InComTech: Jurnal Telekomunikasi dan Komputer
vol.15, no.01, April 2025, 41-55
http://publikasi.mercubuana.ac.id/index.php/Incomtech
P-ISSN: 2085-4811 E-ISSN: 2579-6089

Perbandingan Algoritma *Decision Tree* dan *Gradient Boosting* pada Model Sistem Deteksi Serangan Siber di Jaringan *Internet of Things*

M. Agus Syamsul Arifin[1*], Armanto [2], Susanto [3], A. Taqwa Martadinata [4]

[1] *Informatika, Universitas Jenderal Soedirman,*
Jl. Raya Mayjen Sungkono, Kabupaten Purbalingga, Jawa Tengah 53371, Indonesia
[2] *Rekayasa Sistem Komputer, Universitas Bina Insan,*
Jl. HM. Soeharto, Lubuklinggau 31620, Indonesia
[3] *Sistem Informasi, Universitas Universitas Pembangunan Nasional Veteran,*
Jl. R.S Fatmawati No. 1, Cilandak, Jakarta Selatan 12450, Indonesia
[4] *Informatika, Universitas Bina Insan,*
Jl. HM. Soeharto, Lubuklinggau 31620, Indonesia
*Email : mas.agus1988@gmail.com

***Abstrak (TNR 10 Bold Italic)*:**
Internet of things (IoT) memberikan banyak manfaat dimana membuat banyak perangkat pintar semakin dekat dan mudah digunakan. Penerapan teknologi IoT yang semakin luas memberikan banyak ancaman bari dalam segi keamanan data karena banyak perangkat yang terhubung dengan protocol yang beragam untuk mengatasinya dibutuhkan sebuah *Intrusion Detection System* (IDS) yang handal untuk mendeteksi serangan dalam jaringan IoT. Dalam penelitian ini akan membangun sebuah model IDS menggunakan algoritma *decision tree* dan *gradient boosting* kemudian membandingkan performanya. Dataset yang digunakan pada penelitian ini menggunakan dataset dari CICIoT2023 karena kelas yang tidak seimbang dan ukuran dataset yang besar teknik *Random Under Sampling* (RUS) digunakan juga dalam penelitian ini. Hasil dari penelitian menunjukkan performa yang baik untuk setiap model IDS yang dibuat. Pada data latih, algoritma Decision Tree menunjukkan akurasi yang sangat tinggi, yaitu mencapai 100%, baik sebelum maupun sesudah penerapan teknik RUS. Namun, pada data uji, algoritma Gradient Boosting menghasilkan akurasi yang lebih baik, yakni 99,10% sebelum penerapan teknik RUS, dan menurun menjadi 76,31% setelah teknik RUS diterapkan.

This is an open access article under the CC BY-NC license

***Kata Kunci*:**
Decision Tree;
Gradient Boosting ;
Internet of Things (IoT);
Intrusion Detection System (IDS);
Random Under Sampling (RUS);

Riwayat Artikel:
Diserahkan 06 Maret 2024
Direvisi 25 April 2025
Diterima 02 Mei 2025

DOI:
10.22441/incomtech.v15i1.26096

1. PENDAHULUAN (10%)

Internet of things (IoT) memberikan banyak manfaat dimana membuat banyak perangkat pintar semakin dekat dan mudah digunakan. Penerapan IoT yang luas mencakup bidang kesehatan [1], pertanian [2] bahkan penerapannya juga diadopsi untuk mempermudah memonitoring keamanan dan kondisi rumah tempat tinggal [3],[4]. Penerapan teknologi IoT yang semakin luas memberikan banyak ancaman bari dalam segi keamanan data karena banyak perangkat yang terhubung dengan protocol yang beragam. Keamanan siber adalah teknologi yang dikembangkan untuk melindungi struktur fisik, perangkat lunak, dan data dalam jaringan dari akses yang tidak sah atau perubahan bentuk [5].

Pengamanan komuniakasi data pada jaringan IoT menjadi tujuan penting pada jaringan IoT. Metode enkripsi [6] digunakan untuk mengamankan isi informasi pada komunikasi perangkat IoT, teknologi *blockchain* [7] juga disematkan ke dalam IoT untuk menjamin perangkat yang terhubung dan informasi yang dikirimkan valid. Teknik pengamanan tersebut untuk mengatasi penyerang yang sudah berada dalam jaringan atau *Man in the Middle* (MiTM) namun dengan mendeteksi serangan tersebut sebelum terjadi sangat penting untuk mengurangi kerugian yang mungkin di sebabkan oleh serangan MiTM. Dalam penelitian ini akan membahas penerapan teknologi *machine learning* (ML) untuk membangun model *Intrusion detection system* (IDS) dengan menggunakan algoritma *Decision Tree* (DT) dan *Gradient Boosting* (GB) dalam mendeteksi aktifitas jahat pada jaringan IoT dan membandingkan performa kedua algoritma *machine learning* tersebut.

