OPTIMASI MODEL XGBOOST MENGGUNAKAN SMOTE DAN PCA UNTUK KLASIFIKASI SERANGAN SIBER PADA *INTERNET OF THINGS*

TESIS

Oleh:
AFRIJAL RIZQI RAMADAN
NIM. 210605220001

PROGRAM STUDI MAGISTER INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2025

OPTIMASI MODEL XGBOOST MENGGUNAKAN SMOTE DAN PCA UNTUK KLASIFIKASI SERANGAN SIBER PADA *INTERNET OF THINGS*

TESIS

Diajukan kepada:
Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang
Untuk Memenuhi Salah Satu Persyaratan Dalam
Memperoleh Gelar Magister Komputer (M.Kom)

Oleh:
AFRIJAL RIZQI RAMADAN
NIM. 210605220001

PROGRAM STUDI MAGISTER INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2025

OPTIMASI MODEL XGBOOST MENGGUNAKAN SMOTE DAN PCA UNTUK KLASIFIKASI SERANGAN SIBER PADA INTERNET OF THINGS

TESIS

Oleh:
AFRIJAL RIZQI RAMADAN
NIM. 210605220001

Telah diperiksa dan disetujui untuk diuji
Tanggal: 12 November 2025

Pembimbing I,	Pembimbing II,
Dr. Ir. Mokhamad Amin Hariyadi, M.T	Dr. Agung Teguh Wibowo Almais, M.T
NIP. 19670018 200501 1 001	NIPPPK. 19860301 2023321 1 016

Mengetahui,
Ketua Program Studi Magister Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang

Prof. Dr. Ir. Muhammad Faisal, S.Kom., M.T.
NIP. 19740510 200501 1 007

OPTIMASI MODEL XGBOOST MENGGUNAKAN SMOTE DAN PCA UNTUK KLASIFIKASI SERANGAN SIBER PADA INTERNET OF THINGS

TESIS

Oleh:
AFRIJAL RIZQI RAMADAN
NIM. 210605220001

Telah Dipertahankan di Depan Dewan Penguji
Dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
untuk Memperoleh Gelar Magister Komputer (M.Kom)
Tanggal: 12 November 2025

Susunan Dewan Penguji			**Tanda Tangan**
Penguji I	:	Dr. Zainal Abidin, M.Kom NIP. 19760613 200501 1 004	( )
Penguji II	:	Dr. Yunifa Miftachul Arif, M.T NIP 19830616 201101 1 004	( )
Pembimbing I	:	Dr. Ir. Mokhamad Amin Hariyadi, M.T NIP. 19670018 200501 1 001	( )
Pembimbing II	:	Dr. Agung Teguh Wibowo Almais, M.T NIPPPK. 19860301 2023321 1 016	( )

Mengetahui dan Mengesahkan,
Ketua Program Studi Magister Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang

Prof. Dr. Ir. Muhammad Faisal, S.Kom., M.T.
NIP. 19740510 200501 1 007

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan dibawah ini:

Nama	: Afrijal Rizqi Ramadan
NIM	: 210605220001
Program Studi	: Magister Informatika
Fakultas	: Sains dan Teknologi
Judul Thesis	: "Optimasi Model XGBoost Menggunakan SMOTE dan PCA untuk Klasifikasi Serangan Siber pada *Internet Of Things*"

Menyatakan dengan sebenarnya bahwa Thesis yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambilalihan data, tulisan atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan Thesis ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 12 November 2025
Yang membuat pernyataan,

Afrijal Rizqi Ramadan
NIM. 210605220001

HALAMAN MOTTO

“Jika hidupmu baik syukuri, namun jika hidupmu buruk ubah cara pandangmu lalu syukuri”

HALAMAN PERSEMBAHAN

ٱلْحَمْدُ لِلَّهِ رَبِّ ٱلْعَالَمِينَ

Puji syukur atas kehadirat Allah SWT
Shalawat serta salam kepada Rasulullah SAW

Dengan segenap hati, penulis mempersembahkan sebuah karya ini kepada:

Istri tercinta Nuril Aulia Naiza Ulfa M.Hum yang selalu menjadi penyemangat, dan selalu memberikan do'a dukungan, serta motivasi.

Seluruh orang tua penulis tercinta, Abah Alm H. Muhamad Tohir S.H dan Umi Siti Cholidah serta Abah H Abd Razak S.Pd.I dan Umi Aisah yang selalu membimbing penulis, memberikan do'a, dukungan, serta motivasi yang tidak terhingga.

Saudara dan saudari tercinta terkhusus Siska Tohirowati S.Pd, Riska Maulidiah S.E, Wahyudianto, Hasan Fauzi S.E, Muhammad Ilzam Mubarak.

Dosen pembimbing Dr. Ir. Mokhamad Amin Hariyadi, M.T dan Dr. Agung Teguh Wibowo Almais, M.T yang telah membimbing penelitian ini dengan memberikan banyak pengarahan dan pengalaman yang berharga.

Segenap sivitas akademika Program Studi Magister Informatika, terutama seluruh dosen, terima kasih atas segenap ilmu dan bimbingannya.

Seluruh rekan-rekan mahasiswa Magister Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang

Penulis ucapkan "*jazakumullah khairan katsiiraa"*. Semoga selalu diridhoi Allah SWT. Aamiin Ya Rabbal 'Alamiin.

KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh

Syukur alhamdulillah penulis hanturkan kehadirat Allah SWT yang telah melimpahkan Rahmat dan Hidayah-Nya, sehingga penulis dapat menyelesaikan studi di Program Studi Magister Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang sekaligus menyelesaikan Thesis ini dengan baik. Shalawat dan salam semoga senantiasa tercurahkan kepada junjungan Nabi Muhammad SAW yang senantiasa menjadi sumber inspirasi dan teladan terbaik begitu juga keluarga, para sahabat dan para pengikutnya seluruh umat Islam.

Penulis haturkan ucapan terima kasih seiring do'a dan harapan jazakumullah ahsanal jaza' kepada semua pihak yang telah membantu terselesaikannya Thesis ini. Ucapan terima kasih ini penulis sampaikan kepada:

1. Bapak Dr. Ir. Mokhamad Amin Hariyadi, M.T dan Bapak Dr. Agung Teguh Wibowo Almais, M.T selaku dosen pembimbing Thesis, yang telah banyak memberikan pengarahan dan pengalaman yang berharga.
2. Bapak Dr. Zainal Abidin, M.Kom dan Bapak Dr. Yunifa Miftachul Arif, M.T dan selaku dosen penguji Thesis, yang telah banyak memberikan pengarahan dan pengalaman yang berharga.
3. Keluarga tercinta yang telah banyak memberikan doa dan dukungan kepada penulis secara moril maupun materil hingga Thesis ini dapat terselesaikan.
4. Segenap Civitas Akademika Program Studi Magister Informatika, terutama seluruh Bapak dan Ibu dosen, terima kasih atas segenap ilmu dan bimbingannya
5. Semua pihak yang ikut membantu dalam menyelesaikan Thesis ini baik berupa materiil maupun moril yang tidak bisa penulis sebutkan satu persatu tanpa mengurasi rasa hormat dan terimakasih.

Penulis menyadari bahwa dalam penyusunan Thesis ini masih terdapat kekurangan dan penulis berharap semoga Thesis ini bisa memberikan manfaat kepada para pembaca khususnya bagi penulis secara pribadi. Aamiin Ya Rabbal 'Alamin.

Wassalamu'alaikum Warahmatullahi Wabarakatuh

Malang, 12 November 2025

Penulis

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

ABSTRAK

Ramadan, Afrijal Rizqi. 2025. **Optimasi Model XGBoost Menggunakan SMOTE dan PCA untuk Klasifikasi Serangan Siber pada *Internet Of Things.*** Tesis. Program Studi Magister Informatika Fakultas Sains Dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Dr. Ir. Mokhamad Amin Hariyadi, M.T.(II) Dr. Agung Teguh Wibowo Almais, M.T

Kata kunci: Serangan Siber, *Internet of Things*, *Smart City*, XGBoost, *Principal Component Analysis*, *Synthetic Minority Oversampling Technique*

Kemajuan teknologi *Internet of Things* (IoT) telah mendorong transformasi signifikan di berbagai bidang. Namun, meningkatnya penggunaan IoT juga memperbesar risiko keamanan siber karena keterbatasan sumber daya dan lemahnya mekanisme pertahanan perangkat. Kondisi ini menimbulkan kebutuhan akan model klasifikasi yang kuat dan efisien untuk mendeteksi berbagai jenis serangan. Penelitian ini berfokus pada optimasi kinerja sistem klasifikasi serangan siber pada *Internet of Things* (IoT) menggunakan pendekatan algoritma *Extreme Gradient Boosting* (XGBoost). Kesenjangan utama yang diidentifikasi adalah adanya ketidakseimbangan data serta redundansi fitur yang muncul pada dataset IoT. Untuk mengatasi hal tersebut, penelitian ini mengembangkan model XGBSP-E5, yaitu kombinasi antara XGBoost, *Synthetic Minority Oversampling Technique* (SMOTE), dan *Principal Component Analysis* (PCA). Dataset yang digunakan terdiri dari 350.000 data serangan IoT yang telah melalui proses pra-pemrosesan dan pembagian data dengan rasio 90:10 untuk pelatihan dan pengujian. Hasil penelitian menunjukkan bahwa model XGBSP-E5 memberikan hasil akurasi 99,68%, presisi 96%, *recall* 95%, *F1-score* 95% menunjukkan kemampuan deteksi yang baik terhadap jenis serangan. Temuan ini menegaskan bahwa penerapan kombinasi SMOTE dan PCA pada model XGBoost dapat digunakan dalam melakukan klasifikasi serangan siber pada IoT.

ABSTRACT

Ramadan, Afrijal Rizqi. 2025**. The Optimization of the XGBoost Model Using the SMOTE and PCA to Classify Cyberattack in Internet of Things**. Thesis. Master of Informatics Study Program, Faculty of Science and Technology, Maulana Malik Ibrahim State Islamic University Malang. Supervisor: (I) Dr. Ir. Mokhamad Amin Hariyadi, M.T. (II) Dr. Agung Teguh Wibowo Almais, M.T.

Keywords: Cyberattack, Internet of Things, Smart City, XGBoost, Principal Component Analysis, Synthetic Minority Oversampling Technique

The technological advance of the Internet of Things (IoT) has driven a significant transformation in numerous fields. However, its increasing use also escalates cybersecurity risks due to the limited resources and weak security mechanisms of the hardware. These conditions urge the need for a strong and efficient classification model to detect diverse types of attacks. The research focuses on optimizing the performance of cyberattack classification systems in the IoT environment using the Extreme Gradient Boosting (XGBoost) algorithm. The primary gap identified lies in the presence of data imbalance and feature redundancy within IoT datasets. To solve these issues, the research develops the XGBSP-E5 model, a combination of XGBoost, the Synthetic Minority Oversampling Technique (SMOTE), and Principal Component Analysis (PCA). The research dataset comprises 350,000 IoT attack records that have undergone preprocessing and were split into training and testing sets at a 90:10 ratio. The research results demonstrate that the XGBSP-E5 model achieves 99.68% accuracy, 96% precision, 95% recall, and a 95% F1-score, indicating strong detection capability across attack types. These results affirm that integrating SMOTE and PCA with the XGBoost model can be used to classify cyberattacks in IoT systems.

مستخلص البحث

رمضان، أفرجال رزقي. 2025. تحسين نموذج إكس جي بوست (*XGBoost*) باستخدام *SMOTE* و *PCA* لتصنيف الهجمات السيبرانية على إنترنت الأشياء. رسالة الماجستير. قسم المعلومات، كلية العلوم والتكنولوجيا بجامعة مولانا مالك إبراهيم الإسلامية الحكومية مولانا مالك إبراهيم مالانج. المشرف الأول: د. محمد أمين هريادي، الماجستير؛ المشرف الثاني: د. أغونج تيغوه وبيووو ألماس، الماجستير.

الكلمات الرئيسية: هجمات سيبرانية، إنترنت أشياء، مدينة ذكية، *XGBoost*، تحليل مكونات رئيسية، تقنية إفراط في أخذ عينات اصطناعية لأقلية.

لقد دفع التقدم في تكنولوجيا إنترنت الأشياء (*IoT*) إلى تحولات كبيرة في مختلف المجالات. ومع ذلك، فإن الاستخدام المتزايد لـ *IoT* يزيد أيضًا من مخاطر الأمن السيبراني بسبب محدودية الموارد وضعف آليات الدفاع للأجهزة. تثير هذه الحالة الحاجة إلى نماذج تصنيف قوية وفعّالة للكشف عن أنواع متنوعة من الهجمات. تركز هذه الرسالة على تحسين أداء نظام تصنيف الهجمات السيبرانية على إنترنت الأشياء (*IoT*) باستخدام خوارزمية التعزيز التدريجي المتطرف (*XGBoost*). ومن الفجوات الرئيسة التي تم تحديدها وجود عدم توازن في البيانات بالإضافة إلى التكرار في الميزات التي تظهر في مجموعة بيانات إنترنت الأشياء (*IoT*) . وللتغلب على ذلك، طورت هذه الرسالة نموذج*XGBSP-E5* ، وهو مزيج بين *XGBoost* وتقنية الإفراط في أخذ العينات الاصطناعية للأقلية (*SMOTE*) وتحليل المكونات الرئيسية (*PCA*). تتألف مجموعة البيانات المستخدمة من 350,000 بيانات لهجمات إنترنت الأشياء تم تمريرها عبر عملية ما قبل المعالجة وتقسيم البيانات بنسبة 90:10 للتدريب والاختبار. أظهرت النتائج أن نموذج *XGBSP-E5* حقق دقة بنسبة 99.68٪، وثبات بنسبة 96٪، والاستدعاء 95٪، ودرجة-ف1 95٪، مما يدل على قدرة جيدة على اكتشاف أنواع الهجمات. تؤكد هذه النتائج أن تطبيق مزيج من *SMOTE* و PCA على نموذج *XGBoost* يمكن استخدامه في تصنيف الهجمات السيبرانية على إنترنت الأشياء.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi *Internet of Things* (IoT) telah membawa banyak perubahan dalam berbagai sektor, seperti *smart home*, *smart city*, dan *Industrial Internet of Things* (IIoT). Jaringan IoT berbeda dengan jaringan tradisional yang didominasi oleh komunikasi *human-to-machine* (H2M) dan lebih banyak menggunakan protokol seperti HTTP, HTTPS, FTP, dan DNS. Sistem IoT beroperasi secara *machine-to-machine* dan umumnya banyak menggunakan protokol ringan seperti MQTT, CoAP atau AMQP yang sebagian besar berbasis UDP (*connectionless*) untuk menghemat bandwidth dan energi (Ansari *et al.*, 2018). Adanya IoT memungkinkan perangkat-perangkat dapat saling berkomunikasi dan bertukar data secara *real-time*, yang memberikan manfaat besar dalam efisiensi dan otomatisasi.

