sektor, termasuk industri, komunikasi, dan bisnis, serta mempunyai banyak fungsi lainnya. Secara umum, system komputer berfungsi dengan terhubung melalui jaringan yang memungkinkan transfer data dan informasi.

Didalam beberapa tahun terakhir, ancaman keamanan terhadap jaringan komputer semakin sering terjadi, yang mengharuskan diterapkannya langkah-langkah mitigasi. Satu diantara metode yang dipergunakan yaitu "perlakuan guna mendeteksi serangan jaringan", yang bertujuan guna menaikkan keamanan jaringan komputer secara keseluruhan. Serangan Distributed Denial of Service (DDoS), yang sudah menjadi sangat populer dari tahun 1990, yaitu satu diantara jenis serangan jaringan yang paling sering terjadi. Hacker dan pelaku peretasan mempergunakan DDoS sebagai "senjata pilihan" mereka. Teknik ini sangat populer karena sangat efektif didalam menghadapi ancaman besar di internet [1]. Distributed Denial of Service (DDoS) mempunyai dampak yang signifikan terhadap operasional perusahaan, seringkali menyebabkan kerugian yang substansial. Serangan ini bekerja dengan cara membanjiri server dengan "jumlah paket data yang sangat besar" secara bersamaan, sehingga mengakibatkan server menjadi tidak responsif, tidak bisa diakses, dan kehilangan kapabilitas guna menjalankan fungsinya secara normal [2]. Distributed Denial of Service (DDoS) mempunyai berbagai jenis serangan yang sering terjadi, seperti: "UDP Flooding, SYN Flooding, Ping of Death, dan Remote Controlled Attack". Dampak dari serangan ini biasanya berupa gangguan signifikan pada sistem, termasuk error request, penghentian sistem (halt), hingga kegagalan total fungsi sistem.

"Berdasarkan penelitian yang mengkaji Analisis Perbandingan Akurasi Deteksi Serangan Pada Jaringan Komputer dengan Metode Naïve Bayes dan Support Vector Machine (SVM), serangan Denial

of Service (DoS) didefinisikan sebagai tindakan guna melumpuhkan server komputer pada jaringan internet, sehingga komputer tidak bisa berfungsi dengan baik". Didalam upaya mencegah potensi serangan ini, peneliti mengembangkan Intrusion Detection System (IDS) yang mempergunakan dua pendekatan deteksi, yaitu Rule-Based (Signature-Based) dan Behavior-Based.

Penelitian tersebut memanfaatkan metode Behavior-Based, yang bergantung pada dataset dan algoritma guna menganalisis pola serangan. Algoritma yang dibandingkan didalam penelitian ini meliputi Naïve Bayes, SVM Linear, SVM Polynomial, dan SVM Sigmoid. Dataset yang dipergunakan yaitu ISCX2012 testbed pada 14 Juni 2012. Evaluasi algoritma dilaksanakan mempergunakan metrik akurasi, precision, recall, dan F1-score. Hasil memperlihatkan bahwsanya akurasi tertinggi dicapai oleh SVM Polynomial dengan 99,99%, sedangkan Naïve Bayes mencatat akurasi terendah sebesar 85,55%.

Penelitian yang sedang direncanakan oleh penulis akan mempergunakan dataset CSE-CIC-IDS2018, yang mencakup data normal dan data serangan. Dataset ini dipilih karena baru dikembangkan pada tahun 2019 dan dianggap lebih relevan guna mendeteksi serangan terbaru. Penelitian ini juga akan menerapkan metode Correlation-Based Feature Selection guna menaikkan akurasi deteksi serangan DDoS. Proses seleksi fitur dilaksanakan guna mengidentifikasi fitur-fitur relevan berdasarkan bobot tertinggi. Hasil seleksi akan dibandingkan dengan analisis tanpa seleksi fitur guna mengevaluasi damphak penerapan seleksi fitur terhadap kinerja algoritma klasifikasi, khususnya Support Vector Machine (SVM). Penelitian ini bertujuan guna mengukur sejauh mana seleksi fitur bisa menaikkan performa deteksi serangan DDoS mempergunakan dataset CSE-CIC-IDS2018. .

2. Metode Penelitian

Data penelitian CSE-CIC-IDS2018 dipergunakan didalam penelitian ini. Dengan mempergunakan algoritma Support Vector Machine, anomali jaringan bisa dideteksi oleh serangan Distributed Denial of Service (DDoS). Pada tahap preprocessing akan dilaksanakan 2 proses yaitu cleaning dataset yang membersihkan dataset dari data yang hilang dan terduplikasi setelah itu akan dilaksanakan spliting dataset dan menghasilkan data training dan data testing . Didalam sistem ini ditemukan dua proses utama. Pertama, sistem melaksanakan seleksi fitur terhadap dataset. Tujuan memilih fitur ini yaitu guna memperoleh fitur yang paling sesuai dengan data. Guna tahap seleksi fitur ini, metode Correlation Based Feature Selection (CFS) dipergunakan. Setelah memperoleh fitur yang dipergunakan didalam sistem, data diklasifikasikan mempergunakan metode Support Vector Machine.

2.1 Data Mining

Data mining didefinisikan sebagai "proses ekstraksi ataupun penemuan informasi tersembunyi" dari kumpulan data yang tersedia. Pengetahuan yang dihasilkan melalui proses ini mempunyai potensi aplikasi luas di berbagai sektor, seperti bisnis, pendidikan, kesehatan, dan bidang lainnya. Didalam konteks ini, data mining memanfaatkan kombinasi teknik statistik, matematika, kecerdasan buatan (artificial intelligence), dan machine learning guna mengidentifikasi informasi bernilai dari data berskala besar. Sebagai sebuah disiplin ilmu, data mining berfungsi guna "menemukan, menggali, ataupun menambang pengetahuan" dari data yang sudah tersedia. Proses ini mengintegrasikan berbagai pendekatan multidisipliner, termasuk teknologi basis data dan gudang data, statistik, pembelajaran mesin, komputasi berkinerja tinggi, pengenalan pola, jaringan saraf, dan visualisasi data. Pendekatan tersebut memungkinkan penggalian informasi yang lebih efektif dan efisien dari data yang kompleks. Secara umum, aktivitas didalam data mining bisa dikelompokkan didalam dua kategori utama, yaitu:

a. Descriptive mining merupakan metode yang dipergunakan guna mengidentifikasi informasi penting yang terkandung didalam basis data. Teknik-teknik yang termasuk didalam kategori ini meliputi clustering, association, dan sequential mining.
b. Prediksi Penambangan yaitu pendekatan yang bertujuan guna menemukan pola didalam data dengan mempertimbangkan variabel-variabel yang relevan guna prediksi di masa mendatang. Satu diantara teknik utama yang dipergunakan didalam metode ini yaitu klasifikasi. .