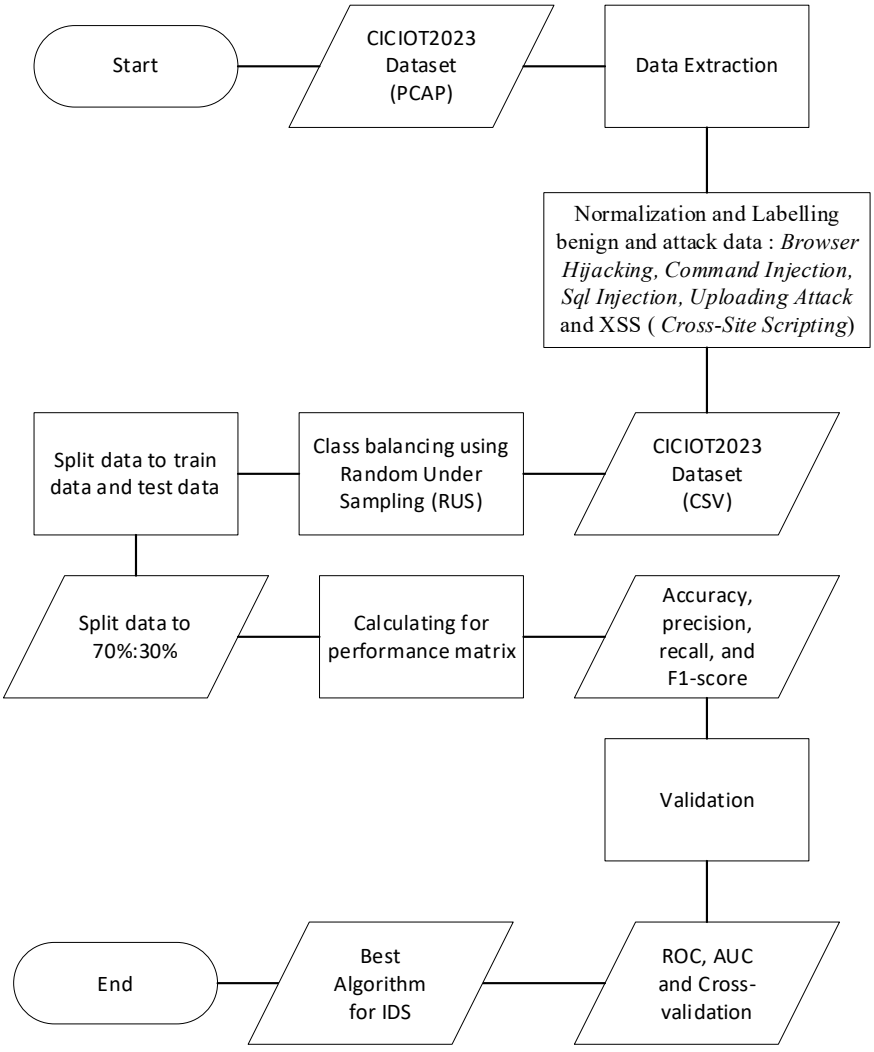
Penelitian ini akan menggunakan dataset CICIoT2023 [8] sebagai bahan untuk melatih model *machine learning* yang akan dibuat, dataset ini dipilih karena dibuat menggunakan perangkat IoT nyata sehingga model yang dihasilkan akan mendekati hasil sebenarnya dalam mendeteksi serangan siber pada jaringan IoT. Percobaan yang dilakukan dalam penelitian ini adalah dengan mendeteksi serangan *Browser Hijacking, Command Injection, Sql Injection, Uploading Attack* dan XSS (*Cross-Site Scripting*) pada dataset. Kontribusi dari penelitian ini yaitu memberikan hasil perbandingan performa model *machine learning* dalam mendeteksi serangan dalam dataset menggunakan algoritma *Decision Tree* dan *Gradient Boosting* dan mendapatkan algoritma terbaik untuk mendeteksi serangan pada jaringan IoT pada dataset CICIoT2023.

2. METODE

Penelitian ini bersifat eksperimental dengan menguji performa algoritma *Decision Tree* dan *Gradient Boosting* dalam melakukan klasifikasi (deteksi) serangan pada dataset CICIoT2023. Pemilihan kedua algoritma ini didasarkan pada keunggulan masing-masing dalam menangani masalah klasifikasi. *Decision Tree* dikenal dengan interpretabilitasnya yang tinggi dan kemampuannya menangani data non-linear, sedangkan *Gradient Boosting* dipilih karena kemampuannya yang unggul dalam meningkatkan akurasi prediksi melalui pendekatan ansambel, sehingga cocok untuk mendeteksi pola serangan kompleks pada jaringan IoT.

2.1 Metode dan Dataset CICIoT2023

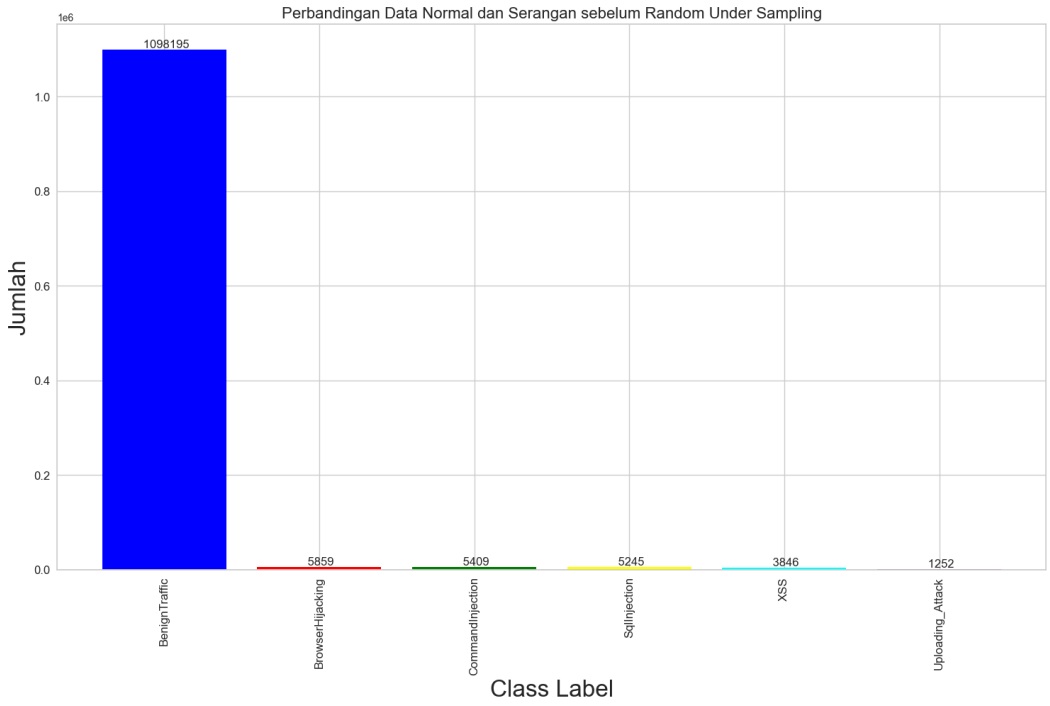
Penelitian ini menggunakan dataset CICIoT2023 yang memiliki perangkat nyata IoT didalam testbed yang digunakan [8]. Dataset ini berisi berbagai skenario serangan pada sistem jaringan IoT, namun penelitian ini hanya akan membahas serangan siber pada jaringan IoT dengan tipe serangan *Browser Hijacking*, *Command Injection, SQL Injection, Uploading Attack, dan XSS (Cross-Site Scripting)*. Pemilihan lima jenis serangan ini didasarkan pada tingkat prevalensi dan dampaknya yang signifikan terhadap kerahasiaan, integritas, dan ketersediaan sistem IoT. Selain itu, kelima serangan tersebut mewakili beragam vektor serangan yang umum terjadi pada aplikasi dan perangkat IoT, sehingga relevan untuk dianalisis lebih dalam dalam konteks deteksi ancaman siber. Gambar 1 berikut menunjukkan alur yang digunakan untuk mencari performa model IDS terbaik antara *Decision tree* dan *Gradient boosting*.