Perangkat IoT umumnya dirancang untuk efisiensi energi dan biaya, sehingga memiliki keterbatasan dalam daya komputasi, kapasitas penyimpanan, serta mekanisme keamanan bawaan (Sibarani *et al*., 2023). Serangan pada IoT memiliki beberapa karakteristik yang berbeda dari jaringan tradisional. Hal ini karena bersifat terdistribusi, otomatis, dan memanfaatkan protokol komunikasi ringan seperti MQTT, CoAP, dan ZigBee. Selain itu, lalu lintas data IoT yang kecil dan dikirim secara periodik sering dimanfaatkan untuk menyembunyikan aktivitas berbahaya. Seperti pada CoAP *Amplification Attack*, penyerang mengeksploitasi

kelemahan protokol CoAP berbasis UDP untuk mengirim permintaan palsu yang membuat perangkat IoT secara tidak sadar membanjiri target dengan lalu lintas data berlebih. Serangan semacam ini tidak terjadi pada jaringan tradisional karena protokol CoAP hanya digunakan di lingkungan IoT (Almeglef *et al.*, 2023).

Pasar IoT tumbuh dengan cepat, dan diperkirakan mencapai nilai sekitar USD 94,2 miliar pada tahun 2026 (Kronlid *et al.*, 2024). Namun, seiring dengan meningkatnya pemanfaatan IoT, muncul pula tantangan dalam hal keamanan siber. Menurut laporan DeepStrike (2025) rata-rata terdapat 820.000 serangan harian yang menargetkan perangkat IoT dengan lonjakan 46 % serangan ransomware pada awal 2025. Badan Siber dan Sandi Negara (BSSN) dalam pernyataan resminya juga menyebut bahwa serangan terhadap infrastruktur IoT dan *supply chain* kini menjadi salah satu fokus utama mitigasi nasional (Sulthon, 2025). Oleh karena itu, isu keamanan menjadi salah satu faktor krusial yang harus diperhatikan dalam pengembangan, implementasi, dan pengelolaan sistem IoT.

Salah satu pendekatan yang banyak dikembangkan untuk mengatasi permasalahan tersebut adalah *Intrusion Detection System* (IDS) berbasis kecerdasan buatan (Kikissagbe & Adda, 2024). IDS berfungsi memantau dan menganalisis aktivitas jaringan guna mendeteksi adanya indikasi serangan atau anomali yang berpotensi membahayakan sistem. Sejumlah penelitian menunjukkan bahwa penerapan *machine learning* dan *deep learning* IDS dapat meningkatkan kemampuan deteksi dengan mengenali pola serangan (Thakkar & Alazab, 2021). Meski demikian, sebagian besar IDS masih berfokus pada implementasi berbasis

cloud, sedangkan sistem IoT memerlukan solusi yang ringan, efisien, dan mampu berjalan pada perangkat *edge* yang memiliki keterbatasan sumber daya komputasi.

Hasil *literature review* oleh Banko *et al.* (2025) menyatakan model berbasis *ensemble learning* seperti *Random Forest* (RF) dan XGBoost banyak digunakan dan memiliki performa yang bagus dalam IDS dengan konsumsi daya yang relatif rendah, sedangkan model *deep learning* seperti *Long Short-Term Memory* (LSTM) dan *Convolutional Neural Network* (CNN) mampu mengenali pola serangan secara lebih kompleks namun membutuhkan daya komputasi yang lebih besar. Sebagaimana di dalam Al-Qur'an, Allah SWT berfirman dalam Surah Yusuf/12:67, yaitu:

وَقَالَ يٰبَنِيَّ لَا تَدْخُلُوْا مِنْۢ بَابٍ وَّاحِدٍ وَّادْخُلُوْا مِنْ اَبْوَابٍ مُّتَفَرِّقَةٍۗ وَمَآ اُغْنِيْ عَنْكُمْ مِّنَ اللّٰهِ مِنْ شَيْءٍۗ اِنِ الْحُكْمُ اِلَّا لِلّٰهِۗ عَلَيْهِ تَوَكَّلْتُ وَعَلَيْهِ فَلْيَتَوَكَّلِ الْمُتَوَكِّلُوْنَ ﴿٦٧﴾

"*Dia (Ya'qub) berkata,* "*Wahai anak-anakku, janganlah kamu masuk dari satu pintu gerbang, dan masuklah dari pintu-pintu gerbang yang berbeda-beda. (Namun,) aku tidak dapat mencegah (takdir) Allah dari kamu sedikit pun. (Penetapan) hukum itu hanyalah hak Allah. Kepada-Nyalah aku bertawakal dan hendaklah kepada-Nya (saja) orang-orang yang bertawakal (meningkatkan) tawakal(-nya)*" (QS. Yusuf /12:67).

Dalam Tafsir Ibnu Katsir, ayat ini menjelaskan bahwa nabi Ya'qub memerintahkan kepada anak-anaknya ketika melepas keberangkatan mereka bersama bunyamin menuju negeri mesir, tidaklah mereka masuk dari satu pintu gerbang, tetapi hendaklah masuk dari berbagai pintu yang berlainan. Menurut Ibnu Abbas, Mohammad Ibnu Ka'ab, Mujahid, dan lainnya hal itu menghindari 'ain atau kesialan (Abdullah bin Muhammad, 2008). Menurut Tafsir Ibnu Katsir, ayat ini mengandung makna kehati-hatian dan strategi dalam menghadapi ancaman. Dalam konteks keamanan siber, nilai tersebut mencerminkan pentingnya membangun

pertahanan yang tidak bergantung pada satu mekanisme tunggal, tetapi menggunakan pendekatan yang saling melengkapi. Dengan demikian, sistem keamanan pada lingkungan IoT diharapkan mampu berfungsi secara lebih tangguh dan presisi dalam mengidentifikasi karakteristik serangan siber.

Hasil *literature review* penelitian mendapatkan bahwa pendekatan berbasis *ensemble learning* seperti XGBoost banyak digunakan dan memberikan akurasi yang tinggi dalam melakukan klasifikasi berbagai jenis serangan dengan skenario *multiclass*. Penelitian oleh Firdaus *et al*. (2025) memperkuat temuan ini pada domain *Internet of Medical Things (IoMT)*, di mana model XGBoost mencapai akurasi 99,8% pada dataset CICIoMT2024, mengungguli *Logistic Regression* dan *Naive Bayes* dalam mendeteksi serangan kompleks. Verma dan Ranga (2020) menemukan bahwa model *ensemble boosting* (termasuk XGBoost dan *Gradient Boosting*) memberikan keseimbangan terbaik antara akurasi dan efisiensi waktu pada berbagai dataset IDS seperti UNSW-NB15 dan NSL-KDD.

Untuk mendukung proses klasifikasi serangan siber pada IoT penelitian ini menggunakan Gotham 2025 dataset, sebuah kumpulan data baru yang memiliki cakupan serangan IoT yang lebih beragam dan realistis dibandingkan dataset terdahulu. Dataset ini dirancang untuk lingkungan IoT berskala besar seperti *smart city*. Dataset Gotham dibentuk dengan meniru 78 perangkat IoT dari berbagai domain, seperti *city power, building monitor*, *air quality sensors, street IP cameras,* dan *predictive maintenance system* yang diemulasikan beroperasi pada berbagai protokol. Data Gotham memiliki distribusi label serangan yang tidak seimbang, di mana serangan minoritas memiliki jumlah data yang lebih sedikit sehingga dapat

terabaikan oleh model deteksi. Dataset yang tidak seimbang menyebabkan bias ke kelas mayoritas dan mengabaikan serangan yang sebenarnya berbahaya sehingga berpengaruh terhadap akurasi (Balla *et al*., 2025). Oleh karena itu, diperlukan metode penyeimbangan data untuk memperbaiki representasi kelas minoritas.

Imani *et al*. (2020) melakukan pengujian *ensemble model* dengan berbagai metode penyeimbangan data menghasilkan temuan dimana kombinasi XGBoost dengan *Synthetic Minority Oversampling Technique* (SMOTE) mampu memberikan akurasi dan deteksi serangan yang lebih baik dibandingkan model lain. Teknik SMOTE berfungsi menyeimbangkan distribusi kelas agar model tidak bias terhadap kelas mayoritas (Doghramachi & Ameen, 2023). Selain itu, untuk mengatasi banyaknya dimensi data pada dataset IoT, akan digunakan metode *Principal Component Analysis* (PCA). PCA dipilih karena mampu mengurangi dimensi data dan menghilangkan fitur-fitur yang redundan, tanpa mengorbankan informasi penting (Gardner & Lo, 2021). Pemanfaatan PCA sebagai bagian optimasi pada model dalam melakukan klasifikasi serangan telah dilakukan oleh beberapa peneliti (Arlandi *et al*., 2025; Winanto *et al*., 2024; Al-Fawa'reh *et al*., 2022; Alqaraleh, 2025). Hal ini tidak hanya mempercepat waktu pelatihan dan inferensi, tetapi juga mengurangi konsumsi memori dan daya komputasi. Allah SWT berfirman dalam Surah Ar-Rahman/55:9, yaitu:

وَاَقِيْمُوا الْوَزْنَ بِالْقِسْطِ وَلَا تُخْسِرُوا الْمِيْزَانَ ۝

"*Dan tegakkanlah timbangan itu dengan adil dan janganlah kamu mengurangi neraca itu*" (QS. Ar-Rahman/55:9).

Dalam tafsir Ibnu Katsir, dijelaskan bahwa ayat ini mengajarkan manusia untuk memperhatikan timbangan yang adil dalam semua amal perbuatan dan

ucapan. Dalam Al-Qur'an Allah tidak saja memberi tahu manusia mengenai ciptaannya, Namun juga memberikan indikasi-indikasi untuk memanfaatkan semua ciptaan untuk kesejahteraan manusia (Abdullah bin Muhammad, 2008). Nilai keseimbangan diimplementasikan secara konsep melalui proses pengolahan data dan pengembangan sistem deteksi serangan berbasis pembelajaran mesin. Penggunaan SMOTE bertujuan untuk menyeimbangkan distribusi data antara kelas mayoritas dan minoritas agar model tidak bias, sedangkan PCA digunakan untuk mereduksi dimensi fitur secara proporsional dengan harapan agar sistem dapat efisien dan akurat. Sementara itu, algoritma XGBoost berperan dalam menjaga keseimbangan antara bias dan variansi model, sehingga mampu mendeteksi serangan secara adil dan objektif.

Dengan mengombinasikan XGBoost dan PCA, penelitian ini diharapkan dapat memberikan kontribusi dalam melakukan klasifikasi serangan siber pada *Internet of Things*.

1.2 Identifikasi masalah

Berdasarkan latar belakang di atas, beberapa permasalahan yang akan dibahas dalam penelitian ini adalah:

1. Bagaimana tingkat kinerja proses klasifikasi serangan siber pada IoT berbasis model XGBoost?
2. Bagaimana pengaruh kinerja metode XGBoost dengan penambahan proses SMOTE dan PCA terhadap proses klasifikasi serangan siber pada IoT?

1.3 Tujuan Penelitian

Tujuan yang ingin dicapai dalam penelitian ini adalah:

1. Mengukur tingkat kinerja proses klasifikasi serangan siber pada IoT berbasis model XGBoost.
2. Menganalisis pengaruh kinerja metode XGBoost dengan penambahan proses SMOTE dan PCA terhadap proses klasifikasi serangan siber pada IoT?

1.4 Manfaat Penelitian

Hasil dari penelitian ini diharapkan dapat memberikan kontribusi dalam pengembangan klasifikasi serangan siber pada IoT**.**

1.5 Batasan Masalah

Adapun batasan masalah dalam penelitian ini adalah:

1. Penelitian ini menggunakan dataset Gotham 2025 dalam konteks IoT *smart city* sebagai sumber data utama untuk pelatihan dan evaluasi model klasifikasi.
2. Atribut *input* yang digunakan frame.time (frt), frame.len (frl), frame.protocols (frp), eth.src (ets), eth.dst (etd), ip.src (ips), ip.dst (ipd), ip.flags (ipf), ip.ttl (ipt), ip.proto (ipp), ip.tos (iptos), ip.checksum (ipc), tcp.srcport (tcs), tcp.dstport (tcd), tcp.flags (tcf), tcp.window_size_value (tcw), tcp.window_size_scalefactor (tcsf), tcp.checksum (tcc), tcp.options (tco), tcp.pdu.size (tcpds), udp.srcport (uds), dan udp.dstport (udd).
3. Atribut *output* yang digunakan yaitu label (lbl) yang menunjukkan jenis aktivitas jaringan.
4. Proses eksperimen dilakukan berdasarkan komposisi data.
5. Model yang digunakan yaitu XGBoost yang dioptimasi dengan menerapkan PCA dan SMOTE.

6. Evaluasi dilakukan menggunakan metrik performa klasifikasi tanpa melibatkan simulasi serangan fisik di perangkat nyata.

BAB II

TINJAUAN PUSTAKA

Penelitian ini melakukan studi literatur untuk mengkaji informasi yang relevan dari berbagai jurnal ilmiah, dengan tujuan memperoleh pemahaman yang komprehensif mengenai konteks, perkembangan, serta kerangka teoritis terkait optimasi model XGBoost untuk klasifikasi serangan siber pada IoT yang telah dibahas dalam penelitian terdahulu.

2.1 Klasifikasi Serangan Cyber pada IoT Berbasis Model XGBoost

Penelitian oleh Firdaus *et al*. (2025) berfokus pada deteksi serangan pada *Internet of Medical Things* (IoMT) yang semakin krusial seiring adopsi perangkat medis berbasis IoT untuk layanan kesehatan. Dengan menggunakan dataset CICIoMT2024, penelitian ini mengevaluasi performa algoritma XGBoost dalam mendeteksi lima jenis serangan, yaitu ARP *Spoofing*, *Recon Attack*, MQTT *Attack*, TCP/IP DoS, dan DDoS. Hasil eksperimen menunjukkan bahwa XGBoost memiliki kinerja unggul dibandingkan algoritma lain seperti *Logistic Regression* dan *Naive Bayes*, dengan capaian akurasi 99,8%, presisi 92,4%, *recall* 96%, dan *F1-score* 93,8%. Tingginya akurasi ini menegaskan keunggulan XGBoost dalam menangani pola serangan kompleks di ekosistem IoMT, yang secara langsung dapat meningkatkan perlindungan terhadap layanan kesehatan kritis.