Data mining dibagi menjadi beberapa kelompok berdasarkan tugas yang bisa dilaksanakan, yaitu:

1. Description

Peneliti dan analis mempergunakan metode sederhana yang dikenal sebagai deskripsi guna menjelaskan pola dan kecenderungan yang ditemukan didalam data. Metode ini sering memberikan penjelasan yang mungkin mendasari pola ataupun kecenderungan yang dilihat..

2. Classification

Didalam data mining, klasifikasi yaitu teknik yang sering dipergunakan guna membuat model ataupun fungsi yang memetakan data didalam kategori tertentu. Proses ini melibatkan mengidentifikasi karakteristik setiap objek dan memasukkannya didalam kelas yang sudah ditentukan sebelumnya. Guna validasi, aturan klasifikasi dipergunakan guna menguji data dan menilai akurasi hasilnya. Aturan bisa diterapkan pada data baru jika dianggap cukup akurat. Algoritma pilihan pohon yaitu contoh yang sering dipergunakan.

3. Clustering

Proses identifikasi kelompok objek berdasarkan kemiripan di diantara mereka dikenal sebagai clustering. Kumpulan rekaman yang serupa satu sama lain tetapi berbeda satu sama lain disebut kluster. Metode ini membantu memahami hubungan diantara atribut dan pola distribusi data. Clustering juga bisa dipergunakan guna membedakan kelompok ataupun kelas objek lebih lanjut..

4. Prediction

Prediksi mempunyai kesamaan dengan klasifikasi dan estimasi, namun perbedaannya terletak pada tujuannya yang berfokus guna memproyeksikan nilai hasil yang akan terjadi di masa depan. Teknik-teknik yang dipergunakan didalam klasifikasi dan estimasi seringkali bisa diterapkan juga didalam konteks prediksi. Sebagai contoh, satu diantara aplikasi prediksi yaitu memperkirakan harga beras didalam tiga bulan mendatang.

5. Association rule

Tujuan aturan asosiasi yaitu guna menemukan atribut yang sering muncul sekaligus. Metode ini dikenal didalam dunia bisnis sebagai analisis keranjang belanja. Organisasi berusaha guna mengukur hubungan diantara dua ataupun lebih fitur.

2.2 Correlation-based Feature Selection (CFS)

Correlation-Based Feature Selection (CFS) yaitu metode guna memilih fitur yang berguna didalam pengklasifikasian data dengan melihat hubungan antar fitur dan target. Metode ini menganggap bahwsanya fitur yang dipilih harus atribut yang mempunyai korelasi tinggi dengan kelasnya serta mempunyai tingkat korelasi rendah dengan atribut lainya dan menaikkan akurasi model prediksi. [4].

Tujuan didalam *Correlation-Based Features Selection* (CFS) agar menghindari rendundansi pada fitur, berikut yaitu rumus utama pada CFS :

- korelasi antar fitur dengan target $(\boldsymbol{r}_{cf})$
- Korelasi antar fitur $(\boldsymbol{r}_{ff})$

Rumus evaluasi subset fitur didalam CFS yaitu :

$$merit_s = \frac{(\boldsymbol{k}.\bar{\boldsymbol{r}}_{cf})}{\sqrt{k} + (k-1).\bar{\boldsymbol{r}}_{ff}} \quad (1)$$

2.3 *Support Vector Machine*

Algoritma pembelajaran mesin Support Vector Machine (SVM) dipergunakan guna menyelesaikan masalah regresi dan klasifikasi. SVM mempergunakan metode "mencari hyperplane optimal", yang memungkinkannya secara efektif memisahkan dua kelas data. SVM dipergunakan guna membuat model yang bisa "memisahkan data normal dari data yang mencurigakan ataupun berpotensi berbahaya" guna mendeteksi ancaman keamanan jaringan. Metode ini membuat SVM sangat penting guna mengidentifikasi serangan dan mengklasifikasikan data dengan akurasi tinggi [4]

Support Vector Machine (SVM) mempunyai beberapa rumus utama yang secara umum dipergunakan. Berikut yaitu penjelasan tentang rumus – rumus utama tersebut :

1. Hyperplane

Hyperplane yaitu batas keputusan yang memisahkan ruang fitur didalam dua kelas. Secara matematis, hyperplane didalam ruang berdimensi nnn bisa dinyatakan sebagai:

$$\boldsymbol{w} . \boldsymbol{x} + b = 0 \quad (2)$$

Di mana:

- $\boldsymbol{w}$ yaitu vektor bobot
- $\boldsymbol{x}$ yaitu vektor fitur
- b yaitu bias ataupun offset

2.4 Skenario Pengujian Sistem

Pada Skenario pengujian dipergunakan guna mengukur tingkat keberhasilan pada sistem yang dibuat. Berikut yaitu langkah – langkah dan penjelasan dari skenario pengujian :

- **Preprocessing dataset**
  - Cleaning Data

    Langkah awal didalam proses preprocessing yaitu melaksanakan pembersihan dataset dengan menghilangkan missing values dan duplikasi data. Tahapan ini sangat penting guna memastikan bahwsanya data yang dipergunakan mempunyai validitas tinggi dan bebas dari kesalahan yang berpotensi memengaruhi akurasi dan performa model yang dihasilkan.