Gambar 1. Metode yang digunakan untuk menentukan algoritma terbaik dalam mendeteksi serangan pada dataset CICIoT2023

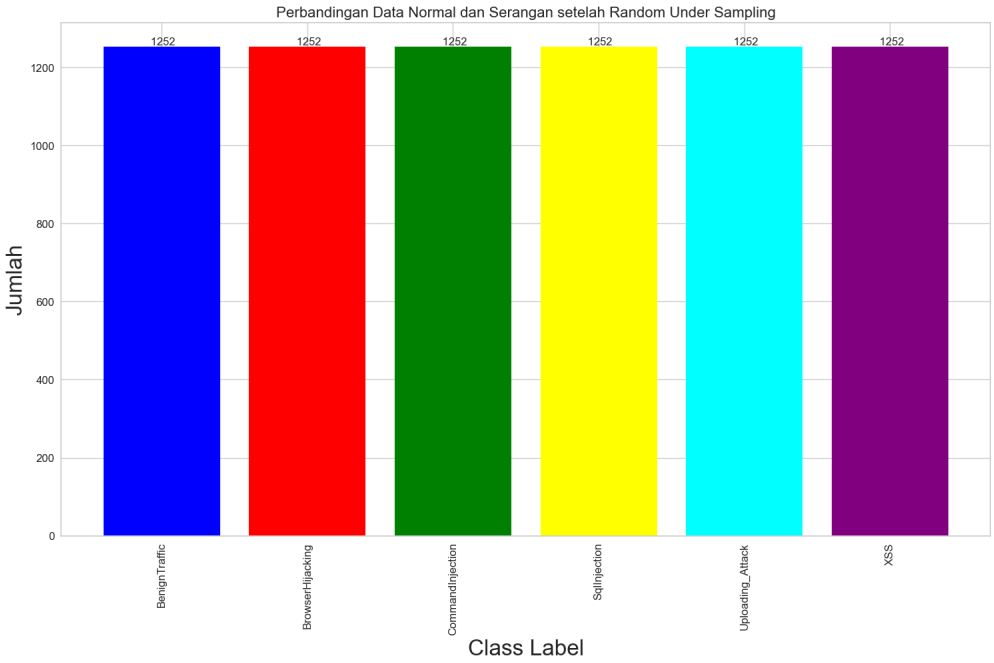
Metode yang dilakukan pada penelitian ini yaitu mengolah dataset CICIoT2023 yang memiliki format data pcap dengan melakukan proses ekstraksi fitur. Setelah

fitur-fitur lalu lintas jaringan berhasil diekstrak, dilakukan proses pelabelan dengan mencocokkan waktu dan skenario serangan yang telah didokumentasikan oleh penyedia dataset. Lalu lintas jaringan yang tidak termasuk dalam rentang waktu serangan diberi label sebagai data normal, sedangkan lalu lintas yang terjadi selama skenario serangan dijalankan diberi label sesuai dengan jenis serangannya. Data yang telah diekstrak dan dilabeli kemudian disimpan dalam format CSV (Comma Separated Values) untuk proses analisis lebih lanjut. Gambar 2 berikut menunjukkan jumlah data dan kelas dalam dataset CICIoT2023.



Gambar 2. Perbandingan data setiap kelas dalam dataset CICIoT 2023 untuk MiTM

Karena jumlah kelas data normal dan serangan tidak seimbang maka peneliti menggunakan metode *Random Under Sampling* (RUS) untuk menyeimbangkan jumlah kelas data normal dan serangan pada dataset berdasarkan dengan jumlah kelas terkecil. Gambar 3 berikut menunjukkan jumlah data dalam setiap kelas setelah proses RUS.



Gambar 3. Perbandingan data setiap kelas dalam dataset CICIoT 2023 untuk MiTM setelah proses *Random Under Sampling*

Setelah proses RUS kami membagi dataset menjadi data latih (70%) dan data uji (30%), kemudian mengukur performa model yang diuji berdasarkan akurasi, presisi, *recall* dan nilai F1-score. Metrik untuk mengukur performa kinerja model yang dibuat memiliki persamaan (1) – (4).

$$\text{Accuracy} = \frac{(\text{TN}+\text{TP})}{(TN+TP+FN+FP)} \quad (1)$$

$$\text{Precision} = \frac{\text{TP}}{(TP+FP)} \quad (2)$$

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

$$\text{F1 Measure} = 2\frac{(\text{Precision x Recall})}{(Precision+Recall)} \quad (4)$$

Pada proses validasi penelitian ini menggunakan kurva ROC (*Receiver Operating Characteristic*) dan nilai AUC (*Area Under the Curve*) untuk melihat kemampuan model dalam mendeteksi serangan dalam dataset [9],[10] selanjutnya *cross-validation* (CV) juga digunakan dalam penelitian ini untuk melihat apakah model yang dibuat terdapat indikasi *overfitting* [11] dengan menggunakan 10-fold, CV akan mengacak setiap *sample* untuk setiap repetisi dengan data yang relatif sama [12], kemudian dari hasil pengujian, pengukuran dan validasi tersebut baru dapat ditentukan algoritma terbaik dalam mendeteksi serangan pada dataset CICIoT2023.

2.2. Algoritma klasifikasi

Penelitian ini menggunkan algoritma *Decision tree* dan *Gradient boosting* yang merupakan algoritma dalam machine learning bersifat *supervised*. Decision tree adalah teknik dengan basis pohon dimana setiap jalur dimulai dari akar yang dijelaskan oleh urutan pemisahan data hingga hasil pada simpul daun tercapai dimana relasi digunakan untuk klasifikasi, simpul mewakili tujuan [13]. Gradient boosting menciptakan model baru dari model ensemble lemah dengan ide setiap model baru dapat meminimalkan *loss function* secara teratur. *Loss function* ini diukur dengan metode pengurangan *gradient*. Dengan menggunakan *loss function*, setiap model baru akan lebih sesuai dengan observasi, dan dengan demikian akurasi secara keseluruhan akan meningkat [14]. Dalam penelitian ini kedua algoritma ini akan di bandingkan untuk melihat algoritma mana yang memeiliki performa terbaik dalam mendeteksi serangan dalam di dalam dataset.