Penelitian Belarbi *et al*. (2025) memperkenalkan Gotham Dataset 2025, yaitu dataset jaringan *Internet of Things* (IoT) berskala besar yang dirancang untuk penelitian *Intrusion Detection System* (IDS) dan keamanan siber. Dataset ini

merekam lebih dari 3 juta paket jaringan dari 78 perangkat IoT dengan berbagai protokol seperti MQTT, CoAP, dan RTSP, mencakup lalu lintas *Benign* serta beragam serangan nyata seperti DoS, *Brute Force*, *Network Scanning*, *Remote Command Execution*, *Ingress Tool Transfer*, dan C&C *Communication*. Untuk menunjukkan penerapannya, peneliti melatih model *Deep Neural Network* (DNN) menggunakan pendekatan *federated learning*. Hasilnya menunjukkan akurasi sebesar 89% dengan presisi dan *recall* rata-rata 0.90 dan 0.89. Dataset ini nantinya akan digunakan sebagai *benchmark* dalam penelitian.

Penelitian Doghramachi dan Ameen (2023) mengusulkan peningkatan keamanan IoT melalui penerapan *Intrusion Detection System* (IDS) berbasis XGBoost dengan dukungan teknik *preprocessing* yang meliputi *data cleaning*, *encoding*, *normalization*, serta penanganan *class imbalance* menggunakan SMOTE. Dataset yang digunakan adalah IoTID20, yang merepresentasikan kondisi IoT modern dengan distribusi kelas tidak seimbang. Penelitian ini membandingkan beberapa algoritma *machine learning* (*Logistic Regression, Multi-Layer Perceptron*, *Decision Tree*, *Random Forest*, dan XGBoost), dan hasilnya menunjukkan bahwa XGBoost dengan integrasi SMOTE memberikan kinerja terbaik, dengan akurasi mencapai 99% pada klasifikasi *biner*, 99% pada *multi-class*, dan 81% pada *sub-classification*. Kelebihan penelitian ini terletak pada kemampuannya mengatasi masalah ketidakseimbangan data yang umum pada dataset IoT.

Martina Karanfilovska *et al*. (2022) melakukan penelitian yang berfokus pada pengembangan *Network Intrusion Detection System* berbasis *machine*

learning untuk mendeteksi serangan siber pada lingkungan *Internet of Things* (IoT). Studi ini menggunakan dataset NF-ToN-IoT-v2. Penelitian ini menerapkan tiga algoritma utama, yaitu *Decision Tree*, *Random Forest*, dan XGBoost, dengan tahapan data *cleaning* dan *scaling* pada proses prapemrosesan. Dataset ini terdiri atas 9 kelas serangan yang mencakup DDoS, DoS, MITM, *Injection*, *Ransomware*, *Scanning*, *Cross-Site Scripting* (XSS), *Password Attack*, dan *Backdoor*. Hasil eksperimen menunjukkan bahwa algoritma XGBoost *Classifier* memberikan performa terbaik dengan akurasi, presisi, *recall*, dan *F1-score* masing-masing sebesar 98,8%

Berbagai studi menunjukkan bahwa metode *ensemble learning* seperti *Random Forest* (RF), XGBoost, *Gradient Boosting Machine* (GBM), dan *AdaBoost* memiliki performa lebih stabil dibanding model tunggal. Verma & Ranga (2019) melakukan analisis komprehensif terhadap beberapa algoritma ML untuk IDS IoT menggunakan dataset CIDDS-001, UNSW-NB15, dan NSL-KDD. Hasilnya, algoritma *ensemble learning* memiliki akurasi dan generalisasi yang lebih baik 97-99%, serta dapat dijalankan pada perangkat IoT seperti *Raspberry Pi*. Hasil performa model XGBoost dari penelitian ini yaitu dengan akurasi 93.15%, *recall* 99.46%, *specificity* 89.15%, dan AUC 98.76%.

2.2 Metode Optimasi pada Klasifikasi Serangan Cyber pada IoT

Penelitian oleh Javed *et al*. (2022) mengusulkan sistem deteksi intrusi cerdas berbasis *machine learning* untuk mengidentifikasi serangan *Advanced Persistent Threat* (APT) pada jaringan *Industrial Internet of Things* (IIoT). Dalam penelitian ini, berbagai algoritma diuji, termasuk *Decision Tree*, *Random Forest*, *Support*

Vector Machine, *Logistic Regression*, *Naive Bayes*, *Bagging*, *AdaBoost*, dan XGBoost. Peneliti juga menerapkan *Pearson Correlation Coefficient* (PCC) untuk proses seleksi dan ekstraksi fitur sebelum tahap klasifikasi. Untuk performa XGBoost terbukti mampu mencapai akurasi 98%. Dengan menggunakan dataset KDDCup99, model XGBoost berhasil mengklasifikasikan lalu lintas normal dan anomali secara akurat, mengatasi masalah *overfitting* dan ketidakseimbangan data.

Penelitian oleh Saed Alqaraleh (2025) mengusulkan sistem deteksi anomali jaringan berbasis *ensemble learning* yang mengombinasikan beberapa algoritma *machine learning* klasik dengan teknik *Principal Component Analysis* (PCA) untuk meningkatkan efisiensi komputasi tanpa menurunkan akurasi deteksi. Studi ini menguji model *K-Nearest Neighbor* (KNN)**,** *Naive Bayes* (NB)**,** *Random Forest* (RF)**,** *AdaBoost***,** dan *Gradient Boosting* (GB)**.** Dataset yang digunakan adalah KDD CUP99, dengan penerapan PCA yang mempertahankan 95% variansi fitur sambil mengurangi dimensi dari 41 menjadi 19, serta teknik *dual-phase* SMOTE untuk menyeimbangkan kelas minoritas. Hasil pengujian menunjukkan bahwa model *Gradien Boosting* mencapai akurasi 90.5%, presisi 92%, *recall* 89%, dan *F1-score* 96%

Winanto *et al*. (2024) melakukan penelitian yang berfokus pada peningkatan akurasi sistem deteksi intrusi (IDS) pada jaringan *Internet of Things* (IoT). Studi ini menggabungkan dua pendekatan utama, yaitu *Principal Component Analysis* (PCA) untuk reduksi dimensi dan *Random Forest* (RF) sebagai model klasifikasi utama. PCA digunakan untuk menghilangkan korelasi antar fitur serta mengekstraksi fitur-fitur penting agar sistem mampu membedakan lalu lintas

normal dan serangan dengan lebih efisien. Sementara itu, algoritma *Random Forest* dimanfaatkan karena kemampuannya mengatasi masalah *overfitting* dan memberikan prediksi yang stabil. Dataset yang digunakan adalah CIC-IoT 2023, yang mencakup tujuh tipe serangan. Pengujian menunjukkan bahwa kombinasi PCA dan *Random Forest* berhasil meningkatkan akurasi deteksi hingga akurasi 97.05%, *recall* 57, dan presisi 45% dengan performa terbaik diperoleh pada konfigurasi 10 fitur utama.

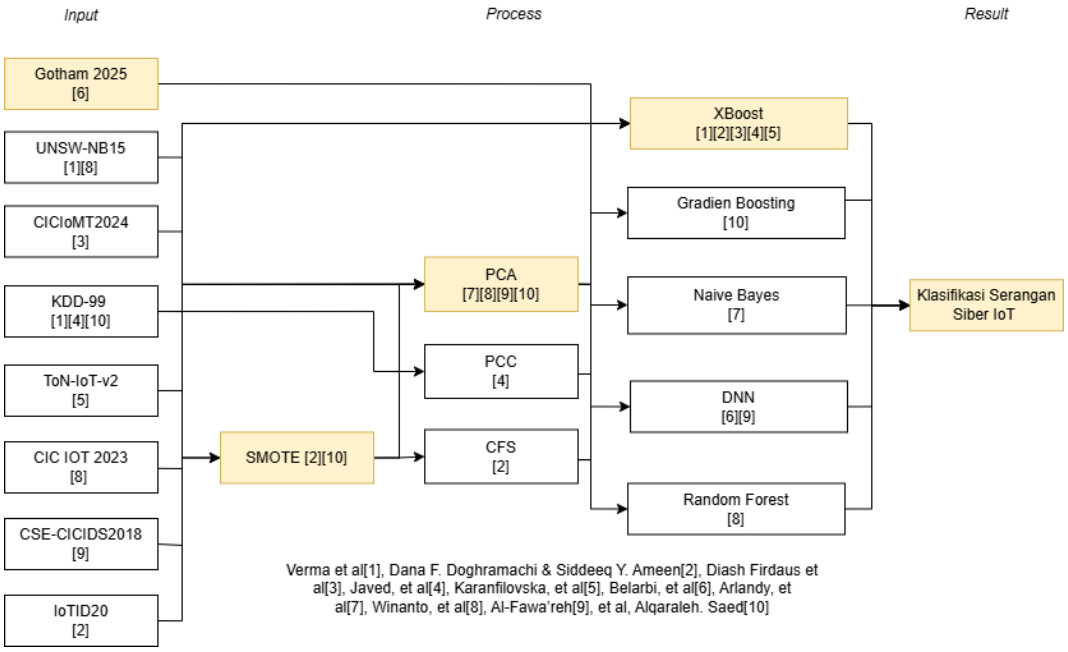
Penelitian yang dilakukan oleh Al-Fawa'reh *et al*. (2022) mengusulkan pendekatan hibrida antara *Principal Component Analysis* (PCA) dan *Deep Neural Network* (DNN) untuk meningkatkan performa sistem deteksi intrusi pada lingkungan jaringan besar. Penelitian ini berangkat dari permasalahan umum pada IDS konvensional, yaitu tingginya *false positive rate*, lamanya waktu deteksi, serta ketidakmampuan mendeteksi serangan *zero-day*. Dengan menggunakan dataset CSE-CICIDS2018, model ini melakukan ekstraksi fitur pada level *flow* dan menerapkan PCA untuk reduksi dimensi agar komputasi lebih efisien, kemudian melatih DNN untuk klasifikasi anomali. Hasil eksperimen menunjukkan bahwa kombinasi PCA-DNN mampu mencapai akurasi sebesar 98% dengan waktu pelatihan yang lebih singkat dibandingkan DNN tanpa PCA.

Penelitian yang dilakukan oleh Arlandy *et al.* (2025) berjudul mengusulkan kombinasi *Naive Bayes Classifier* dengan *Principal Component Analysis* (PCA) untuk meningkatkan efisiensi dan akurasi sistem deteksi intrusi. Studi ini menggunakan dataset UNSW-NB15 yang merepresentasikan berbagai jenis serangan. PCA diterapkan untuk mereduksi dimensi data menjadi 30 komponen

utama tanpa kehilangan informasi penting, sehingga mengurangi kompleksitas model dan mempercepat proses komputasi. Hasil eksperimen menunjukkan bahwa model dengan PCA mencapai akurasi 97.05%, *recall* 81%, presisi 89%, dan *F1-score* 85%**.** Meskipun demikian, presisi pada kelas minoritas masih rendah (0.49), menunjukkan perlunya teknik tambahan seperti *feature balancing* atau *hybrid learning*.

2.3 Kerangka Teori

Bagian kerangka teori membahas teori-teori yang mendukung penelitian terkait klasifikasi serangan siber berbasis model XGBoost pada *Internet of Things* (IoT). Kerangka teori pada Gambar 2.1 memuat berbagai pendekatan yang telah banyak digunakan dan metode terbaik dari penelitian terdahulu dipaparkan untuk memberikan landasan konseptual, sekaligus menjadi acuan dalam pemilihan teknik yang relevan untuk penelitian ini.



Gambar 2.1 Kerangka Teori

Studi-studi pada Tabel 2.1 menegaskan bahwa pendekatan berbasis *ensemble learning* seperti XGBoost banyak digunakan dan memberikan akurasi yang tinggi dalam melakukan klasifikasi berbagai jenis serangan dengan skenario *multiclass*. Firdaus *et al*. (2025) memperkuat temuan ini pada domain *Internet of Medical Things (IoMT)*, di mana model XGBoost mencapai akurasi 99,8% pada dataset CICIoMT2024, mengungguli *Logistic Regression* dan *Naive Bayes* dalam mendeteksi serangan kompleks. Verma dan Ranga (2020) menemukan bahwa model *ensemble boosting* memberikan keseimbangan terbaik antara akurasi dan efisiensi waktu pada berbagai dataset IDS seperti UNSW-NB15 dan NSL-KDD. Sementara Javed *et al*. (2022) membuktikan bahwa algoritma *Adaboost* dan XGBoost efektif untuk mendeteksi *Advanced Persistent Threats (APT)* di lingkungan *Industrial* IoT dengan akurasi 98%, menunjukkan bahwa *boosting-based models* lebih efisien daripada model konvensional berbasis *deep learning* dalam konteks perangkat terbatas sumber daya.

Permasalahan yang dihadapi pada dataset IDS IoT adalah ketidakseimbangan distribusi kelas dan tingginya dimensi fitur yang berpotensi menyebabkan *overfitting* serta menurunkan akurasi klasifikasi (Firdaus *et al*., 2025). Untuk itu, berbagai penelitian mengusulkan penerapan teknik *Synthetic Minority Oversampling Technique* (SMOTE) untuk menambah representasi kelas minoritas sehingga model mampu mengenali serangan langka secara lebih baik (Doghramachi & Ameen, 2023; Alqaraleh, 2025). Di sisi lain, *Principal Component Analysis* (PCA) terbukti efektif dalam melakukan reduksi dimensi tanpa menghilangkan informasi penting, sehingga proses pembelajaran menjadi

lebih cepat dan efisien (Arlandi *et al*., 2025; Winanto *et al*., 2024; Al-Fawa'reh *et al*., 2022; Alqaraleh, 2025).