  - Spliting Data

    Pembagian dataset menjadi dua bagian, biasanya 80% guna data latih dan 20% guna data uji, dilaksanakan setelah tahap pembersihan data selesai. Data latih dipergunakan guna melatih model, sementara data uji dipergunakan guna menguji seberapa baik model bekerja.
- **Seleksi fitur**
  - Menghitung Nilai Korelasi Guna Setiap Fitur

    Guna setiap fitur didalam dataset, hitung nilai korelasi diantara fitur tersebut dan label target. Korelasi menunjukan seberapa kuat hubungan diantara fitur dan tabel

  - Mengurutkan Fitur Berdasarkan Nilai Korelasi

    Setelah Menghitung nilai korelasi guna setiap fitur, urutkan fitur – fitur tersebut dari yang mempunyai nilai korelasi tertinggi hingga terendah, Fitur dengan nilai korelasi tertinggi dianggep paling relevan guna model

- **Melatih Model SVM**

  Gunakan data latih yang sudah dibagi guna melatih model SVM. Model SVM akan belajar dari data latih guna membedakan diantara serangan DDoS dan lalu lintas normal.
- **Pengujian Hyperparameter**

  Pengujian hyperparameter sangat penting guna menemukan kombinasi yang paling optimal yang akan memberikan performa terbaik pada dataset yang dipergunakan. Setiap kernel didalam SVM mempunyai karakteristik yang berbeda dan memerlukan penyetelan yang tepat guna mencapai hasil yang optimal.
  - C (Regularization Parameter)

    Mengelola trade-off diantara memperoleh decision margin yang lebar dan mengklasifikasikan data pelatihan dengan benar merupakan bagian penting didalam proses pelatihan model. Pengaturan nilai C yang kecil akan menghasilkan margin yang lebih lebar, namun memungkinkan sejumlah data pelatihan diklasifikasikan secara tidak tepat. Sebaliknya, nilai C yang besar akan berfokus pada upaya mengklasifikasikan semua data pelatihan dengan benar, meskipun decision margin menjadi lebih sempit dan model menjadi lebih rentan terhadap overfitting. Hyperparameter ini mempunyai rentang nilai yang bisa diatur, seperti contoh berikut: `C`: [0.1, 1, 10, 100].

- **Evaluasi Model**

Setelah model dilatih, gunakan data uji guna mengevaluasi performa model. Prediksi dilaksanakan pada data uji, dan hasilnya dibandingkan dengan label sebenarnya guna menghitung metrik kinerja.

- **Menghitung Metrik Kinerja**
  - Precision

Precision yaitu proporsi dari prediksi positif yang benar – benar positif. Mengukur akurasi dari deteksi positif

$$\text{Precision} = \frac{TP}{TP + FP} \quad (3)$$

- Recall

Recall yaitu proporsi dari total data positif yang berhasil terdeteksi. Mengukur kapabilitas model didalam mendeteksi serangan

$$Recall = \frac{TP}{TP + FN} \quad (4)$$

- F1-Score

F1-Score yaitu rata – rata harmonis dari *Precision* dan *Recall*. Memberikan keseimbangan diantara Precision dan Recall

$$F1 - Score = \frac{2 \text{ X } Precision \text{ X } Recall}{Precision + Recall} \quad (5)$$

- Accuracy

Accuracy yaitu proporsi dari total prediksi yang benar terhadap total data. Mengukur keakuratan keseluruhan model.

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

- **Proses Pengujian Terakhir**
  - Mempergunakan Data Uji guna melaksanakan Prediksi

Setelah model SVM dilatih dan dievaluasi, lakukan prediksi pada data uji yang sudah disiapkan. Ini yaitu pengujian akhir guna melihat seberapa baik model berperformansi

- Menghitung Metrik Kinerja pada Data Uji

Hitung kembali metrik kinerja (*Precision, Recall, F1-Score, dan Accuracy*) berdasarkan hasil prediksi pada data uji. Ini akan memberikan gambaran tentang performa akhir model.

3. Implementasi sistem

Guna penelitian ini, penulis mempergunakan dataset CSE-CIC-IDS2018, yang bisa ditemukan di www.unb.ca data normal, yang dikategorikan didalam kelas serangan DDoS HOIC dan DDoS LOIC UDP. Namun, penulis hanya berfokus pada dua kelas serangan DDoS, yaitu serangan DDoS HOIC (Denial of Service) normal. Tabel 1 memperlihatkan jumlah paket data yang ada didalam dataset yang dipergunakan.

Tabel 1. *Jumlah Paket Data*

No	Nama	Jumlah
1	Normal	360.833
2	DDoS HOIC	686.012
3	DDoS LOIC UDP	1.730
Total		1.048.575

High Orbit Ion Cannon (HOIC), yang sering disingkat, yaitu aplikasi Denial of Service (DoS) yang dikembangkan didalam bahasa pemrograman BASIC dan dirancang guna menyerang hingga 256 URL secara bersamaan. Didalam skenario yang dipergunakan oleh UNB guna memperoleh dataset CSE-CIC-IDS2018, para peneliti memanfaatkan alat HOIC yang tersedia secara gratis guna melaksanakan serangan DDoS dengan mempergunakan empat komputer yang berbeda. Beberapa contoh dari dataset CSE-CIC-IDS2018 yang dipergunakan didalam penelitian ini tampak pada Tabel 2.