2.3. Random Under Sampling (RUS)

Teknik RUS adalah teknik dengan pengambilan sampel acak untuk menyeimbangkan kelas yang diinginkan dengan menghilangkan contoh dari kelas mayoritas [15],[16]. Dalam penelitian ini teknik RUS digunakan untuk menyeimbangkan kelas sehingga performa yang dihasilkan lebih *valid* karena kelas yang seimbang, selain itu teknik ini digunakan untuk membuat ukuran data latih lebih kecil sehingga waktu *training* yang dilakukan untuk melatih model dalam mendeteksi serangan dalam dataset menjadi lebih singkat dibandingkan dengan sebelum penerapan teknik RUS pada dataset.

3. HASIL DAN PEMBAHASAN

Jumlah data pada dataset setelah proses normalisasi adalah 1.119.805 dengan data normal (*benign*) berjumlah 1.098.195 data, serangan *BrowserHijacking* 5.859 data, *CommandInjection* 5.409 data, *SqlInjection* 5.245 data dan XSS 1.252. Setelah penerapan teknik *Random Under Sampling* berjumlah 7.512 dengan jumlah data setiap kelas sama yaitu 1.252. Sub bab 3.1 akan menjabarkan hasil dari performa model IDS dengan *Decision tree* dan *Gradient boosting* yang menggunakan dataset sebelum diterapkan teknik RUS dan sub bab 3.2 akan menjabarkan hasil dari performa model IDS setelah penerapan proses RUS pada dataset.

3.1. Perbandingan model IDS tanpa teknik RUS

Penelitian model IDS dengan algoritma *Decision tree* dan *Gradient boosting* pada sudah dilakukan diantaranya penelitian yang dilakukan oleh M. Alshamkhany, et.al [17] menggunakan algoritma DT untuk mendeteksi serangan botnet pada jaringan computer, kemudian penelitian yang dilakukan oleh K. Alissa, et.al [18] mendeteksi serangan botnet pada jaringan IoT dengan menggunakan algoritma DT. Penelitian IDS menggunakan algoritma GB dilakukan oleh D. Upadhyay, et.al [19] untuk seleksi fitur dalam mendeteksi serangan dalam jaringan SCADA dan penelitian yang dilakukan oleh B. Tama [20] menggunakan GB untuk mendeteksi anomaly dalam dataset NSL-KDD. Dari penelitian yang sudah dilakukan penelitia sebelumnya dengan menggunakan algoritman DT dan GB menunjukkan kedua algoritma ini populer dalam penelitian IDS menggunakan machine learning. Tabel

1 berikut menunjukkan perbandingan performa model IDS yang menggunakan algoritma DT dan GB pada data latih dan table 2 menggunakan data uji.

Tabel 1. Perbandingan performa model IDS dengan algoritma *Decision tree* dan *Gradient boosting* sebelum penerapan teknik *Random Under Sampling* pada data latih

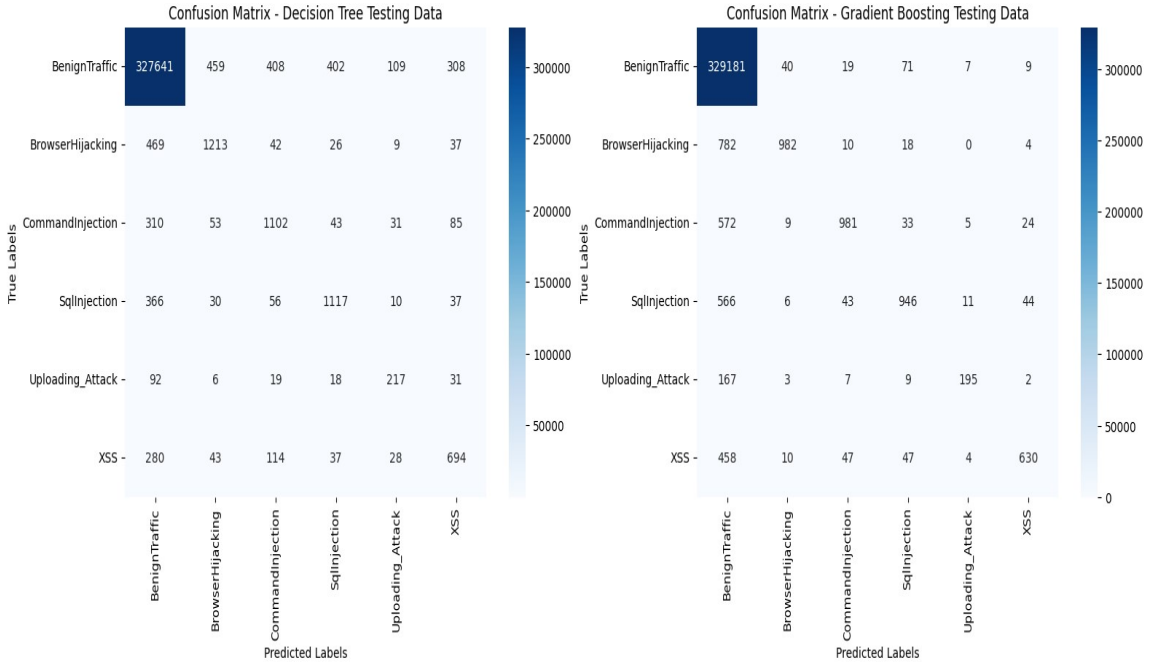
Model	Evaluation	Performance			
		Accuracy	Precision	Recall	F1-Score
Decision Tree	BenignTraffic	100%	1,00	1,00	1,00
	Browser Hijacking		1,00	1,00	1,00
	Command Injection		1,00	1,00	1,00
	SqlInjection		1,00	1,00	1,00
	Uploading Attack		1,00	1,00	1,00
	XSS		1,00	1,00	1,00
Gradient Boosting	BenignTraffic	99,18%	0,99	1,00	1,00
	Browser Hijacking		0,95	0,56	0,70
	Command Injection		0,90	0,61	0,72
	SqlInjection		0,87	0,62	0,72
	Uploading Attack		0,94	0,57	0,71
	XSS		0,92	0,58	0,71

Tabel 2. Perbandingan performa model IDS dengan algoritma *Decision tree* dan *Gradient boosting* sebelum penerapan teknik *Random Under Sampling* pada data uji