Tabel 2.1 Literatur Penelitian Sebelumnya

Peneliti	**Dataset**	**Algoritma**	***Preprocessing***	**Optimasi Fitur**	**Performa**
Verma *et al*. (2019)	UNSW-NB15, NSL-KDD, CIDDS-001	**XGBoost,** RF, GBM, MLP, CART, AB, ETC	*Cleaning*, Normalisasi	–	Akurasi=93.15% Recall=99.46% Specificity= 89.15% AUC=98.76%
Dana F. Doghramachi, Siddeeq Y. Ameen (2023)	IoTID20	*Logistic Regression*, MLP, *Decision Tree*, *Random Forest*, **XGBoost**	*Cleaning*, *Encoding*, Normalisasi, **SMOTE**	*Correlation Feature Reduction* (CFR)	Akurasi=81%
Diash Firdaus *et al*. (2025)	CICIoMT2024 (2024)	**XGBoost**, *Logistic Regression*, *Naive Bayes*	*Encoding*, Normalisasi	–	Akurasi=99.8% Presisi=92.4% Recall=96% f1-Score=93.8%
Javed, *et al* (2022)	KDD-99 (1999)	NB, DT, RF, SVM, LR, *Bagging*, **XGBoost** and *AdaBoost*	*Cleaning*, Normalisasi	*Pearson Correlation Coefficient* (PCC)	Akurasi=98% Recall=97% Presisi=96% f1 Score=96%
Karanfilovska, *et al* (2022)	ToN-IoT-v2	*Decision Tree*, *Random Forest*, **XGBoost**	*Cleaning*, *Scalling*		Akurasi=98.8% Recall=98.8% Presisi=98.8% f1 Score=98.8%
Belarbi, *et al* (2025)	**Gotham 2025**	DNN	*Cleaning*,		Akurasi=89% Recall=89% Presisi=90% f1 Score=87%
Arlandy, *et al* (2025)	UNSW-NB15	*Naïve Bayes*	*Clearning*, Normalisasi, *Scalling*	***Principal Component Analysis* (PCA)**	Akurasi=97.05% Recall=81% Presisi=89% F1 Score=85%
Winanto, *et al* (2024)	CIC IOT 2023	*Random Forest*		***Principal Component Analysis* (PCA)**	Akurasi=97.05% Recall=57% Presisi=45%
Al-Fawa'reh, *et al* (2022)	CSE-CICIDS2018	DNN	*Clearning*, Normalisasi, *Scalling*	***Principal Component Analysis* (PCA)**	Akurasi=98%
Alqaraleh. Saed (2025)	KDD-99 (1999)	NB, *AdaBoost*, RF, KNN *Gradien Boosting*	*Cleaning*, Normalisasi, *One Hot Encoding*, **SMOTE**	***Principal Component Analysis* (PCA)**	Akurasi=90.5% Recall=89% Presisi=92% F1 Score=96.%

Hasil penelitian pada Tabel 2.1 menjadi landasan bahwa kombinasi antara tahap pra-proses reduksi dan penyeimbangan data dengan model klasifikasi berbasis *boosting* . Selain itu, pendekatan *ensemble* berbasis XGBoost dinilai lebih ringan dibandingkan model *deep learning* seperti CNN atau LSTM, sehingga lebih sesuai diterapkan pada perangkat IoT dengan keterbatasan sumber daya (Adewole *et al*., 2025). Dengan mengacu pada temuan tersebut, kerangka teori penelitian ini menempatkan XGBoost sebagai algoritma utama klasifikasi, dengan dukungan SMOTE untuk mengatasi ketidakseimbangan data dan PCA untuk reduksi dimensi fitur. Dengan konsep tersebut diharapkan dapat menghasilkan model yang efisien, akurat, dan adaptif dalam mengklasifikasikan berbagai jenis serangan siber pada lingkungan *Internet of Things*.

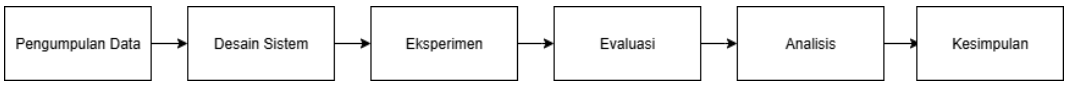
BAB III

DESAIN PENELITIAN

Desain penelitian merupakan rancangan sistematis yang berfungsi sebagai arah dalam pelaksanaan seluruh tahapan penelitian, mulai dari identifikasi masalah hingga proses analisis data. Desain penelitian berperan penting dalam memastikan bahwa metode, teknik, serta prosedur yang digunakan dapat menghasilkan temuan yang valid dan ilmiah. (Khanday & Khanam, 2019). Evaluasi model dilakukan menggunakan *confusion matrix* sebagaimana dijelaskan oleh Sathyanarayanan, S. (2024) untuk menilai tingkat akurasi, presisi, *recall*, dan *F1-score*. Dengan demikian, desain penelitian ini diharapkan mampu memberikan gambaran utuh mengenai proses dan hasil analisis klasifikasi serangan siber pada *Internet of Things* (IoT).

3.1 Prosedur Penelitian

Prosedur penelitian merupakan rangkaian langkah yang dilakukan peneliti untuk mengumpulkan data serta menjawab rumusan masalah yang telah ditetapkan. Pada bagian ini dijelaskan secara rinci metode yang digunakan dalam pengumpulan data, proses pengolahan, serta tahapan pelaksanaan eksperimen. Setiap skenario eksperimen yang dilakukan akan mendapatkan kinerja dari model, sehingga dapat mengetahui model mana yang menghasilkan kinerja terbaik. Langkah terakhir dari prosedur penelitian yaitu melakukan evaluasi dan analisis hasil dari experimen seperti yang ditunjukkan pada Gambar 3.1.



Gambar 3.1 Prosedur Penelitian

3.1.1 Pengumpulan Data

Data yang digunakan dalam penelitian ini adalah Gotham Dataset, yaitu data sekunder yang diperoleh dari laman Kaggle (*https://www.kaggle.com/datasets/sachins8201/gotham*). Dataset ini dirancang khusus dengan menyediakan lingkungan yang realistis dan heterogen untuk mendukung penelitian di bidang *Intrusion Detection System* (IDS) dan keamanan siber pada *Internet of Things* (IoT). Testbed ini mencakup 78 perangkat IoT yang diemulasikan beroperasi pada berbagai protokol, termasuk MQTT, CoAP, dan RTSP dari berbagai domain seperti *city power, building monitor, air quality sensors, street IP cameras,* dan *predictive maintenance system*. Dataset Gotham terdiri dari data lalu lintas jaringan yang merepresentasikan aktivitas normal dan serangan siber yang terjadi pada lingkungan IoT. Proses pengumpulan data yang dilakukan dalam penelitian ini dilakukan dengan megunduh dataset Gotham 2025.

frt	frl	frp	ets	etd	ips	ipd	ipf	ipt	ipp	ipc	iptos	tcs	tcd	tcf	tcw	tcsf	tcc	tco	tcpds	uds	udd	lbl
Jan 14, 2025 18:40:22.447710000 GMT	67	eth:ethertype:ip:udp:dns	02:42:52:d7:fa:00	0c:6e:9c:16:00:00	192.168.0.2	192.168.18.17	0x02	64	17	0xb19d		48322	53									Benign
Jan 14, 2025 18:40:22.453402000 GMT	83	eth:ethertype:ip:udp:dns	0c:6e:9c:16:00:00	02:42:52:d7:fa:00	192.168.18.17	192.168.0.2	0x02	61	17	0x49d9		53	48322									Benign
Jan 14, 2025 18:40:22.453507000 GMT	90	eth:ethertype:ip:udp:ntp	02:42:52:d7:fa:00	0c:6e:9c:16:00:00	192.168.0.3	192.168.18.17	0x02	64	17	0xa71f		46343	123									Benign
Jan 14, 2025 18:40:22.458119000 GMT	90	eth:ethertype:ip:udp:ntp	0c:6e:9c:16:00:00	02:42:52:d7:fa:00	192.168.18.17	192.168.0.3	0x02	61	17	0xab96		123	46343									Asign
Jan 14, 2025 18:40:22.560013000 GMT	76	eth:ethertype:ip:udp:dns	02:42:52:d7:fa:00	0c:6e:9c:16:00:00	192.168.0.2	192.168.18.17	0x02	64	17	0x2e8d		36848	53									Asign

Gambar 3. 2 Contoh Dataset Gotham 2025

Pengumpulan data dilakukan pada tanggal 23 September 2025 dengan jumlah data sekitar 35.134.283 data. Gambar 3.2 merupakan contoh dataset Gotham

2025 yang terdiri dari atribut antara lain frame.time (frt), frame.len (frl), frame.protocols (frp), eth.src (ets), eth.dst (etd), ip.src (ips), ip.dst (ipd), ip.flags (ipf), ip.ttl (ipt), ip.proto (ipp), ip.tos (iptos), ip.checksum (ipc), tcp.srcport (tcs), tcp.dstport (tcd), tcp.flags (tcf), tcp.window_size_value (tcw), tcp.window_size_scalefactor (tcsf), tcp.checksum (tcc), tcp.options (tco), tcp.pdu.size (tcpds), udp.srcport (uds), udp.dstport (udd), dan label (lbl). Label pada dataset ini terdiri dari delapan (8) kelas yaitu *Denial of Service* (DoS), *Benign* (BEN), *Network Scanning* (NS), CoAP *Amplification Attack Data* (CAAD), *Telnet Brute Forcing* (TBF), *Ingress Tool Transfer* (ITT), dan *Periodic Command and Control* (C&C) *Communication* (PC3C).

3.1.2 Desain Sistem

Desain sistem dibuat untuk memberikan gambaran rancangan secara umum rancangan sistem yang dikembangkan untuk melakukan klasifikasi serangan siber pada lingkungan *Internet of Things* (IoT) menggunakan pendekatan berbasis XGBoost. Desain sistem disusun agar alur pemrosesan data, penerapan metode optimasi, serta implementasi model dapat terlihat jelas dan terukur. Proses dimulai dari pemanfaatan dataset Gotham 2025 sebagai sumber data utama, kemudian dilanjutkan dengan tahapan pra pemrosesan untuk menyiapkan fitur agar sesuai dengan kebutuhan model. Setelah itu, data diproses melalui tiga varian model, yaitu XGB (XGBoost *non-balancing*), XGBS (XGBoost dengan pra-pemrosesan SMOTE), dan XGBSP (XGBoost dengan kombinasi pra-pemrosesan SMOTE dan PCA). Setiap varian digunakan untuk menguji sejauh mana teknik optimasi

mempengaruhi performa model dalam mendeteksi berbagai jenis serangan siber pada IoT.

3.1.2.1 Data Gotham 2025

Tahapan awal data Gotham 2025 akan diproses di dalam sistem. Tahap ini, data input diolah menjadi data mentah dalam format *Comma Separated Values* (CSV), yang terdiri dari atribut berikut.

Tabel 3.1 Detail Atribut Gotham 2025

No	Kode	Nama Atribut	Tipe Data	Deskripsi
1	frt	frame.time	*Datetime*	Waktu tangkap paket jaringan
2	frl	frame.len	Numerik	Panjang total *frame* dalam *byte*
3	frp	frame.protocols	Kategorik	Protokol yang digunakan pada *frame*
4	ets	eth.src	Kategorik	Alamat sumber Ethernet (MAC)
5	etd	eth.dst	Kategorik	Alamat tujuan Ethernet (MAC)
6	ips	ip.src	Kategorik	Alamat sumber IP
7	ipd	ip.dst	Kategorik	Alamat tujuan IP
8	ipf	ip.flags	Kategorik	Status *flag* dari *header* IP
9	ipt	ip.ttl	Numerik	*Time To Live* (TTL) paket IP
10	ipp	ip.proto	Kategorik	Jenis protokol IP (TCP/UDP/ICMP)
11	iptos	ip.tos	Numerik	Tipe layanan pada IP
12	ipc	ip.checksum	Numerik	Nilai *checksum* validasi paket IP
13	tcs	tcp.srcport	Numerik	Port sumber dari koneksi TCP
14	tcd	tcp.dstport	Numerik	Port tujuan dari koneksi TCP
15	tcf	tcp.flags	Kategorik	Status *flag* pada *header* TCP
16	tcw	tcp.window_size_value	Numerik	Ukuran jendela (*window size*) TCP
17	tcsf	tcp.window_size_scalefactor	Numerik	Faktor skala ukuran jendela TCP
18	tcc	tcp.checksum	Numerik	Nilai checksum untuk validasi TCP
19	tco	tcp.options	Kategorik	Opsi tambahan dalam *header* TCP
20	tcpds	tcp.pdu.size	Numerik	Ukuran data *payload* TCP
21	uds	udp.srcport	Numerik	Port sumber dari koneksi UDP
22	udd	udp.dstport	Numerik	Port tujuan dari koneksi UDP
23	lbl	Label	Kategorik	Kategori serangan / normal sebagai variabel target

Tabel 3.1 merupakan keterangan data hasil tangkapan dari aktivitas jaringan *Internet of Things* (IoT) yang berisi berbagai informasi penting mengenai lalu lintas data antar perangkat. Tujuan utama dataset ini adalah untuk mengklasifikasikan apakah suatu aktivitas jaringan bersifat normal atau mengandung serangan. Setiap baris data mewakili satu paket jaringan yang dikirim atau diterima pada waktu

tertentu. Atribut frt (frame.time) menunjukkan waktu persis ketika paket ditangkap oleh sistem. Hal ini penting dalam analisis temporal, misalnya untuk melihat pola serangan dalam kurun waktu tertentu atau mendeteksi anomali berbasis waktu.

Atribut frl (frame.len) memberikan informasi panjang total paket dalam satuan *byte*. Nilai ini sering digunakan untuk mendeteksi *Packet Flooding Attack* atau *Denial of Service* (DoS) Attack, di mana ukuran paket dan frekuensi kirimnya menjadi tidak wajar. Atribut frp (frame.protocols), ets (eth.src), dan etd (eth.dst) berhubungan dengan lapisan Ethernet. Atribut ini menunjukkan protokol yang digunakan serta alamat MAC sumber dan tujuan. Perubahan pola atau kemunculan alamat MAC yang tidak dikenal dapat mengindikasikan aktivitas mencurigakan pada jaringan IoT. Sementara itu, atribut ips (ip.src) dan ipd (ip.dst) merepresentasikan alamat IP sumber dan tujuan dari paket. Atribut ini sangat penting untuk mengidentifikasi arah komunikasi data, serta mendeteksi serangan seperti IP *spoofing* atau akses tidak sah antar node IoT.

Kemudian, atribut ipf (ip.flags) dan ipt (ip.ttl) digunakan untuk mendeskripsikan karakteristik dari protokol IP. Nilai TTL (*Time To Live*) yang tidak wajar sering menjadi indikator serangan rekayasa jaringan, karena dapat menunjukkan adanya manipulasi rute atau perangkat perantara. Selain itu, ipp (ip.proto) menunjukkan jenis protokol yang digunakan pada lapisan transport (misalnya TCP, UDP, atau ICMP). Kombinasi protokol tertentu sering dikaitkan dengan pola serangan spesifik, seperti UDP *flooding* atau SYN *attack*. Atribut iptos

dan ipc menambahkan konteks lebih lanjut mengenai konfigurasi layanan dan validasi paket.