Tabel 2. Contoh Data CSE-CIC-IDS2018

	Dst Port	Protocol	Timestamp	...	Idle Max	Idle Min	Label
0	80	6	21/02/2018 08:33:25	...	0	0	Benign
1	500	17	21/02/2018 08:33:06	...	75600000	42000000	Benign
2	500	17	21/02/2018 08:33:06	...	75600000	42000000	Benign
3	500	17	21/02/2018 08:33:11	...	75600000	7200397	Benign
4	500	17	21/02/2018 08:33:11	...	75600000	7200399	Benign

Implementasi seleksi fitur diawali dengan memasukan dataset CSE-CIC-IDS2018. Setelah itu penulis melaksanakan perhitungan nilai relasi dan menentukan fitur yang dipilih yaitu sebanyak 31 fitur karena mempunyai nilai relasi diatas 0.1. Fitur – fitur tersebut diuraikan pada Tabel 3

Tabel 3. Relasi 31 Fitur Correlation-Based Feature Selection

No	Nama Fitur	Nilai Relasi	No	Nama Fitur	Nilai Relasi
1	Init Bwd Win Byts	0.99116	17	Flow Byts/s	0.68469
2	Dst Port	0.98643	18	Bwd Seg Size Avg	0.47521
3	Fwd Pkt Len Max	0.96074	19	Bwd Pkt Len Mean	0.47521
4	Fwd Pkt Len Std	0.93834	20	Bwd Pkts/s	0.39738
5	Fwd Seg Size Avg	0.90416	21	Tot Bwd Pkts	0.3311
6	Fwd Pkt Len Mean	0.90416	22	Subflow Bwd Pkts	0.3311
7	ACK Flag Cnt	0.72197	23	Down/Up Ratio	0.31072
8	Pkt Len Mean	0.71553	24	Flow Pkts/s	0.28222
9	Pkt Len Max	0.71511	25	Bwd Pkt Len Std	0.27725
10	Pkt Size Avg	0.71508	26	Bwd Header Len	0.25472
11	PSH Flag Cnt	0.71463	27	Fwd Pkts/s	0.21332
12	Pkt Len Std	0.71419	28	Bwd IAT Min	0.14286
13	Pkt Len Var	0.71413	29	TotLen Bwd Pkts	0.10837
14	RST Flag Cnt	0.71397	30	Subflow Bwd Byts	0.10837
15	ECE Flag Cnt	0.71397	31	Bwd Pkt Len Max	0.10826

p-ISSN: 2301-5373
e-ISSN: 2654-5101

16	Init Fwd Win Byts	0.69601			

Didalam proses klasifikasi, fitur-fitur yang tidak penting dikurangi melalui proses seleksi fitur. Persamaan Correlation-based Feature Selection (CFS) bisa ditemukan [5]. Setelah perhitungan selesai, langkah selanjutnya yaitu memilih fitur dengan kualitas terbaik. Didalam penelitian ini, teknik seleksi fitur berbasis korelasi dipergunakan. Teknik ini menghitung dan membandingkan tingkat korelasi diantara atribut dengan kelas dan diantara satu atribut dengan atribut lainnya. Atribut yang dipilih yaitu atribut yang mempunyai korelasi tinggi dengan kelasnya tetapi mempunyai korelasi rendah dengan atribut lainnya.

Pada titik ini, evaluasi model dilaksanakan mempergunakan berbagai macam kernel; namun, pengujian ini tidak menerapkan seleksi fitur berbasis korelasi. Ini memperlihatkan bahwsanya fitur-fitur yang dipergunakan didalam model tidak mempunyai nilai relasi yang jelas, mulai dari nilai tertinggi hingga terendah. Akibatnya, fitur-fitur tersebut tetap dianggap acak.

Evaluasi model yang dihasilkan dari berbagai *kernel support vector machine* yaitu sebagai berikut :

a. Hasil evaluasi model pada *kernel linear*

Tabel 4. Evaluasi Hasil Model Linear

```
Best hyperparameters found for linear kernel: {'C': 0.1}
Accuracy for linear kernel: 0.9998039215686274
Confusion Matrix:
[[3252    1]
 [   0 1847]]

Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      3253
           1       1.00      1.00      1.00      1847

    accuracy                           1.00      5100
   macro avg       1.00      1.00      1.00      5100
weighted avg       1.00      1.00      1.00      5100


ROC AUC Score for linear kernel: 1.0
```

Pada Tabel 4, terlihat bahwsanya parameter terbaik yang ditemukan guna kernel linear yaitu ‘C: 0.1’. Parameter C ini berfungsi guna mengatur trade-off diantara upaya memaksimalkan margin dan meminimalkan kesalahan klasifikasi, yang memperlihatkan bahwsanya model lebih fokus pada margin. Dengan tingkat akurasi 99,98%, model tersebut berhasil mengklasifikasikan hampir seluruh sampel dengan benar.. Selanjutnya ada *confusion matrix* yang menunjukan bahwsanya :

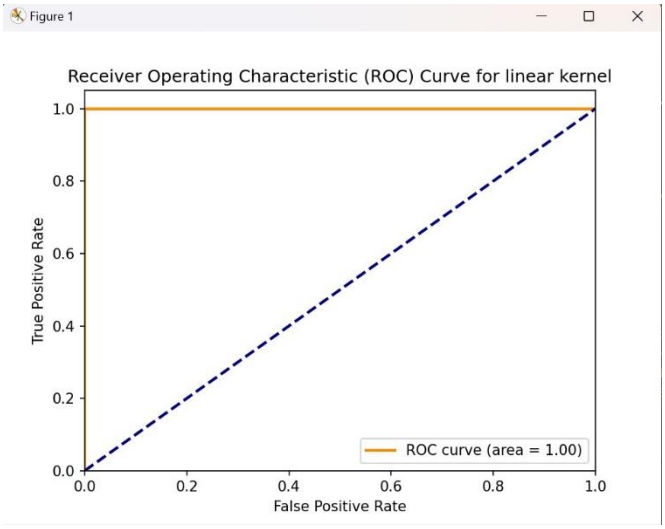
- 3253 sampel dari kelas 0 (Benign) diklasifikasikan dengan benar sebagai kelas 0
- 1847 sampel dari kelas 1 (DDoS) diklasifikasikan dengan benar sebagai kelas 1
- 1 sampel dari kelas 0 diklasifikasikan sebagai kelas 1, menunjukan bahwsanya satu kesalahan pada klasifikasi

Pada tahap *classification report* memperoleh nilai 1.0 guna semua semua kategori mulai dari *precision, recall,* dan *f1-score* yang berarti semua prediksi positif, sampel aktual terdeteksi dengan benar dan juga keseimbangan diantara *precision* dan *recall* menunjukan hasil yang sempurna. Nilai *support* menunjukan jumlah sampel disetiap kelas (3253 guna kelas 0 dan 1847 guna kelas 1).