Model	Evaluation	Performance			
		Accuracy	Precision	Recall	F1-Score
Decision Tree	BenignTraffic	98,82%	1,00	0,99	1,00
	Browser Hijacking		0,67	0,68	0,67
	Command Injection		0,63	0,68	0,65
	SqlInjection		0,68	0,69	0,69
	Uploading Attack		0,54	0,57	0,55
	XSS		0,58	0,58	0,58
Gradient Boosting	BenignTraffic	99,10%	0,99	1,00	1,00
	Browser Hijacking		0,94	0,55	0,69
	Command Injection		0,89	0,60	0,72
	SqlInjection		0,84	0,59	0,69
	Uploading Attack		0,88	0,51	0,64
	XSS		0,88	0,53	0,66

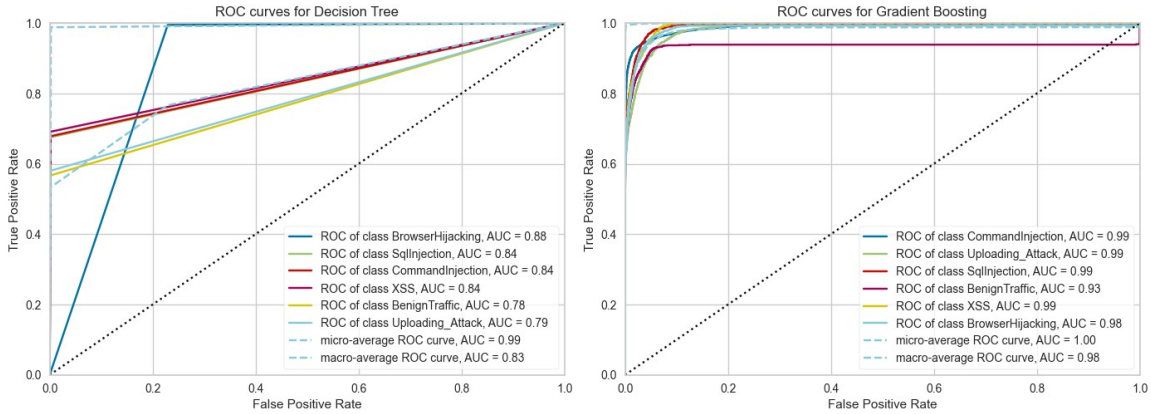
Gambar 4 berikut menunjukkan *confusion matrix* model IDS ketika menggunakan data latih sedangkan Gambar 5 menunjukkan *confusion matrix* model IDS ketika menggunakan data uji.

Gambar 4. *Confusion matrix* untuk model IDS dengan menggunakan data latih

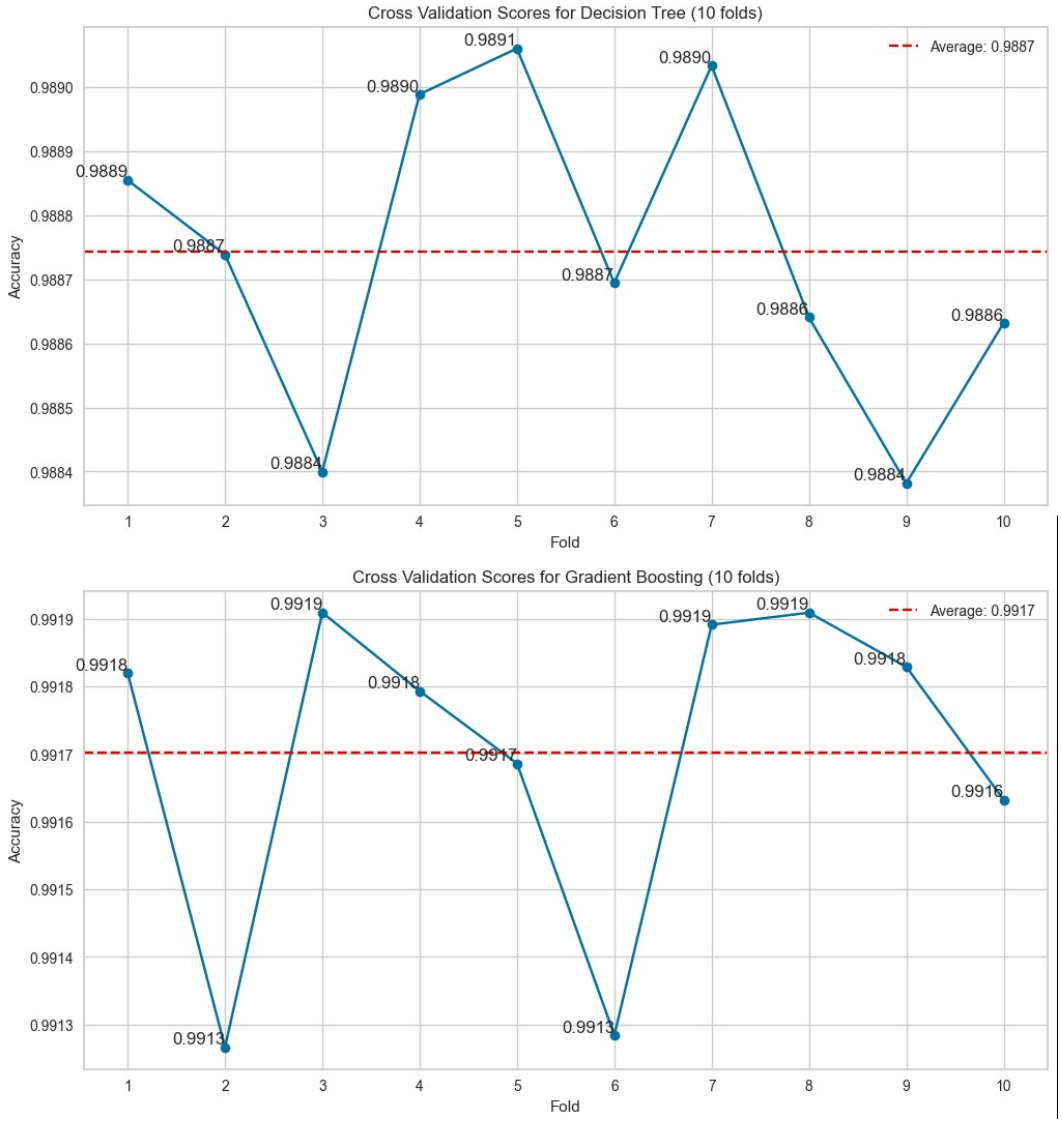


Gambar 5. *Confusion matrix* untuk model IDS dengan menggunakan data uji

Gambar 6 berikut menunjukkan kurva ROC dan nilai AUC model IDS dan Gambar 7 menunjukkan hasil *cross-validation* dari model IDS.