Pada bagian transport layer, atribut seperti tcs (tcp.srcport), tcd (tcp.dstport), tcw (tcp.window_size_value), dan tcpds (tcp.pdu.size) memberikan gambaran tentang perilaku komunikasi TCP. Nilai-nilai port yang sering muncul (misalnya 53, 80, 443) mengindikasikan layanan umum seperti DNS atau HTTP, sedangkan port tidak biasa bisa menandakan komunikasi malware atau command and control (C2). Sedangkan untuk protokol UDP, atribut uds (udp.srcport) dan udd (udp.dstport) menunjukkan pola komunikasi tanpa koneksi, yang sering digunakan dalam serangan DDoS. Atribut terakhir, lbl (label), Label *Benign* merepresentasikan lalu lintas jaringan yang normal atau aktivitas sah tanpa indikasi serangan. Sementara itu, label lainnya mengacu pada berbagai jenis aktivitas berbahaya seperti *Denial of Service* (DoS) yang membanjiri layanan hingga tak dapat merespon, *Telnet Brute-Force* yang mencoba menebak kata sandi secara berulang, dan *Network Scanning* yang bertujuan memetakan target dalam jaringan. Selain itu terdapat *Remote Command/Code Execution* yang memungkinkan penyerang menjalankan perintah atau kode di perangkat korban, *Ingress Tool Transfer* yaitu proses pengunggahan alat atau *malware* ke sistem, serta *Periodic Command and Control Communication* (C&C) di mana perangkat terinfeksi secara rutin terhubung dengan server penyerang. Aktivitas *Reporting* menunjukkan upaya penyerang mengekstraksi, sedangkan CoAP *Amplification Attack* mengeksploitasi protokol IoT untuk memperbesar dampak serangan. Atribut inilah yang akan

digunakan sebagai *ground truth* dalam pelatihan model klasifikasi menggunakan algoritma XGBoost dan variannya (XGB, XGBS, XGBSP).

3.1.2.2 *Preprocessing* Data

Tahapan *preprocessing* data dilakukan pembersihan dan dinormalisasi untuk memastikan bahwa data yang digunakan memiliki kualitas yang baik dan konsisten. Hal ini dilakukan dengan menggunakan metode skala *min-max* pada kumpulan data yang telah disiapkan untuk mengurangi kesalahan saat pengujian model prediktif. Rumus 3.1 digunakan untuk menghitung skala *min-max*.

$$x_{norm} \frac{x - \min(x)}{\max(x) - \min(x)} \quad 3.1$$

Dimana:

x = data yang dinormalisasi

$xnorm$ = data yang ternormalisasi

$\min(x)$ = nilai terkecil dari keseluruhan data

$\max(x)$ = nilai terbesar dari keseluruhan data.

Contoh hasil dari normalisasi data pada proses ini dapat dilihat pada Tabel 3.2.

Tabel 3.2 Contoh Hasil Normalisasi Data

frt_norm	**frl_norm**	**ipt_norm**	**ipp_norm**	**tcs_norm**	**tcd_norm**	**lbl**
0	0	1	1	1	0	*Benign*
0.14	0.42	0	1	0	1	*Benign*
0.28	1	1	1	0.96	0.23	*Benign*
0.42	1	0	1	0.23	0.96	*Benign*
1	0.48	1	1	0.61	0	*Benign*

Setelah data dinormalisasi, proses selanjutnya adalah pemisahan data yang bertujuan untuk memisahkan data pelatihan dari data pengujian. Tahapan

pendistribusian data membagi data ke dalam dua kategori, yaitu data pelatihan dan data pengujian. Data yang didistribusikan pada penelitian ini terdiri dari beberapa komposisi untuk memperoleh kinerja model terbaik. Pendistribusian data dilakukan berdasarkan hasil eksperimen, dengan mempertimbangkan pembagian data yang mungkin menghasilkan nilai kinerja atau performa terbaik selama pengujian algoritma dilakukan.

3.1.2.3 SMOTE

Ketidakseimbangan data terjadi ketika jumlah sampel dalam satu kelas jauh lebih banyak dibandingkan kelas lainnya. Kondisi ini menyebabkan model pembelajaran mesin cenderung berpihak pada kelas dengan jumlah data lebih besar (kelas mayoritas) dan mengabaikan kelas dengan jumlah data lebih kecil (kelas minoritas). Akibatnya, performa model dalam mengenali pola dari kelas minoritas yang sering kali justru merepresentasikan aktivitas serangan menjadi rendah (Azmatul *et al*., 2013). Dalam konteks *Intrusion Detection System* (IDS) pada jaringan *Internet of Things* (IoT), hal ini dapat menyebabkan model gagal mendeteksi tipe serangan tertentu yang jarang terjadi namun berdampak kritis terhadap keamanan sistem.

Untuk mengatasi hal tersebut, penelitian ini menggunakan metode *Synthetic Minority Oversampling Technique* (SMOTE) sebagaimana dikemukakan oleh Chawla *et al*. (2002). SMOTE merupakan teknik *oversampling* yang menghasilkan data sintetis baru untuk kelas minoritas, bukan sekadar menyalin data yang sudah ada. Tujuannya adalah untuk menyeimbangkan distribusi antar kelas sehingga

algoritma klasifikasi dalam hal ini XGBoost dapat belajar secara proporsional dari kedua kelas, baik normal maupun serangan.

Secara umum, SMOTE bekerja dengan mencari sejumlah *k-nearest neighbors* (tetangga terdekat) dari setiap sampel kelas minoritas, lalu membuat data baru di antara titik-titik tersebut. Data sintetis tersebut dibentuk dengan cara mengambil sebagian atribut dari data asli dan sebagian dari tetangga terdekatnya secara acak. Besarnya jumlah data sintetis yang dihasilkan ditentukan oleh *oversampling factor* yang diinginkan (Azmatul dkk., 2013). Adapun langkah-langkah utama proses pembangkitan data buatan dengan SMOTE adalah sebagai berikut:

1. Data Numerik:
    - Hitung selisih vektor fitur antara data minoritas dengan k-tetangga terdekatnya menggunakan jarak *Euclidean*.
    - Kalikan selisih tersebut dengan bilangan acak antara 0 dan 1.
    - Tambahkan hasilnya ke data asli untuk membentuk sampel sintetis baru.
2. Data Kategorik:
    - Pilih nilai nominal berdasarkan mayoritas dari tetangga terdekat.
    - Jika terjadi nilai yang sama, pilih secara acak untuk menjaga variasi data sintetis.

Untuk data berskala kategorik, perhitungan jarak antar contoh data dilakukan menggunakan *Value Difference Metric* (VDM), yang

mempertimbangkan perbedaan nilai antar atribut kategorik. Rumus VDM didefinisikan pada 3.2 berikut (Azmatul dkk., 2013).

$$\Delta(X,Y) = i = 1 \sum_{i=1}^{N} \delta(V1,V2)$$

$$\delta(V1,V2) = \sum_{i=1}^{n} \left| \frac{C_{1i}}{C_1} - \frac{C_{2i}}{C_2} \right| \quad 3.2$$

Keterangan:

- $\Delta(X,Y)$ = jarak antara dua amatan X dan Y
- $\boldsymbol{N}$ = jumlah variabel prediktor
- C_{1i}, C_{2i} = jumlah kategori ke-i dalam dua kelas berbeda
- n =: banyaknya kategori pada variabel ke-i

Dalam penelitian ini, teknik SMOTE diterapkan pada dataset Gotham 2025, yang digunakan untuk klasifikasi serangan siber berbasis IoT. Dataset ini memiliki distribusi kelas yang tidak seimbang, di mana trafik normal mendominasi dibandingkan jenis-jenis serangan seperti *Ingress Tool Transfer* dan *C&C Communication*. Dengan menerapkan SMOTE, peneliti menyeimbangkan jumlah sampel antar kelas sebelum proses pelatihan model XGBoost dilakukan. Langkah ini penting untuk memastikan bahwa model tidak bias terhadap trafik normal, sehingga kemampuan klasifikasi terhadap tipe serangan langka meningkat secara signifikan. Kombinasi SMOTE dengan XGBoost diharapkan dapat menghasilkan sistem IDS yang lebih adil, efisien, dan akurat dalam mengenali pola serangan pada lingkungan IoT yang dinamis dan beragam.

3.1.2.4 Principal Component Analysis

Principal Component Analysis (PCA) adalah sebuah teknik analisis data yang digunakan untuk melakukan pengurangan dimensi pada sebuah dataset dengan tetap mempertahankan informasi pada dataset. PCA bekerja dengan mencari vektor utama (*principal components*) dari dataset yang memiliki variansi terbesar, yang kemudian digunakan untuk mewakili data. PCA umumnya digunakan ketika variabel dalam data memiliki kemiripan atau korelasi yang tinggi antar kolom. Kemiripan ini sering disebutnya data yang berulang atau *redundant*. Berikut adalah langkah-langkah dalam melakukan PCA:

1. Melakukan fitur ekstraksi: Dataset Gotham memiliki kombinasi atribut bertipe numerik dan kategorik. Pada bagian ini ekstraksi fitur dilakukan dengan memilih atribut yang bertipe numerik.
2. Normalisasi data: Seluruh fitur numerik dinormalisasi menggunakan metode *standard scaler* agar setiap atribut memiliki skala yang sebanding.
3. Menghitung matrik kovariansi: Matrik kovariansi digunakan untuk menghitung variansi dan korelasi antar fitur dalam dataset dengan rumus 3.3 berikut. (Dinanti & Purwadi, 2023)

$$C_{i,j} = \frac{1}{(n\ -\ 1)} \sum_{k=1}^{n} (C_{ki}\ -\ \bar{x}_{ki})(\bar{x}_{kj}\ -\ \bar{x}_{j}) \qquad 3.3$$

4. Menghitung *eigenvectors* dan *eigenvalues*: *Eigenvectors* adalah vektor utama yang memiliki variansi terbesar dan *eigenvalues* adalah variansi yang terkait dengan masing-masing *eigenvector*. *Eigenvectors* dan *eigenvalues* dari matrik kovariansi C dapat ditemukan dengan memecahkan persamaan *eigenvalue* pada 3.4 berikut:

$$Cv = \lambda v \qquad 3.4$$

di mana λ adalah *eigenvalue* dan v adalah *eigenvector*

5. Pemilihan *eigenvectors*: *Eigenvectors* dengan variansi terbesar dipilih sebagai *principal components*.
6. Proyeksi data: Data dapat dikompresi dengan memproyeksikan data ke dalam subruang spasial yang didefinisikan oleh *principal components*. Hasil proyeksi inilah yang digunakan sebagai input pada model XGBoost.

3.1.2.5 Algoritma XGBoost

Metode XGBoost merupakan pohon regresi yang memiliki aturan keputusan yang sama dengan decision tree. XGBoost dapat digunakan untuk regresi dan klasifikasi. Algoritma ini termasuk dalam varian yang efisien dan terukur dari *Gradient Boosting Machine* (GBM) yang telah banyak diterapkan dalam computer vision, data mining dan bidang lainnya. XGBoost telah ditingkatkan dalam dua aspek yaitu untuk mempercepat konstruksi pohon dan mengusulkan algoritma terdistribusi baru untuk pencarian pohon (Chen & Carlos., 2016).

Algoritma XGBoost pada pohon variabel berfungsi untuk memperkirakan variabel target. Pohon pertama berfungsi untuk memprediksi nilai target, pohon kedua berfungsi untuk memprediksi selisih antara estimasi tujuan dan pohon pertama, dan seterusnya hingga residunya diminimalkan (Chen & Carlos., 2016). Nilai prediksi pada algoritma XGBoost ditentukan oleh pohon K seperti dalam persamaan 3.5 berikut:

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i), \quad f_k \in \mathcal{Y} \qquad 3.5$$

Keterangan:

$\hat{y}_i$ = prediksi model untuk daya ke-i

K = jumlah total pohon keputusan yang dibangun oleh model

$f_k(x_i)$ = Fungsi prediksi dari pohon ke-k untuk data input x_i. Setiap pohon f_k merupakan *regression tree.*

$\mathcal{Y}$ = Ruang fungsi kemungkinan dari semua pohon keputusan.

Untuk meminimalkan sisa fungsi kerugian (L) pada tahap pelatihan algoritma XGBoost menggunakan persamaan 3.6 berikut antara nilai sebenarnya (y) dan nilai prediksi (ŷ):

$$L = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{i=1}^{n} \Omega(f_k) \quad 3.6$$

Keterangan:

L = total loss (fungsi kerugian total)

$l(y_i, \hat{y}_i)$ = fungsi kerugian untuk sampel ke-i

y_i = nilai aktual

$\hat{y}_i$ = nilai prediksi

n = jumlah sampel data

$\Omega(f_k)$ = regularisasi untuk model ke-k

k = jumlah model dasar

3.1.2.6 Ekstraksi Fitur

Tujuan ekstraksi fitur adalah untuk mengurangi jumlah fitur dalam suatu dataset. Hal ini dilakukan untuk menentukan atribut-atribut yang menjadi dasar pengujian data. Prosedur ekstraksi fitur dijabarkan pada Tabel 3.3.

Tabel 3.3 Ekstraksi fitur

Nama Atribut	Tipe Data	Keterangan
frame.time	Datetime	Dikurangi
frame.len	Numerik	-
frame.protocols	Kategorik	Dikurangi

Nama Atribut	Tipe Data	Keterangan
eth.src	Kategorik	Dikurangi
eth.dst	Kategorik	Dikurangi
ip.src	Kategorik	Dikurangi
ip.dst	Kategorik	Dikurangi
ip.flags	Kategorik	Dikurangi
ip.ttl	Numerik	-
ip.proto	Kategorik	Dikurangi
ip.tos	Numerik	Data Kosong
ip.checksum	Numerik	Data Kosong
tcp.srcport	Numerik	-
tcp.dstport	Numerik	-
tcp.flags	Kategorik	Dikurangi
tcp.window_size_value	Numerik	-
tcp.window_size_scalefactor	Numerik	-
tcp.checksum	Numerik	-
tcp.options	Kategorik	Dikurangi
tcp.pdu.size	Numerik	-
udp.srcport	Numerik	-
udp.dstport	Numerik	-
Label	Kategorik	Variabel target

Tabel 3.3 menunjukkan bahwa terdapat 23 atribut pada penelitian ini, 10 atribut yaitu frame.time, frame.protocols, eth.src, eth.dst, ip.src ip.dst, ip.flags, ip.proto, tcp.flags, tcp.options dikurangi karena tidak digunakan. Sedangkan atribut label dijadikan sebagai variabel target dalam proses klasifikasi serangan siber pada IoT.