Tabel 5. Hyperparameter Linear

Kernel Linear	C : 0.1	C : 1	C : 10
Accuracy	99,98%	99,98%	99,98%

Pada tabel 5 bisa dilihat bahwsanya hasil dari ketiga *hyperparameter* mempunyai nilai yang sama yaitu 99,98% namun walaupun mempunyai nilai yang sama penggunaan *hyperparameter* C : 0.1 tetap yang terbaik didalam klasifikasi model SVM



Gambar 1. ROC Linear

Pada gambar 1 bisa dilihat bahwsanya nilai ROC AUC pada *kernel linear* yaitu 1.0 yang berarti model mempunyai kapabilitas sempurna guna membedakan diantara kelas 0 dan kelas 1.

b. Hasil evaluasi model kernel polynomial

Tabel 6. Evaluasi Hasil Model Polynomial

```
Best hyperparameters found for poly kernel: {'C': 0.1, 'degree': 3, 'gamma': 'scale'}
Accuracy for poly kernel: 0.9998039215686274
Confusion Matrix:
[[3252    1]
 [   0 1847]]

Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      3253
           1       1.00      1.00      1.00      1847

    accuracy                           1.00      5100
   macro avg       1.00      1.00      1.00      5100
weighted avg       1.00      1.00      1.00      5100


ROC AUC Score for poly kernel: 1.0
```

Pada tabel 6 diatas menunjukan bahwsanya hasil pelatihan *kernel polynomial* menunjukan performa yang sangat baik meskipun sedikit kurang apabila dibandingakan dengan *kernel linear*. Parameter terbaik yang dipergunakan yaitu ‘C: 0.1’, degree : 3’, ‘gamma’ : ‘scale’, sama seperti *kernel linear* yaitu model lebih memfokuskan pada memaksimalkan margin dan juga *degree* = 3 menunjukan derajat dari kernel polynomial. *Degree* = 3 berarti polynomial pangkat 3 dipergunakan didalam transformasi data guna menaikkan separabilitasnya. Selanjutnya guna *gamma* = ‘scale’ itu berarti skala *gamma* mengatur berdasarkan jumlah fitur dan varians. Gamma ini mengontrol seberapa jauh satu titik data mempengaruhi sekitar, dimana ‘scale’ yaitu nilai otomatis yang menghitung gamma secara proporsional terhadap jumlah fitur. Pada tingkat akurasi *kernel polynomial* memperoleh nilai 99,98%yang berarti model telah mengklasifikasikan hampir semua sampel itu benar dengan hanya sedikit kesalahan. Selanjutnya pada nilai *confusion matrix* yang menunjukan bahwsanya :

- 3252 kelas sampel dari kelas 0 (Benign) diklasifikasikan dengan benar sebagai kelas 0

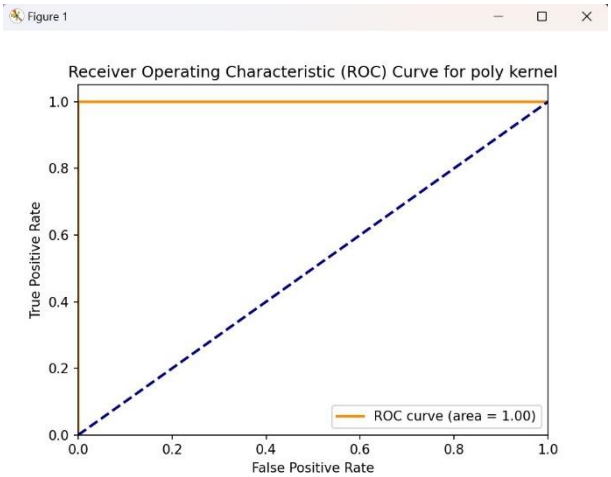
- 1847 sampel dari kelas 1 (DDoS) diklasifikasikan dengan benar sebagai kelas 1
- 1 sampel dari kelas 0 diklasifikasikan sebagai kelas 1, menunjukan bahwsanya satu kesalahan pada klasifikasi

Pada tahap *classification report, kernel polynomial* mempunyai nilai yang sama seperti *kernel linear* yang berarti menunjukan hasil yang sempurna

Tabel 7. Hyperparameter Poly

No	C	Degree	Gamma	Accuracy
1	0.1	2	Scale	99,97%
2	0.1	2	Auto	99,97%
3	0.1	3	Scale	99,98%
4	0.1	3	Auto	99,98%
5	0.1	4	Scale	99,97%
6	0.1	4	Auto	99,97%
7	1	2	Scale	99,97%
8	1	2	Auto	99,97%
9	1	3	Scale	99,97%
10	1	3	Auto	99,97%
11	1	4	Scale	99,97%
12	1	4	Auto	99,97%
13	10	2	Scale	99,97%
14	10	2	Auto	99,97%
15	10	3	Scale	99,98%
16	10	3	Auto	99,98%
17	10	4	Scale	99,97%
18	10	4	Auto	99,97%

Pada tabel 7 bisa dilihat bahwsanya hasil dari *hyperparameter* mempunyai nilai yang berbeda dan nilai terbaik didalam beberapa penggunaan *hyperparameter* diatas yaitu penggunaan C : 0.1, degree = 3, gamma = scale dengan nilai 99.98%