Gambar 6. Perbandingan Kurva ROC dan nilai AUC untuk model IDS yang di uji



Gambar 7. Perbandingan hasil *cross-validation* untuk model IDS yang di uji

3.2. Perbandingan hasil model IDS dengan teknik *Random Under Sampling*

Dalam penelitian ini teknik *Random Under Sampling* digunakan untuk membuat setiap kelas pada dataset menjadi seimbang sehingga hasil yang di dapatkan dari model IDS yang di bangun menjadi lebih relevan. Penelitian dengan teknik RUS untuk membuat model IDS sudah dilakukan oleh Mas. Arifin, et.al [21] dengan membandingkan beberapa teknik untuk menyeimbangkan jumlah data pada setiap kelas di dalam dataset dan teknik RUS mendapatkan hasil yang terbaik untuk deteksi serangan pada jaringan SCADA, kemudian penelitian yang dilakukan oleh H. Yasir, et.al [22] untuk system deteksi serangan siber dalam jaringan komputer dengan model *neural network*. Tabel 3 berikut menunjukkan perbandingan performa model IDS setelah penerapan teknik *Random under sampling* menggunakan algoritma DT dan GB pada data latih dan table 4 menggunakan data uji.

Tabel 3. Perbandingan performa model IDS dengan algoritma *Decision tree* dan *Gradient boosting* setelah penerapan teknik *Random Under Sampling* pada data latih

Model	Evaluation	Performance			
		Accuracy	Precision	Recall	F1-Score
Decision Tree	BenignTraffic	100%	1,00	1,00	1,00
	Browser Hijacking		1,00	1,00	1,00
	Command Injection		1,00	1,00	1,00
	SqlInjection		1,00	1,00	1,00
	Uploading Attack		1,00	1,00	1,00
	XSS		1,00	1,00	1,00
Gradient Boosting	BenignTraffic	88,91%	0,98	0,96	0,97
	Browser Hijacking		0,93	0,88	0,90
	Command Injection		0,93	0,81	0,87
	SqlInjection		0,89	0,92	0,90
	Uploading Attack		0,75	0,91	0,82
	XSS		0,90	0,85	0,87

Tabel 2. Perbandingan performa model IDS dengan algoritma *Decision tree* dan *Gradient boosting* setelah penerapan teknik *Random Under Sampling* pada data uji

Model	Evaluation	Performance			
		Accuracy	Precision	Recall	F1-Score
Decision Tree	BenignTraffic	72,54%	0,91	0,89	0,90
	Browser Hijacking		0,73	0,72	0,72
	Command Injection		0,63	0,62	0,62
	SqlInjection		0,75	0,77	0,76
	Uploading Attack		0,65	0,69	0,67
	XSS		0,67	0,66	0,66
Gradient Boosting	BenignTraffic	76,31%	0,96	0,93	0,95
	Browser Hijacking		0,84	0,74	0,79
	Command		0,75	0,60	0,67

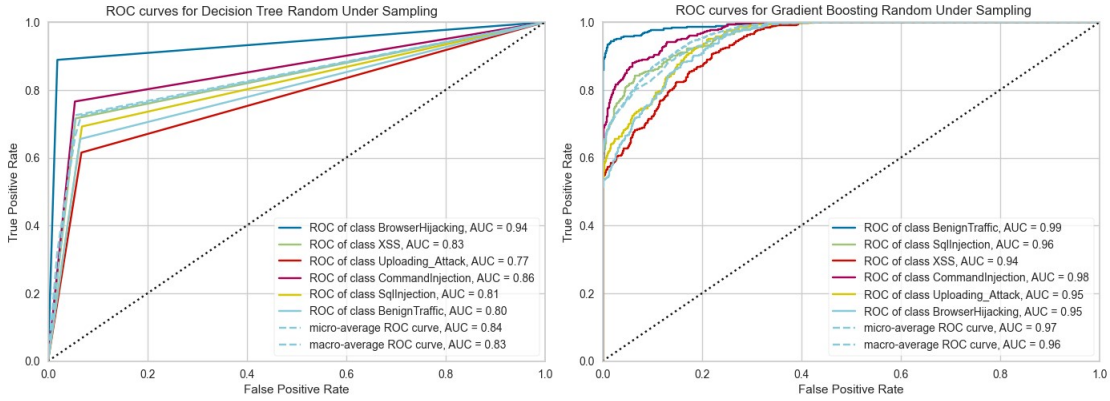
	Injection			
	SqlInjection	0,78	0,84	0,81
	Uploading Attack	0,57	0,78	0,66
	XSS	0,72	0,67	0,69

Gambar 8 berikut menunjukkan *confusion matrix* model IDS setelah penerapan teknik *Random under sampling* ketika menggunakan data latih sedangkan Gambar 9 menunjukkan *confusion matrix* model IDS ketika menggunakan data uji.