3.1.3 Eksperimen

Tahap uji coba dalam penelitian ini dilakukan menggunakan bahasa pemrograman Python untuk mengklasifikasikan berbagai jenis serangan siber pada jaringan IoT dengan menerapkan algoritma XGBoost**.** Penelitian ini menggunakan tiga varian model, yaitu XGBoost *non-balancing* (XGB)**,** XGBoost pra-pemrosesan SMOTE (XGBS)**,** dan XGBoost pra-pemrosesan SMOTE dan PCA (XGBSP)**.** Setiap model diuji untuk menilai kemampuan klasifikasinya dengan menggunakan

confusion matrix sebagai parameter utama evaluasi, yang mencakup nilai akurasi, presisi, *recall*, dan *F1-score.* Proses *tuning hyperparameter* dilakukan melalui pendekatan *grid search* untuk memperoleh performa klasifikasi paling optimal. Dengan demikian, hasil akhir penelitian ini didasarkan pada model dengan nilai akurasi tertinggi dan skor AUC terbesar.

Tabel 3.4 Komposisi Data Latih dan Data Uji

Nama Model	**Kode Model**	**Komposisi Data**	**Kode Pelatihan**
XGBoost *non-balancing*	XGB	50:50	XGB-E1
		60:40	XGB-E2
		70:30	XGB-E3
		80:20	XGB-E4
		90:10	XGB-E5
XGBoost pra-pemrosesan SMOTE	XGBS	50:50	XGBS-E1
		60:40	XGBS-E2
		70:30	XGBS-E3
		80:20	XGBS-E4
		90:10	XGBS-E5
XGBoost pra-pemrosesan SMOTE dan PCA	XGBSP	50:50	XGBSP-E1
		60:40	XGBSP-E2
		70:30	XGBSP-E3
		80:20	XGBSP-E4
		90:10	XGBSP-E5

Tabel 3.4 menunjukkan komposisi yang dibagi menjadi lima kelompok dengan komposisi data latih dan data uji yang model E1 sebesar 50% : 50%, model E2 sebanyak 60% : 40%, model E3 sebesar 70% : 30%, model E4 sebanyak 80% : 20%, dan model E5 sebesar 90% : 10%.

3.1.4 Evaluasi

Evaluasi kinerja dalam penelitian ini menggunakan *confusion matrix* yang digunakan untuk mengukur nilai akurasi, presisi, *recall*, *F1-score,* TPR dan FPR. *Confusion matrix* merupakan metode yang sering digunakan pada *machine learning* untuk mengevaluasi atau mengukur kinerja sistem yang dibangun. Berikut

merupakan konsep dari *confusion matrix* yang ditunjukkan pada Tabel 3.5.

Tabel 3.5 Konsep *Confusion Matrix*

Confusion Matrix		Kelas Aktual	
		Positif	Negatif
Kelas Prediksi	Positif	TP	FP
	Negatif	FN	TN

Metode *confusion matrix* ini digunakan untuk perbandingan antara data hasil klasifikasi dengan data aktual. Pengujian dilanjutkan dengan menghitung nilai akurasi, presisi, *recall*, *F1-score,* TPR dan FPR. Beberapa metrik yang digunakan adalah sebagai berikut:

a. Akurasi

Akurasi digunakan untuk mengatahui seberapa akurat model dapat mengklasifikasikan data target dengan benar. Persamaan yang digunakan yaitu pada 3.7 sebagai berikut.

$$Akurasi = \frac{Jumlah\ Klasifikasi\ Benar}{Jumlah\ Dokumen\ Uji} = \frac{TP + TN}{TP + TN + FP + FN}\ X\ 100\% \qquad 3.7$$

b. Presisi

Presisi menggambarkan sejauh mana keakuratan data yang diminta dengan hasil prediksi model. Adanya presisi untuk mengukur proporsi deteksi serangan yang benar (*True Positive*) dibandingkan seluruh prediksi serangan (*True Positive* + *False Positive*). Persamaan yang digunakan yaitu pada 3.8 sebagai berikut.

$$Presisi = \frac{TP}{TP + FP}\ X\ 100\% \qquad 3.8$$

c. *Recall*

Recall menunjukkan tingkat keberhasilan sistem dalam mengidentifikasi semua kasus dan menemukan kembali suatu informasi Dalam hal ini yaitu

menunjukkan kemampuan model dalam mendeteksi serangan yang sebenarnya ada. Persamaan yang digunakan yaitu pada 3.9 sebagai berikut.

$$Recall = \frac{TP}{TP + FN} \; X \; 100\% \qquad 3.9$$

d. F1-Score

F1-score merupakan metrik gabungan dari presisi dan *recall*. Metrik ini ditujukan untuk menggambarkan kondisi dataset *imbalance* karena menjaga keseimbangan antara keduanya. Persamaan yang digunakan yaitu pada 3.10 sebagai berikut.

$$F1\ Score = \frac{2\ X\ Precision + Recall}{Precision \times Recall} \qquad 3.10$$

Keterangan:

- *True Positive* (TP): Kasus serangan yang sebenarnya merupakan serangan, dan berhasil diklasifikasikan sebagai serangan oleh sistem. Contoh: paket jaringan berisi aktivitas DoS *attack* dan diprediksi dengan benar sebagai DoS *attack.*
- *False Positive* (FP): Kasus yang sebenarnya bukan serangan, namun diklasifikasikan sebagai serangan oleh sistem. Contoh: lalu lintas IoT normal salah diklasifikasikan sebagai serangan brute force.
- *False Negative* (FN): Kasus yang sebenarnya serangan, tetapi tidak diklasifikasikan oleh sistem dan dianggap sebagai aktivitas normal. Contoh: data jaringan berisi *Telnet Brute Force Attack*, namun diprediksi sebagai normal *traffic*.
- *True Negative* (TN): Kasus yang sebenarnya bukan serangan, dan diklasifikasikan benar sebagai normal oleh sistem. Contoh: lalu lintas

jaringan IoT normal diprediksi dengan benar sebagai normal trafik.

3.2 Instrumen Penelitian

Instrumen penelitian menggambarkan variabel apa saja yang akan digunakan dalam proses penelitian. Tabel 3.6 menyajikan daftar parameter yang digunakan dalam penelitian ini, yang mencakup variabel *independen* (bebas), variabel *intervening* (penghubung), serta variabel *dependen* (terikat) yang saling berperan dalam menjelaskan hubungan antar unsur penelitian.

Tabel 3.6 Variabel penelitian

Independen Variable	***Main Process***	***Intervening Variable***	***Dependen Variabel***
Atribut Gotham 2025	XGBoost	Klasifikasi Serangan *Cyber* pada IoT	Kinerja Model (Akurasi, Presisi, *Recall*, *F1-Score*)

Variabel independen yang digunakan dalam penelitian ini yaitu atribut pada Gotham Dataset (2025) yang berisi data lalu lintas jaringan *Internet of Things* (IoT). Dataset ini mencakup berbagai fitur numerik dan kategorikal seperti protokol komunikasi, port sumber dan tujuan, ukuran paket, serta durasi koneksi yang merepresentasikan aktivitas jaringan baik normal maupun berpotensi serangan. Selanjutnya, *main process* dalam penelitian ini adalah algoritma XGBoost, yang berfungsi untuk mempelajari pola dari atribut jaringan dan membangun model klasifikasi jenis serangan pada jaringan IoT. Model ini nantinya juga akan dilakukan eksperimen dengan penerapan teknik *balancing* data (SMOTE) untuk menangani ketidakseimbangan kelas dan reduksi dimensi (PCA) guna meningkatkan efisiensi pemrosesan data serta mengurangi redundansi fitur. Variabel *intervening* dalam penelitian ini adalah proses klasifikasi serangan siber

pada IoT, yang mencerminkan tahap transformasi data hasil *preprocessing* menjadi keluaran prediksi kategori serangan. Terakhir, variabel *dependen* dalam penelitian ini adalah kinerja model klasifikasi yang dievaluasi menggunakan beberapa metrik, yaitu akurasi, presisi, *recall*, dan *F1-score*. Nilai-nilai ini digunakan untuk mengukur tingkat keberhasilan model dalam mendeteksi dan mengklasifikasikan serangan siber pada jaringan IoT secara akurat.

BAB IV

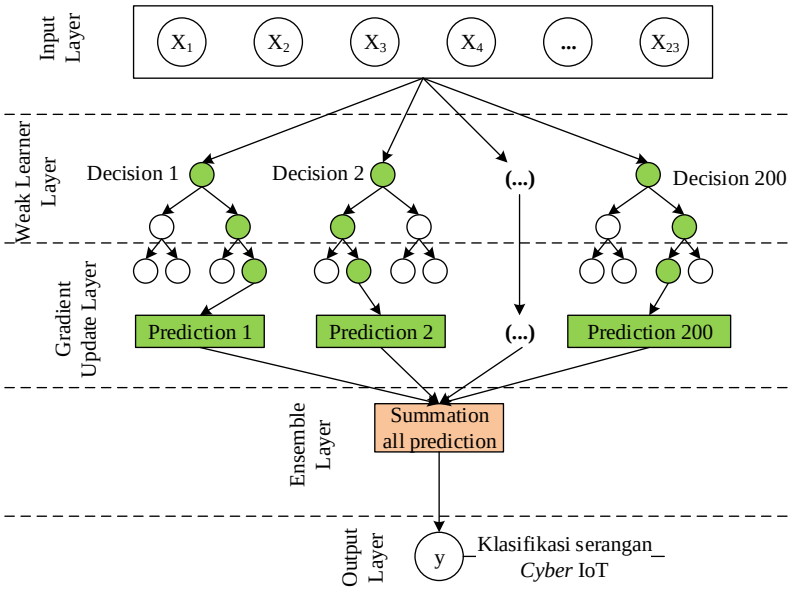
MODEL XGBOOST *NON-BALANCING*

Pemodelan sistem menggunakan algoritma XGBoost *non-balancing* diterapkan dalam penelitian ini untuk klasifikasi serangan siber. Pendekatan *non-balancing* digunakan untuk melihat parameter yang optimal serta melihat hasil dari karakteristik alami distribusi data serangan, sehingga model dapat belajar secara lebih realistis terhadap kondisi sebenarnya di lingkungan jaringan. Penjelasan lebih rinci mengenai evaluasi performa model, parameter yang dioptimasi, serta interpretasi hasil klasifikasi akan dijabarkan pada sub-bab berikut.

4. 1 Desain Model XGBoost *Non-Balancing*

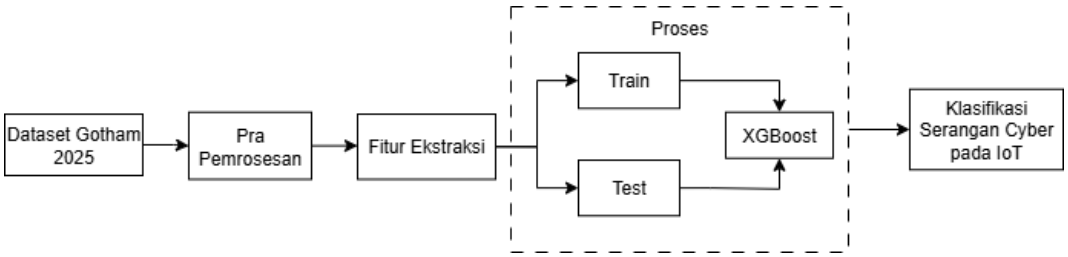
Tahapan penelitian ini menggunakan model XGBoost *non-balancing* dengan skema pelatihan bertahap mulai dari XGB-E1 hingga XGB-E5 untuk melakukan klasifikasi serangan siber. Model XGBoost yang dikembangkan memanfaatkan 200 estimasi pohon keputusan (n_estimators = 200), dengan harapan mampu menghasilkan prediksi yang lebih konsisten, stabil, dan efisien dalam mendeteksi pola serangan yang kompleks. Gambaran lebih rinci mengenai rancangan arsitektur model XGBoost *non-balancing* ini dapat dilihat pada Gambar 4.1. Proses *training* model XGBoost *non-balancing* dimulai dengan memanggil data *training* yang berisi berbagai fitur serangan siber yang telah melalui tahap pra-pemrosesan. Pemilihan nilai *hyperparameter* pada algoritma XGBoost dalam penelitian ini dilakukan menggunakan pendekatan *grid search* untuk menemukan kombinasi terbaik yang mampu menghasilkan kinerja model paling optimal.

Hyperparameter yang diuji meliputi learning rate**,** max depth**,** n_estimators. Variasi nilai pada setiap hiperparameter dipilih berdasarkan literatur yang sudah dikaji.



Gambar 4.1 Arsitektur Model XGBoost *Non-Balancing*

Pada setiap iterasi pelatihan (XGB-E1 hingga XGB-E5), model menjalani proses optimasi bertahap untuk menemukan kombinasi parameter terbaik tanpa melakukan balancing terhadap distribusi kelas. Hasil prediksi dari seluruh pohon keputusan kemudian digabungkan menggunakan mekanisme *weighted boosting*, di mana bobot setiap pohon ditentukan berdasarkan tingkat kesalahannya. Proses *training* dilanjutkan hingga seluruh 200 pohon keputusan terbentuk atau model mencapai titik konvergensi dalam hal kestabilan performa. Setelah proses pelatihan selesai, model XGBoost *non-balancing* yang telah terlatih akan menyimpan struktur pohon dan bobot *boosting* untuk digunakan pada tahap pengujian. Desain sistem dari eksperimen ini ditunjukkan pada Gambar 4.2.



Gambar 4.2 Desain Sistem Model XGBoost Non-Balancing

4. 2 Implementasi Model XGBoost *Non-Balancing*

Dalam eksperimen ini, algoritma XGBoost digunakan untuk membangun model klasifikasi yang mampu mengenali berbagai jenis serangan siber pada sistem IoT. Pengolahan data dan implementasi model dilakukan menggunakan Google Colab dengan bahasa pemrograman Python. Dataset yang digunakan berisi beberapa kategori serangan siber, meliputi *Denial of Servic*e (DoS), *Benign* (BEN), *Network Scanning* (NS), CoAP *Amplification Attack Data* (CAAD), *Telnet Brute Forcing* (TBF), *Ingress Tool Transfer* (ITT), dan *Periodic Command and Control* (C&C) *Communication* (PC3C).