Gambar 2. ROC Polynomial

Pada gambar 2 bisa dilihat bahwsanya nilai ROC AUC pada *kernel polynomial* yaitu 99,98% yang berarti model mempunyai kapabilitas yang hampir sempurna guna membedakan diantara kelas 0 dan 1 walaupun tidak sebaik *kernel linear.*

c. Hasil evaluasi model *kernel rbf*

Tabel 8. Evaluasi Hasil Model rbf

```
Best hyperparameters found for rbf kernel: {'C': 1, 'gamma': 'scale'}
Accuracy for rbf kernel: 0.9998039215686274
Confusion Matrix:
[[3252    1]
 [   0 1847]]

Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      3253
           1       1.00      1.00      1.00      1847

    accuracy                           1.00      5100
   macro avg       1.00      1.00      1.00      5100
weighted avg       1.00      1.00      1.00      5100


ROC AUC Score for rbf kernel: 1.0
```

Pada tabel 8 diatas menunjukan bahwsanya hasil pelatihan *kernel* rbf menunjukan performa yang sangat baik. Parameter terbaik yang dipergunakan yaitu ‘C: 1’, *gamma* = Scale. Parameter C = 1 berarti regulasi yang mengontrol margin antar kelas dan nilai 1 menunjukan bahwsanya model berusaha menyeimbangkan diantara margin yang lebih besar dan jumlah kesalahan klasifikasi yang kecil. Sedangkan *gamma* = scale mengontrol seberapa jauh pengaruh sebuah titik data terhadap sekitarnya didalam model dan nilai ‘scale’menghitung *gamma* secara proporsional terhadap jumlah fitur, yang merupakan pilihan otomatis. Pada tingkat akurasi *kernel* rbf memperoleh nilai yang sama seperti *kernel polynomal* yaitu 99,98% yang berarti model telah mengklasifikasikan hampir semua sampel itu benar dengan hanya sedikit kesalahan. Selanjutnya pada nilai *confusion matrix kernel* rbf mempunyai nilai yang sama seperti *kernel polynomial* yaitu :

- 3252 kelas sampel dari kelas 0 (Benign) diklasifikasikan dengan benar sebagai kelas 0
- 1847 sampel dari kelas 1 (DDoS) diklasifikasikan dengan benar sebagai kelas 1
- 1 sampel dari kelas 0 diklasifikasikan sebagai kelas 1, menunjukan bahwsanya satu kesalahan pada klasifikasi

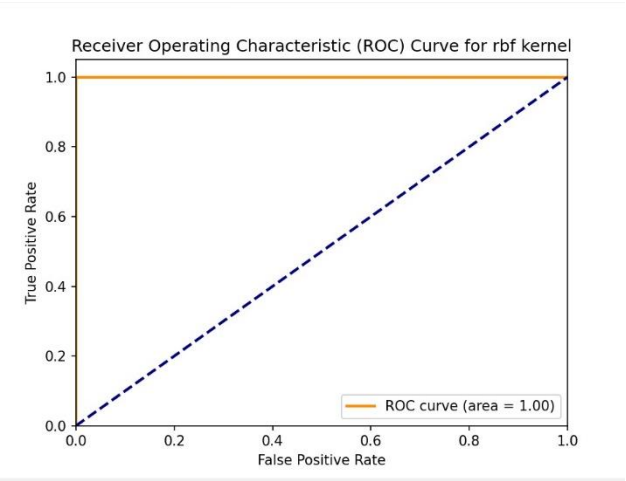
Tabel 9. Hyperparameter rbf

No	C	Gamma	Accuracy
1	0.1	Scale	99,97%
2	0.1	Auto	99,97%
3	1	Scale	99,98%
4	1	Auto	99,98%
5	10	Scale	99,98%
6	10	Auto	99,98%

Pada tabel 9 bisa dilihat bahwsanya hasil dari *hyperparameter* mempunyai nilai yang berbeda dan nilai terbaik didalam beberapa penggunaan *hyperparameter* diatas yaitu penggunaan C: 1’, *gamma* = Scale dengan nilai 99.98%

Pada tahap *classification report, kernel* rbf mempunyai nilai yang sama seperti kedua kernel sebelumnya yang berarti menunjukan hasil yang sempurna.

p-ISSN: 2301-5373
e-ISSN: 2654-5101



Gambar 3. ROC rbf

Pada gambar 3 bisa dilihat bahwsanya nilai ROC AUC pada *kernel* rbf yaitu 1.0 yang berarti model mempunyai kapabilitas yang sempurna guna membedakan diantara kelas 0 dan 1 seperti *kernel linear*.

Pengujian selanjutnya yaitu dilaksanakannya proses evaluasi model dengan berbagai macam *kernel* namun pada pengujian ini penulis akan mempergunakan seleksi fitur *Correlation-Based Feature Selection* dan fitur yang dipilih yaitu sebanyak 31 fitur karena mempunyai nilai relasi diatas 0.1. Fitur – fitur.

Evaluasi model yang dihasilkan dari berbagai *kernel support vector machine* dibantu dengan *correlation-based featured selection* yaitu sebagai berikut :

a. Hasil evaluasi model pada *kernel linear* dengan cfs

Tabel 10. Evaluasi Model Linear dengan CFS

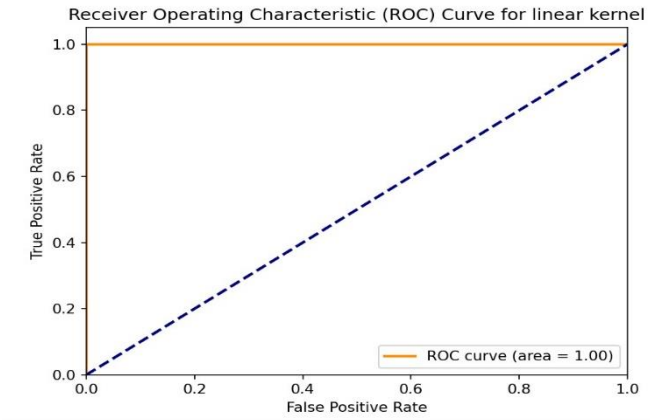
```
Best hyperparameters found for linear kernel: {'C': 0.1}
Accuracy for linear kernel: 0.9998039215686274
Confusion Matrix:
[[3252    1]
 [   0 1847]]

Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      3253
           1       1.00      1.00      1.00      1847

    accuracy                           1.00      5100
   macro avg       1.00      1.00      1.00      5100
weighted avg       1.00      1.00      1.00      5100


ROC AUC Score for linear kernel: 1.0
```