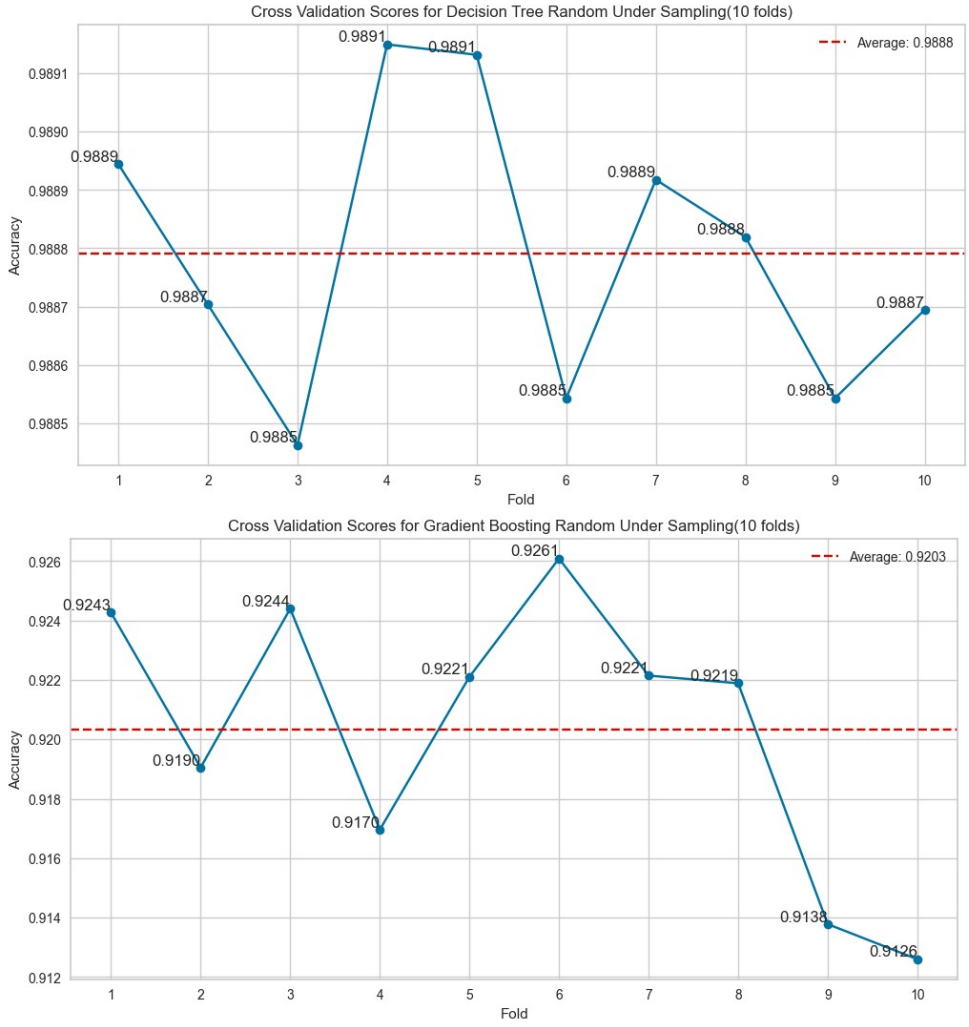
Gambar 8. *Confusion matrix* model IDS setelah penerapan teknik *Random under sampling* dengan menggunakan data latih

Gambar 9. *Confusion matrix* model IDS setelah penerapan teknik *Random under sampling* dengan menggunakan data uji

Gambar 10 berikut menunjukkan kurva ROC dan nilai AUC model IDS setelah penerapan teknik *Random under sampling* dan Gambar 11 menunjukkan hasil *cross-validation* dari model IDS setelah penerapan teknik *Random under sampling*.



Gambar 10. Perbandingan Kurva ROC dan nilai AUC untuk model IDS yang di uji setelah penerapan teknik *Random under sampling*



Gambar 11. Perbandingan hasil *cross-validation* untuk model IDS yang di uji setelah penerapan teknik *Random under sampling*

3.3. Pembahasan

Teknik klasifikasi dalam *machine learning* sangat membutuhkan dataset yang ideal dengan pembagian jumlah data pada setiap kelas yang seimbang pada dataset yang digunakan karena mempengaruhi performa dan kepercayaan untuk model yang dibangun [23]. Dari pengujian yang sudah dilakukan pada penelitian ini dimana data dari kelas *benign* memiliki jumlah mayoritas sehingga dapat mempengaruhi skor performa yang di dapatkan pada setiap model IDS yang diujikan baik menggunakan algoritma DT maupun GB.

Decision tree mendapatkan hasil yang sangat tinggi ketika menggunakan data latih dengan akurasi 100% baik sebelum menerapkan teknik RUS maupun belum menerapkan teknik RUS, namun untuk data uji dimana data yang diujikan tidak dimasukkan dalam data latih algoritma Gradient boosting mendapatkan hasil yang lebih baik dari Decision tree ini menunjukan algoritma Gradient boosting merupakan model IDS yang lebih baik untuk mendeteksi serangan yang belum dipelajari sebelumnya dibandingkan dengan model IDS dengan algoritma decision tree.

Kurva ROC dan nilai AUC menunjukkan hasil yang baik ketika tidak menggunakan teknik RUS maupun menggunakan teknik RUS. Dari hasil validasi menggunakan cross-validation menunjukkan tidak terdapat gejala overfitting dari model yang dibangun sehingga model IDS yang dibangun merupakan model yang valid untuk akurasinya baik ketika diterapkan teknik RUS maupun tidak.

4. KESIMPULAN

Berdasarkan hasil penelitian, dapat disimpulkan bahwa keseimbangan jumlah data pada setiap kelas dalam dataset sangat berpengaruh terhadap performa dan reliabilitas model IDS yang dibangun. Ketimpangan kelas, khususnya dominasi kelas benign, menyebabkan model cenderung memiliki bias terhadap kelas mayoritas, sebagaimana terlihat pada hasil akurasi 100% yang dicapai oleh algoritma Decision Tree pada data latih, baik sebelum maupun sesudah penerapan teknik Random Under Sampling (RUS). Namun, hasil tersebut tidak sepenuhnya mencerminkan kemampuan generalisasi model.

Pengujian pada data uji menunjukkan bahwa algoritma Gradient Boosting memberikan hasil yang lebih baik dibandingkan Decision Tree, khususnya dalam mendeteksi serangan yang tidak terdapat dalam data latih. Hal ini menunjukkan bahwa Gradient Boosting lebih unggul dalam hal kemampuan generalisasi terhadap serangan baru.

Selain itu, kurva ROC dan nilai AUC menunjukkan performa yang baik pada kedua algoritma, baik sebelum maupun sesudah penerapan teknik RUS. Hasil validasi dengan metode cross-validation juga menunjukkan bahwa tidak terdapat indikasi overfitting, sehingga model IDS yang dibangun dapat dikatakan valid dan andal, baik dengan maupun tanpa teknik RUS.

Dengan demikian, penerapan teknik balancing seperti RUS tetap penting untuk mengurangi bias, dan pemilihan algoritma yang memiliki kemampuan adaptasi yang baik seperti Gradient Boosting dapat meningkatkan efektivitas sistem deteksi intrusi, khususnya pada lingkungan jaringan IoT yang dinamis dan kompleks.

REFERENSI

[1] F. Ahamed and F. Farid, "Applying internet of things and machine-learning for personalized healthcare: Issues and challenges," *Proc. - Int. Conf. Mach. Learn. Data Eng. iCMLDE 2018*, pp. 22–29, 2019, doi: 10.1109/iCMLDE.2018.00014.

[2] A. Khanna and S. Kaur, "Evolution of Internet of Things (IoT) and its significant impact in the field of Precision Agriculture," *Comput. Electron. Agric.*, vol. 157, no. December 2018, pp. 218–231, 2019, doi: 10.1016/j.compag.2018.12.039.