Model XGBoost diinisialisasi dengan percobaan beberapa parameter yang disesuaikan untuk mencapai kinerja optimal tanpa melakukan *balancing* terhadap distribusi kelas. Pengaturan ini bertujuan agar model mampu menemukan parameter terbaik menjaga stabilitas performa dan menghindari *overfitting*. Hasil akhir prediksi diperoleh melalui kombinasi bobot dari seluruh pohon keputusan, sehingga menghasilkan klasifikasi yang lebih akurat dan adaptif terhadap variasi pola serangan. Dengan konfigurasi ini, diharapkan model XGBoost *non-balancing* mampu memberikan hasil klasifikasi yang optimal terhadap berbagai jenis serangan siber pada lingkungan *Internet of Things* (IoT). Rincian lengkap mengenai

parameter dan konfigurasi model XGBoost yang digunakan dalam pelatihan disajikan pada Tabel 4.1.

Tabel 4.1 Ujicoba Parameter Model XGBoost *Non-Balancing*

Parameter	Nilai	Keterangan	Penjelasan
n_estimators	100, **200**, 300, 500	Jumlah Pohon	Menentukan jumlah pohon keputusan yang akan dibangun dalam model.
learning_rate	0.01, 0,05, **0.1,** 0,2	Laju Pembelajaran	Mengontrol seberapa besar kontribusi setiap pohon baru terhadap model akhir.
max_depth	4,**6**,8,10	Kedalaman Maksimum Pohon	Menentukan kedalaman maksimum setiap pohon keputusan.
subsample	0.8	Proporsi Subset Data	Mengatur persentase data pelatihan yang digunakan untuk membangun setiap pohon.
colsample_bytree	0.8	Proporsi Fitur per Pohon	Menentukan proporsi fitur (kolom) yang akan digunakan dalam pembangunan setiap pohon.
random_state	42	Pengaturan Angka Acak	Menetapkan *seed* acak untuk memastikan hasil eksperimen dapat direproduksi secara konsisten setiap kali model dijalankan.
use_label_encoder	FALSE	Penggunaan Label *Encoder* Internal	Menonaktifkan encoder label bawaan XGBoost agar tidak terjadi konflik dengan proses encoding yang telah dilakukan sebelumnya dalam tahap *preprocessing*.
eval_metric	'mlogloss'	Metrik Evaluasi	Menggunakan *multiclass* log loss sebagai metrik evaluasi utama untuk mengukur seberapa baik model memprediksi probabilitas kelas yang benar pada data multi-kelas.

Selain itu, Dalam eksperimen ini, proses pelatihan model XGBoost *non-balancing* dilakukan dengan membagi data awal menjadi dua bagian utama, yaitu data pelatihan dan data pengujian. Pembagian ini bertujuan untuk memastikan bahwa model dapat belajar secara optimal dari sebagian data, kemudian diuji keakuratannya menggunakan data lain yang belum pernah dilihat sebelumnya. Langkah ini penting agar performa model mencerminkan kemampuan generalisasi terhadap data baru yang mewakili kondisi serangan siber di dunia nyata.

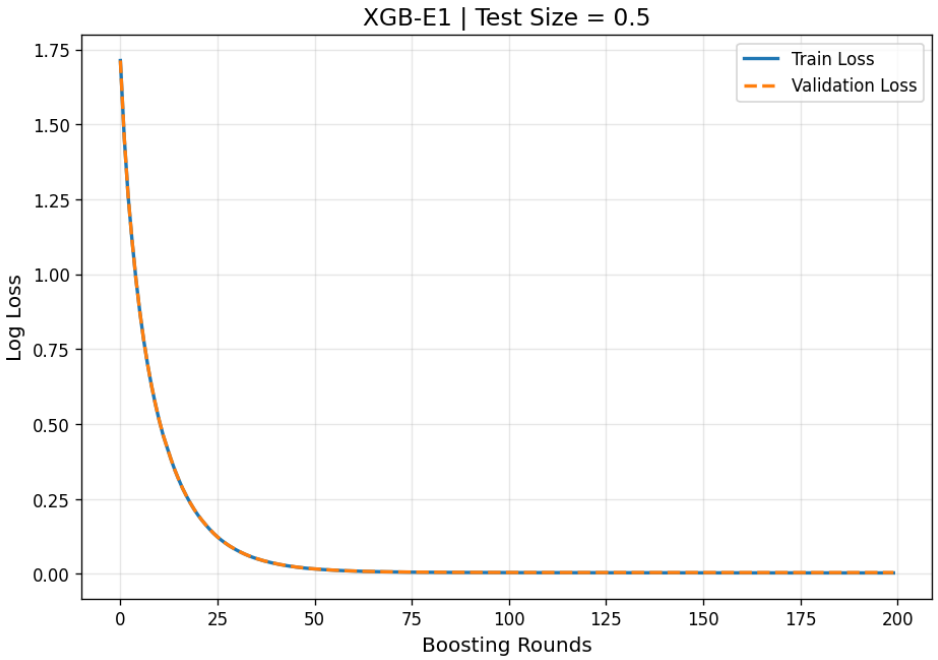
Tabel 4.2 Distribusi Data Pelatihan Model XGBoost *Non-Balancing*

Nama Pelatihan	**Komposisi Data**	**Data Latih**	**Data Uji**
XGB-E1	50:50	175.000	175.000
XGB-E2	60:40	210.000	140.000
XGB-E3	70:30	245.000	105.000
XGB-E4	80:20	280.000	70.000
XGB-E5	90:10	315.000	35.000

Untuk mengeksplorasi bagaimana proporsi pembagian data memengaruhi hasil pelatihan sebagaimana ditunjukkan pada Tabel 4.2 digunakan beberapa variasi skema pelatihan yang dibedakan berdasarkan kode model. Setiap variasi tersebut diberi kode XGB-E1, XGB-E2, XGB-E3, XGB-E4, dan XGB-E5, yang masing-masing merepresentasikan perbedaan dalam rasio pembagian data pelatihan dan pengujian. Melalui serangkaian pelatihan dari XGB-E1 hingga XGB-E5, model XGBoost dioptimalkan untuk menemukan kombinasi parameter dan pembagian data terbaik guna menghasilkan performa klasifikasi yang stabil, akurat, dan adaptif terhadap berbagai jenis serangan siber pada lingkungan *Internet of Things* (IoT).

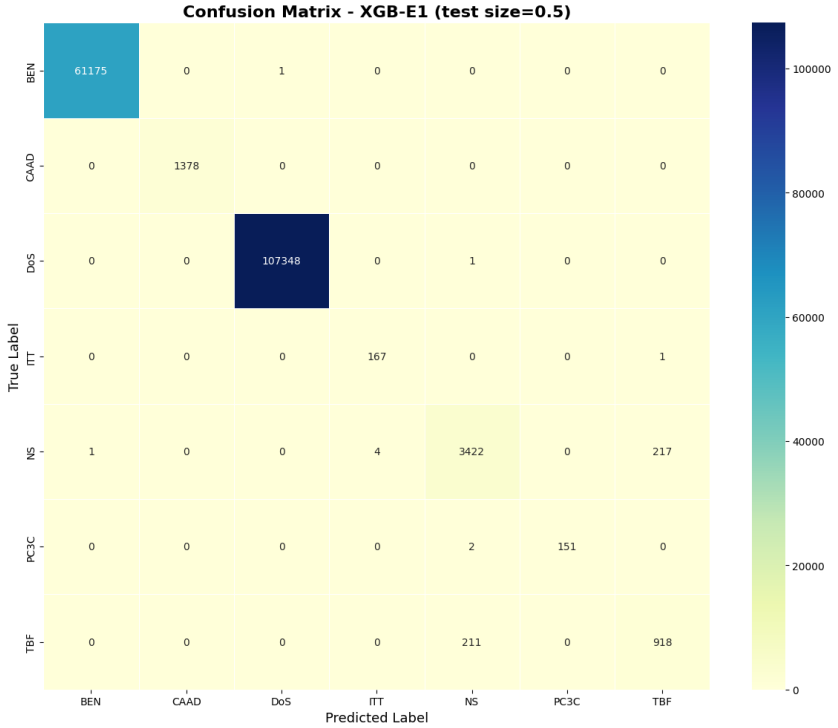
4.2.1 Pelatihan Model XGB-E1

Proses eksperimen dari pelatihan model XGB-E dalam melakukan klasifikasi serangan siber pada IoT digunakan proporsi data latih sebesar 50% atau sekitar 175.000 data dan data uji sebesar 50% atau sekitar 175.000 data. Model dilatih dengan jumlah n_estimators yaitu sebanyak 200 pohon. Hasil pelatihan menunjukkan nilai *train loss* sekitar 0.003311 dan nilai *validation loss* sekitar 0.004526 sebagaimana ditampilkan pada Gambar 4.3.



Gambar 4.3 Grafik Nilai *Train* dan *Validation Loss* Model XGB-E1

Gambar 4.3 menunjukkan nilai *train* dan *validation loss* pada proses pelatihan model XGB-E1 yang terus mengalami penurunan hingga mencapai titik konvergen pada n_estimators ke-200, sehingga model tersebut dianggap telah optimal. Selanjutnya, model optimal yang telah didapatkan, kemudian digunakan untuk melakukan klasifikasi serangan siber berdasarkan kategori serangan siber. Hasil dari klasifikasi tersebut disajikan pada Gambar 4.4.



Gambar 4.4 Hasil Klasifikasi Model XGB-E1 Terhadap Serangan Siber

Hasil klasifikasi model XGB-E1 pada Gambar 4.4 menunjukkan performa klasifikasi yang sangat baik pada sebagian besar kelas, khususnya BEN, CAAD, DoS, dan ITT, yang seluruhnya diprediksi hampir sempurna. Kelas BEN memiliki 61.175 data benar dengan hanya 1 kesalahan, sementara DoS mencapai 107.348 prediksi tepat dari total data uji. Namun, kelemahan model terlihat pada kelas NS dan TBF, di mana terjadi kekeliruan antara keduanya. Sekitar 218 data NS salah diprediksi sebagai TBF. Kesalahan ini menandakan bahwa pola lalu lintas antara NS dan TBF memiliki kemiripan karakteristik, sehingga model mengalami ambiguitas.

Secara keseluruhan, model XGB-E1 memiliki akurasi yang tinggi untuk kelas mayoritas tetapi masih menghadapi tantangan dalam membedakan serangan

minor dengan fitur yang tumpang tindih. Selanjutnya, untuk menghitung nilai metrik akurasi, presisi, *recall*, dan *F1-score* maka dibutuhkan nilai TP, TN, FP, dan FN dari hasil klasifikasi serangan siber berdasarkan kategori serangan siber secara rinci disajikan dalam Tabel 4.3.

Tabel 4.3 Nilai TP, TN, FP, dan FN Setiap Kategori pada Model XGB-E1

Kelas	***True Positives* (TP)**	***True Negatives* (TN)**	***False Positives* (FP)**	***False Negatives* (FN)**
BEN	61175	113822	2	1
CAAD	1378	173622	0	0
DoS	107348	67646	5	1
ITT	167	174832	0	1
NS	3422	171142	214	222
PC3C	151	174847	0	2
TBF	918	173653	218	211

Setelah didapatkan nilai TP, TN, FP, dan FN dari setiap kategori serangan siber, langkah selanjutnya adalah menghitung metrik evaluasi yang digunakan untuk mengetahui performa model XGB-E1 dalam klasifikasi serangan siber pada IoT berdasarkan kategorinya yang secara rinci disajikan pada Tabel 4.4.

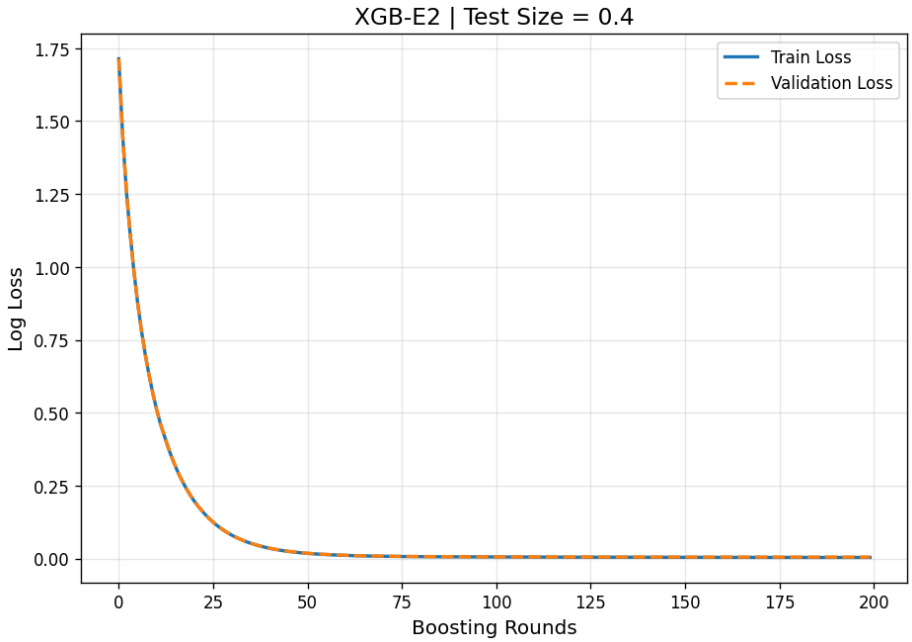
Tabel 4.4 Performa Hasil Pengujian Model XGB-E1

Kelas	**Akurasi (%)**	**Presisi**	***Recall***	***F1-Score***	***Support***
BEN	99.74	1	1	1	61176
CAAD		1	1	1	1378
DoS		1	1	1	107349
ITT		1	0.99	1	168
NS		0.94	0.94	0.94	3644
PC3C		1	0.99	0.99	153
TBF		0.81	0.81	0.81	1129

4.2.2 Pelatihan Model XGB-E2

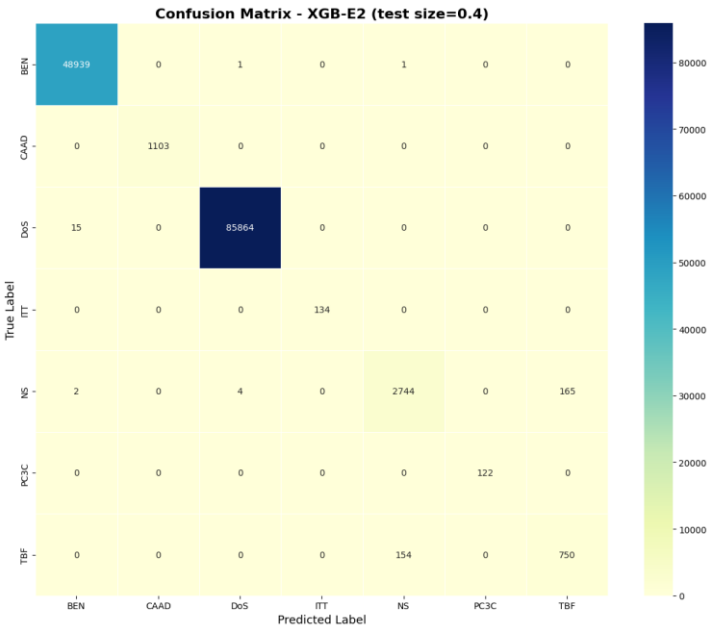
Pada proses eksperimen dari pelatihan model XGB-E2 dalam melakukan klasifikasi serangan siber pada IoT digunakan proporsi data latih sebesar 60% atau

sekitar 210.000 data dan data uji sebesar 40% atau sekitar 140.000 data. Model dilatih dengan jumlah n_estimators yang telah ditentukan, yaitu sebanyak 200 pohon. Hasil pelatihan menunjukkan nilai *train loss* sekitar 0.003639 dan nilai validation loss sekitar 0.004814 sebagaimana ditampilkan pada Gambar 4.5.