Gambar 4. ROC Linear dengan CFS

b. Hasil evaluasi model pada kernel polynomial dengan cfs

Tabel 11. Evaluasi Model Polynomial dengan CFS

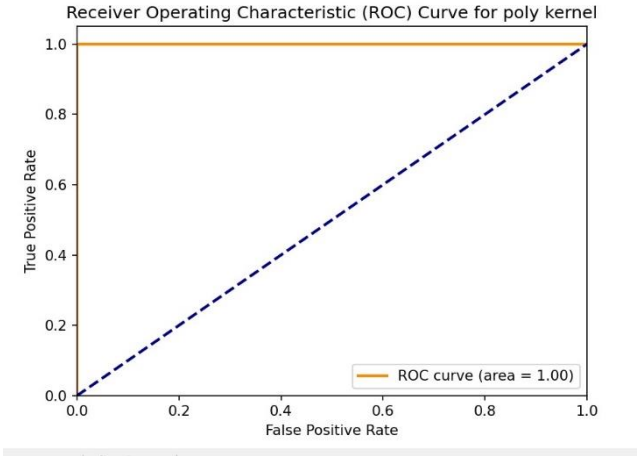
```
Best hyperparameters found for poly kernel: {'C': 0.1, 'degree': 3, 'gamma': 'scale'}
Accuracy for poly kernel: 0.9998039215686274
Confusion Matrix:
[[3252    1]
 [   0 1847]]

Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      3253
           1       1.00      1.00      1.00      1847

    accuracy                           1.00      5100
   macro avg       1.00      1.00      1.00      5100
weighted avg       1.00      1.00      1.00      5100


ROC AUC Score for poly kernel: 1.0
```



Gambar 5. ROC Polynomial dengan CFS

p-ISSN: 2301-5373
e-ISSN: 2654-5101

c. Hasil evaluasi model pada kernel rbf dengan cfs

Tabel 12. Evaluasi Model RBF dengan CFS

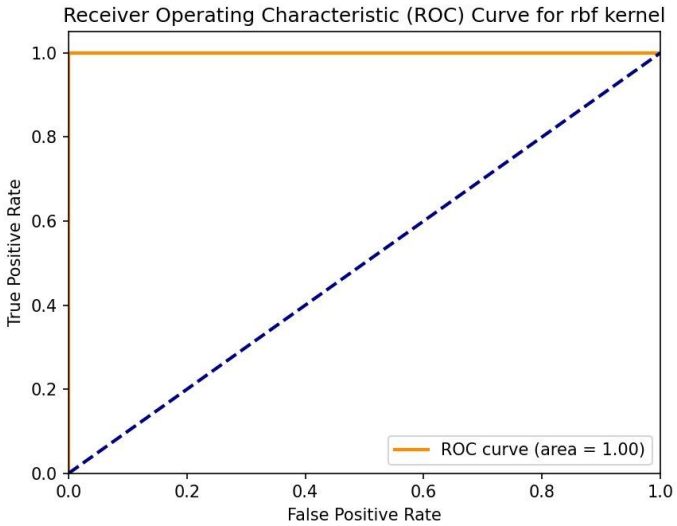
```
Training SVM with rbf kernel
Best parameters found for rbf kernel: {'C': 1}
Accuracy for rbf kernel: 0.9995833333333334
Confusion Matrix:
[[1555    0]
 [   1  844]]

Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      1555
           1       1.00      1.00      1.00       845

    accuracy                           1.00      2400
   macro avg       1.00      1.00      1.00      2400
weighted avg       1.00      1.00      1.00      2400


ROC AUC Score for rbf kernel: 1.0
```

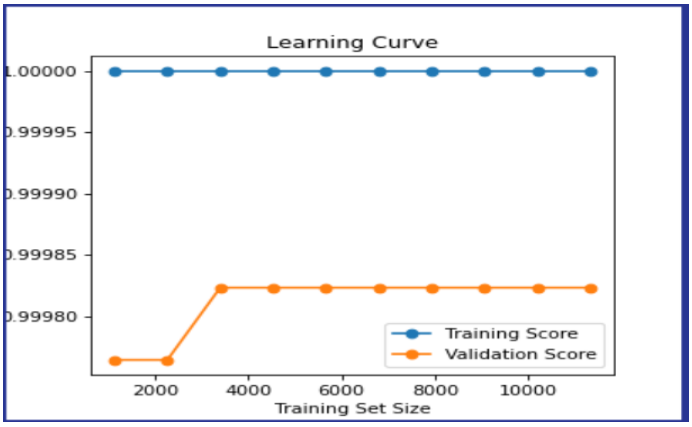


Gambar 6. ROC RBF dengan CFS

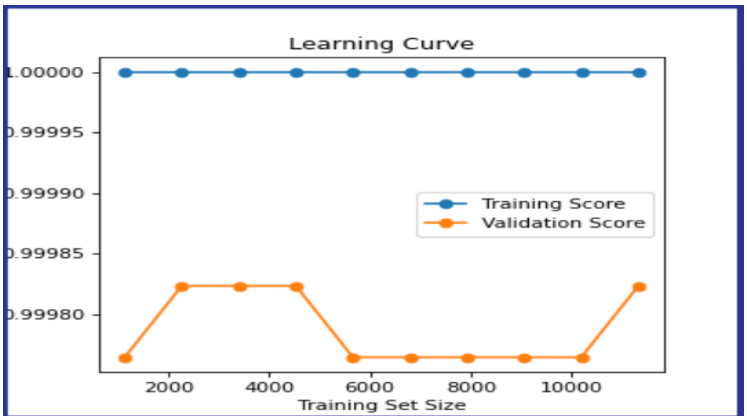
4. Hasil dan Pembahasan

Terlihat grafik yang memperlihatkan nilai *accuracy* dari training score dan validation score. Nilai pada training score menyatakan 1.0 yang berarti pelatihan model menunjukan bahwsanya model sangat baik didalam mengenali pola dari data yang dilihat dan dipelajarin selama proses pelatihan. Selanjutnya nilai pada validation score menunjukan nilai mulai dari 0.9980 lalu naik ke 0.9985 yang berarti kapabilitas model didalam mengevaluasi data dengan data baru sangat baik. Bisa dilihat pada gambar 7 dibawah ini