[3] F. B. Setiawan and Magfirawaty, "Securing Data Communication Through MQTT Protocol with AES-256 Encryption Algorithm CBC Mode on ESP32-Based Smart Homes," *2021 Int. Conf. Comput. Syst. Inf. Technol. Electr. Eng. COSITE 2021*, no. October, pp. 166–170, 2021, doi: 10.1109/COSITE52651.2021.9649577.

[4] M. Ibrahim and B. Sugiarto, "Rancang Bangun Rumah Pintar (Smart Home) Berbasis Internet Of Things (IoT)," *Infotek J. Inform. dan Teknol.*, vol. 6, no. 1, pp. 1–10, 2023, doi: 10.29408/jit.v6i1.5365.

[5] H. Ahmetoglu and R. Das, "A comprehensive review on detection of cyber-attacks: Data sets, methods, challenges, and future research directions," *Internet of Things (Netherlands)*, vol. 20, no. September, p. 100615, 2022, doi: 10.1016/j.iot.2022.100615.

[6] A. J. Hintaw, S. Manickam, S. Karuppayah, M. A. Aladaileh, M. F. Aboalmaaly, and S. U. A. Laghari, "A Robust Security Scheme Based on Enhanced Symmetric Algorithm for MQTT in the Internet of Things," *IEEE Access*, vol. 11, no. May, pp. 43019–43040, 2023, doi: 10.1109/ACCESS.2023.3267718.

[7] M. Picone, S. Cirani, and L. Veltri, "Blockchain security and privacy for the internet of things," *Sensors (Switzerland)*, vol. 21, no. 3, pp. 1–4, 2021, doi: 10.3390/s21030892.

[8] E. C. P. Neto, S. Dadkhah, R. Ferreira, A. Zohourian, R. Lu, and A. A. Ghorbani, "CICIoT2023: A Real-Time Dataset and Benchmark for Large-Scale Attacks in IoT Environment," *Sensors*, vol. 23, no. 13, p. 5941, 2023, doi: 10.3390/s23135941.

[9] N. Gupta, V. Jindal, and P. Bedi, "LIO-IDS: Handling class imbalance using LSTM and improved one-vs-one technique in intrusion detection system," *Comput. Networks*, vol. 192, no. March, pp. 1–19, 2021, doi: 10.1016/j.comnet.2021.108076.

[10] P. I. priyadarsini, *ABC-BSRF: Artificial Bee Colony and Borderline-SMOTE RF Algorithm for Intrusion Detection System on Data Imbalanced Problem*, vol. 56. Springer Singapore, 2021. doi: 10.1007/978-981-15-8767-2_2.

[11] H. Shafique, A. A. Shah, M. A. Qureshi, and M. K. Ehsan, "Machine Learning Empowered Efficient Intrusion Detection Framework," *VFAST Trans. Softw. Eng.*, vol. 10, no. 2, pp. 27–35, 2022, doi: http://dx.doi.org/10.21015/vtse.v10i2.1017.

[12] M. Artur, "Review the performance of the Bernoulli Naïve Bayes Classifier in Intrusion Detection Systems using Recursive Feature Elimination with Cross-validated selection of the best number of features," *Procedia Comput. Sci.*, vol. 190, no. 2019, pp. 564–570, 2021, doi: 10.1016/j.procs.2021.06.066.

[13] B. Charbuty and A. Abdulazeez, "Classification Based on Decision Tree Algorithm for Machine Learning," *J. Appl. Sci. Technol. Trends*, vol. 2, no. 01, pp. 20–28, 2021, doi: 10.38094/jastt20165.

[14] S. Rahman, M. Irfan, M. Raza, K. M. Ghori, S. Yaqoob, and M. Awais, "Performance analysis of boosting classifiers in recognizing activities of daily living," *Int. J. Environ. Res. Public Health*, vol. 17, no. 3, 2020, doi: 10.3390/ijerph17031082.

[15] R. Zuech, J. Hancock, and T. M. Khoshgoftaar, "Detecting web attacks using random undersampling and ensemble learners," *J. Big Data*, vol. 8, no. 1, 2021, doi: 10.1186/s40537-021-00460-8.

[16] A. Divekar, M. Parekh, V. Savla, R. Mishra, and M. Shirole, "Benchmarking datasets for Anomaly-based Network Intrusion Detection: KDD CUP 99 alternatives," *Proc. 2018 IEEE 3rd Int. Conf. Comput. Commun. Secur. ICCCS 2018*, pp. 1–8, 2018, doi: 10.1109/CCCS.2018.8586840.

[17] M. Alshamkhany, W. Alshamkhany, M. Mansour, M. Khan, S. Dhou, and F. Aloul, "Botnet Attack Detection using Machine Learning," *Proc. 2020 14th Int. Conf. Innov. Inf. Technol. IIT 2020*, pp. 203–208, 2020, doi: 10.1109/IIT50501.2020.9299061.

[18] K. Alissa, T. Alyas, K. Zafar, Q. Abbas, N. Tabassum, and S. Sakib, "Botnet Attack

Detection in IoT Using Machine Learning," *Comput. Intell. Neurosci.*, vol. 2022, 2022, doi: 10.1155/2022/4515642.

[19] D. Upadhyay, J. Manero, M. Zaman, and S. Sampalli, "Gradient Boosting Feature Selection With Machine Learning Classifiers for Intrusion Detection on Power Grids," *IEEE Trans. Netw. Serv. Manag.*, vol. 18, no. 1, pp. 1104–1116, 2021.

[20] B. A. Tama and K. H. Rhee, "An in-depth experimental study of anomaly detection using gradient boosted machine," *Neural Comput. Appl.*, vol. 31, no. 4, pp. 955–965, 2019, doi: 10.1007/s00521-017-3128-z.

[21] M. A. S. Arifin, D. Stiawan, and B. Y. Suprapto, "Oversampling and undersampling for intrusion detection system in the supervisory control and data acquisition IEC 60870 - 5 - 104," *IET Cyber-Physical Syst. Theory Appl.*, no. November 2023, 2024, doi: 10.1049/cps2.12085.

[22] Y. Hamid, F. A. Shah, and M. Sugumaran, "Wavelet neural network model for network intrusion detection system," *Int. J. Inf. Technol.*, vol. 11, no. 2, pp. 251–263, 2018, doi: 10.1007/s41870-018-0225-x.

[23] S. Shaikh, S. M. Daudpota, A. S. Imran, and Z. Kastrati, "Towards improved classification accuracy on highly imbalanced text dataset using deep neural language models," *Appl. Sci.*, vol. 11, no. 2, pp. 1–20, 2021, doi: 10.3390/app11020869.