Gambar 7. Learning Curve Linear



Pada gambar 8 terlihat grafik yang memperlihatkan nilai *accuracy* dari training score dan validation score. Nilai pada training score menyatakan 1.0 yang berarti pelatihan model menunjukan bahwsanya model sangat baik didalam mengenali pola dari data yang dilihat dan dipelajarin selama proses pelatihan. Selanjutnya nilai pada validation score menunjukan nilai mulai dari 0.9973 lalu naik ke 0.9983 lalu turun dan naik kembali yang berarti kapabilitas model didalam mengevaluasi data dengan data baru cukup baik walaupun terjadi ketidakstabilan didalam nilai accuracy namun masih bisa diterima karena nilainya masih tergolong tinggi.



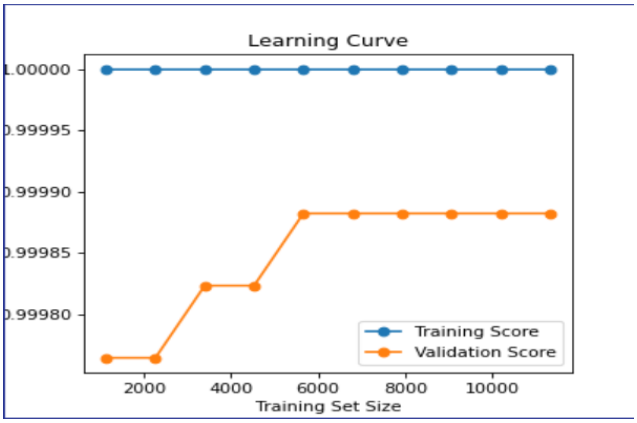
Gambar 8. Learning Curve Polynomial

Pada gambar 9 terlihat grafik yang memperlihatkan nilai *accuracy* dari training score dan validation score. Nilai pada training score menyatakan 1.0 yang berarti pelatihan model menunjukan bahwsanya model sangat baik didalam mengenali pola dari data yang dilihat dan dipelajarin selama proses pelatihan. Selanjutnya nilai pada validation score menunjukan nilai mulai dari 0.9975 lalu naik menjadi 0.9984 dan terus naik sampai 0.9990 yang berarti model sangat baik didalam mempergunakan data baru. Dan model semakin mampu mengenali pola yang relevan didalam data

Dilihat dari ketiga grafik diatas model terbaik yang bisa disimpulkan yaitu model rbf karena ketika model dipergunakan pada data baru model rbf mempunyai nilai yang perlahan naik yang berarti model rbf bisa beradaptasi didalam mengenali pola yang berbeda.

p-ISSN: 2301-5373
e-ISSN: 2654-5101

Gambar 9. Learning Curve rbf



3 Kesimpulan

Kesimpulan yang bisa diambil dari implementasi metode klasifikasi Support Vector Machine (SVM) dan seleksi fitur Correlation-Based Feature Selection (CFS) didalam klasifikasi serangan Distributed Denial of Service (DDoS) mempergunakan dataset CSE-CIC-IDS2018 yaitu sebagai berikut:Semua kernel bekerja sangat baik pada dataset ini dengan model training mencapai akurasi 1.0 namun guna model testing memperoleh nilai akurasi 99.98%

1. Penerapan CFS didalam kasus ini tidak memperlihatkan peningkatan yang berarti didalam performa model SVM. Namun, CFS bermanfaat guna menyederhanakan model dan mengurangi jumlah fitur yang dipergunakan tanpa mengorbankan kinerja.
2. Dari semua kernel baik mempergunakan CFS ataupun tidak kernel RBF menjadi pilihan terbaik karena saat menguji guna data yang baru kernel RBF mempunyai kapabilitas yang bisa memahami model dengan baik yang membuat dia memperoleh nilai akurasi yang terus bertambah disetiap pengujian dan memperoleh nilai akurasi sebanyak 99.98%

Referensi

[1] M. A. Ridho and M. Arman, “Analisis Serangan DDoS Mempergunakan Metode Jaringan Saraf Tiruan,” *J. Sisfokom (Sistem Inf. dan Komputer)*, vol. 9, no. 3, pp. 373–379, Oct. 2020, doi: 10.32736/sisfokom.v9i3.945.

[2] B. Hijriyanto and F. Ulum, “Comparison of Mod_Evasive and DDoS Deflate for Slow Post Attack Mitigation,” 2021.

[3] A. S. B. Asmoro, W. S. G. Irianto, and U. Pujianto, “Perbandingan Kinerja Hasil Seleksi Fitur pada Prediksi Kinerja Akademik Siswa Berbasis Pohon Keputusan,” *J. Edukasi dan Penelit. Inform.*, vol. 4, no. 2, pp. 84–89, 2018.

[4] A. T. Zy, A. T. Sasongko, and A. Z. Kamalia, “Penerapan Naïve Bayes Classifier, Support Vector Machine, dan Decision Tree guna Menaikkan Deteksi Ancaman Keamanan Jaringan,” *Media Online)*, vol. 4Zy, A. T., no. 1, pp. 610–617, 2023, doi: 10.30865/klik.v4i1.1134.

[5] A. Basuki and F. Abdurrachman Bachtiar, “METODE DETEKSI INTRUSI MEMPERGUNAKAN ALGORITME EXTREME LEARNING MACHINE DENGAN CORRELATION-BASED FEATURE SELECTION,” vol. 8, no. 1, pp. 103–110, 2021, doi: 10.25126/jtiik.202183358.

This page is intentionally left blank.