Gambar 4.5 Grafik Nilai *Train* dan *Validation Loss* Model XGB-E2

Grafik nilai *train* dan *validation loss* pada Gambar 4.5 menunjukkan proses pelatihan model XGB-E2 yang terus mengalami penurunan hingga mencapai titik konvergensi pada n_estimators ke-200, sehingga model tersebut dianggap telah optimal. Selanjutnya, model optimal yang telah didapatkan, kemudian digunakan untuk melakukan klasifikasi serangan siber berdasarkan kategori serangan siber. Hasil dari klasifikasi tersebut disajikan pada Gambar 4.6.



Gambar 4.6 Hasil Klasifikasi Model XGB-E2 Terhadap Serangan Siber

Hasil klasifikasi model XGB-E2 pada Gambar 4.6 mempertahankan akurasi tinggi pada kelas utama seperti BEN 48.939 benar dan DoS 85.864 benar. Hasil ini menunjukkan kemampuan model dalam mengenali pola dominan masih kuat meskipun ukuran data uji lebih kecil. Kesalahan klasifikasi terutama muncul pada kelas NS dan TBF, di mana 168 data NS salah diarahkan ke TBF, serupa dengan pola pada XGB-E1. Namun, dibandingkan dengan model sebelumnya, tingkat kesalahan relatif menurun dan kestabilan prediksi meningkat, khususnya pada kelas PC3C yang kini berhasil diprediksi tanpa *error*. Hal ini menandakan peningkatan generalisasi model terhadap kelas dengan distribusi data minoritas. Selanjutnya, untuk menghitung nilai metrik akurasi, presisi, *recall*, dan *F1-score* maka dibutuhkan nilai TP, TN, FP, dan FN dari hasil klasifikasi serangan siber berdasarkan kategori serangan siber secara rinci disajikan dalam Tabel 4.5.

Tabel 4.5 Nilai TP, TN, FP, dan FN Setiap Kategori pada Model XGB-E2

Kelas	***True Positives* (TP)**	***True Negatives* (TN)**	***False Positives* (FP)**	***False Negatives* (FN)**
BEN	48939	91042	17	2
CAAD	1103	138897	0	0
DoS	85864	54116	5	15
ITT	134	139866	0	0
NS	2744	136930	155	171
PC3C	122	139878	0	0
TBF	750	138931	165	154

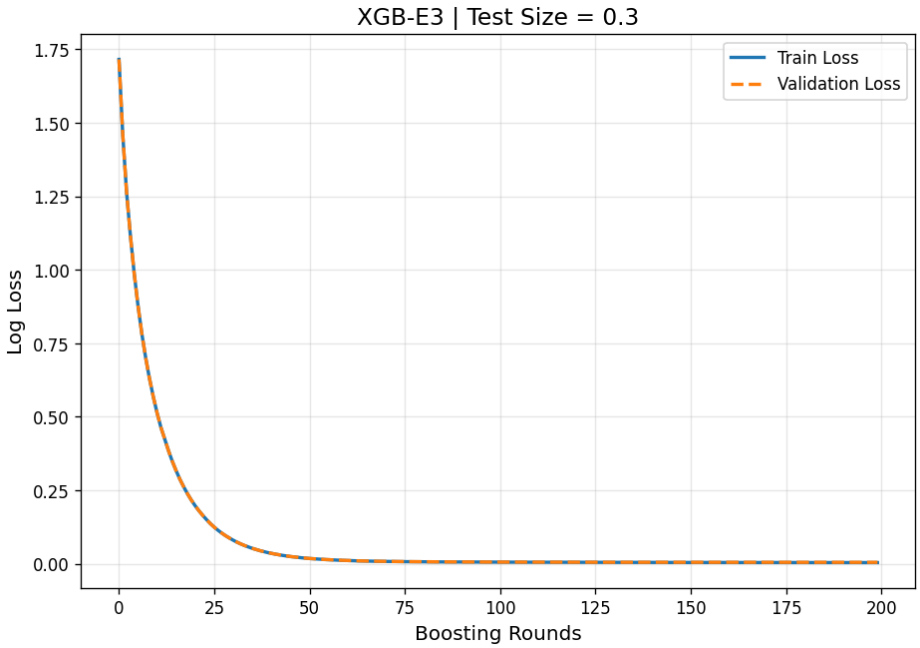
Setelah didapatkan nilai TP, TN, FP, dan FN dari setiap kategori serangan siber, langkah selanjutnya adalah menghitung metrik evaluasi yang digunakan untuk mengetahui performa model XGB-E2 dalam klasifikasi serangan siber pada IoT berdasarkan kategorinya yang secara rinci disajikan pada Tabel 4.6.

Tabel 4.6 Performa Hasil Pengujian Model XGB-E2

Kelas	**Akurasi (%)**	**Presisi**	***Recall***	***F1-Score***	***Support***
BEN	99.75	1	1	1	48941
CAAD		1	1	1	1103
DoS		1	1	1	85879
ITT		1	1	1	134
NS		0.95	0.94	0.94	2915
PC3C		1	1	1	122
TBF		0.82	0.83	0.82	904

4.2.3 Pelatihan Model XGB-E3

Pada proses eksperimen dari pelatihan model XGB-E3 dalam melakukan klasifikasi serangan siber pada IoT digunakan proporsi data latih sebesar 70% atau sekitar 245.000 data dan data uji sebesar 30% atau sekitar 105.000 data. Model dilatih dengan jumlah n_estimators yaitu sebanyak 200 pohon. Hasil pelatihan menunjukkan nilai *train loss* sekitar 0.003692 dan nilai *validation loss* sekitar 0.004646 sebagaimana ditampilkan pada Gambar 4.7.

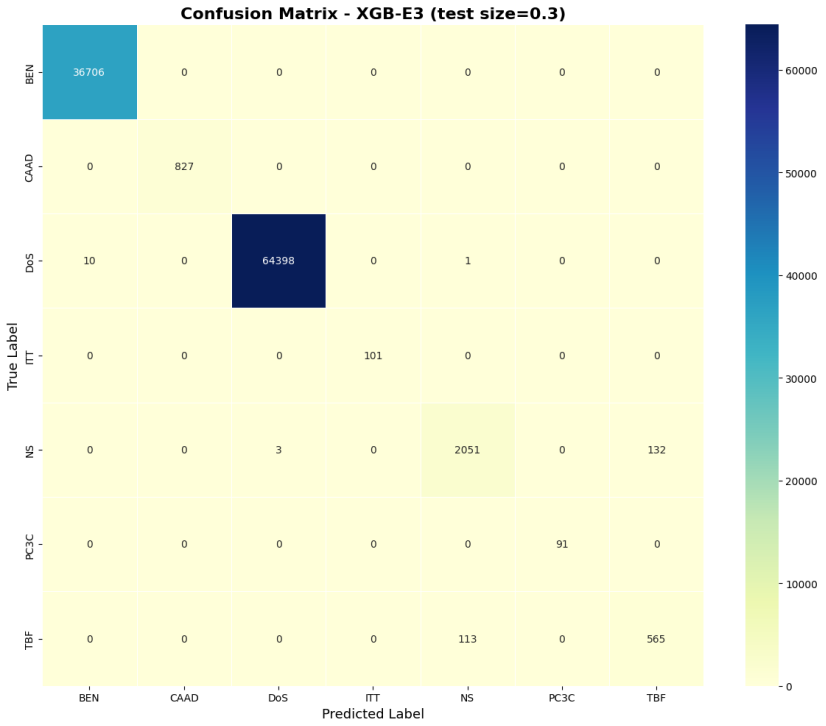


Gambar 4.7 Grafik Nilai *Train* dan *Validation Loss* Model XGB-E3

Grafik yang ditunjukkan pada Gambar 4.7 yaitu nilai *train* dan *validation* loss pada proses pelatihan model XGB-E3 terus mengalami penurunan hingga mencapai titik konvergensi pada n_estimators ke-200, sehingga model tersebut dianggap telah optimal. Selanjutnya, model optimal yang telah didapatkan, digunakan untuk melakukan klasifikasi serangan siber berdasarkan kategori serangan siber. Hasil dari klasifikasi tersebut disajikan pada Gambar 4.8.

Pada model XGB-E3, akurasi klasifikasi tetap tinggi pada kelas BEN, DoS, dan ITT, dengan kesalahan yang sangat kecil. Model memprediksi 64.398 data DoS dengan benar, menandakan kemampuan deteksi serangan DoS sangat konsisten. Meski demikian, pola kesalahan masih ditemukan pada kelas NS dan TBF, di mana 130 data NS salah dikenali sebagai TBF. Hasil ini mengindikasikan bahwa walaupun jumlah data latih meningkat dan model belajar lebih dalam, fitur dari dua

kelas tersebut masih sulit dipisahkan. Namun, tingkat overfitting mulai berkurang dan model menunjukkan stabilitas yang baik terhadap variasi data uji.



Gambar 4.8 Hasil Klasifikasi Model XGB-E3 Terhadap Serangan Siber

Bagian perhitungan nilai metrik akurasi, presisi, *recall*, dan *F1-score* maka dibutuhkan nilai TP, TN, FP, dan FN dari hasil klasifikasi serangan siber berdasarkan kategori serangan siber secara rinci disajikan dalam

Tabel 4.7.

Tabel 4.7 Nilai TP, TN, FP, dan FN Setiap Kategori pada Model XGB-E3

Kelas	***True Positives* (TP)**	***True Negatives* (TN)**	***False Positives* (FP)**	***False Negatives* (FN)**
BEN	36706	68284	10	0
CAAD	827	104173	0	0
DoS	64398	40588	3	11
ITT	101	104899	0	0
NS	2051	102700	114	135
PC3C	91	104909	0	0
TBF	565	104190	132	113

Setelah didapatkan nilai TP, TN, FP, dan FN dari setiap kategori serangan siber, langkah selanjutnya adalah menghitung metrik evaluasi yang digunakan untuk mengetahui performa model XGB-E3 dalam klasifikasi serangan siber pada IoT berdasarkan kategorinya yang secara rinci disajikan pada Tabel 4.8.

Tabel 4.8 Performa Hasil Pengujian Model XGB-E3

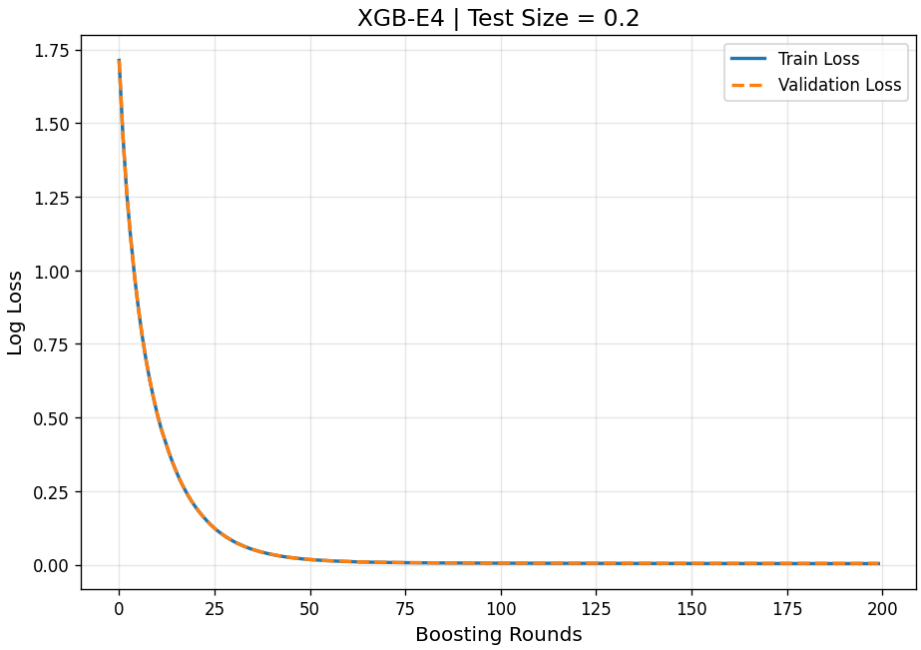
Kelas	**Akurasi (%)**	**Presisi**	***Recall***	***F1-Score***	***Support***
BEN	99.75	1	1	1	36706
CAAD		1	1	1	827
DoS		1	1	1	64409
ITT		1	1	1	101
NS		0.95	0.94	0.94	2186
PC3C		1	1	1	91
TBF		0.81	0.83	0.82	678

4.2.4 Pelatihan Model XGB-E4

Pada proses eksperimen dari pelatihan model XGB-E4 dalam melakukan klasifikasi serangan siber pada IoT digunakan proporsi data latih sebesar 80% atau sekitar 280.000 data dan data uji sebesar 20% atau sekitar 70.000 data. Model dilatih dengan jumlah n_estimators yaitu sebanyak 200 pohon. Hasil pelatihan menunjukkan nilai *train loss* sekitar 0.00367 dan nilai *validation loss* sekitar 0.004575 sebagaimana ditampilkan pada Gambar 4.9.

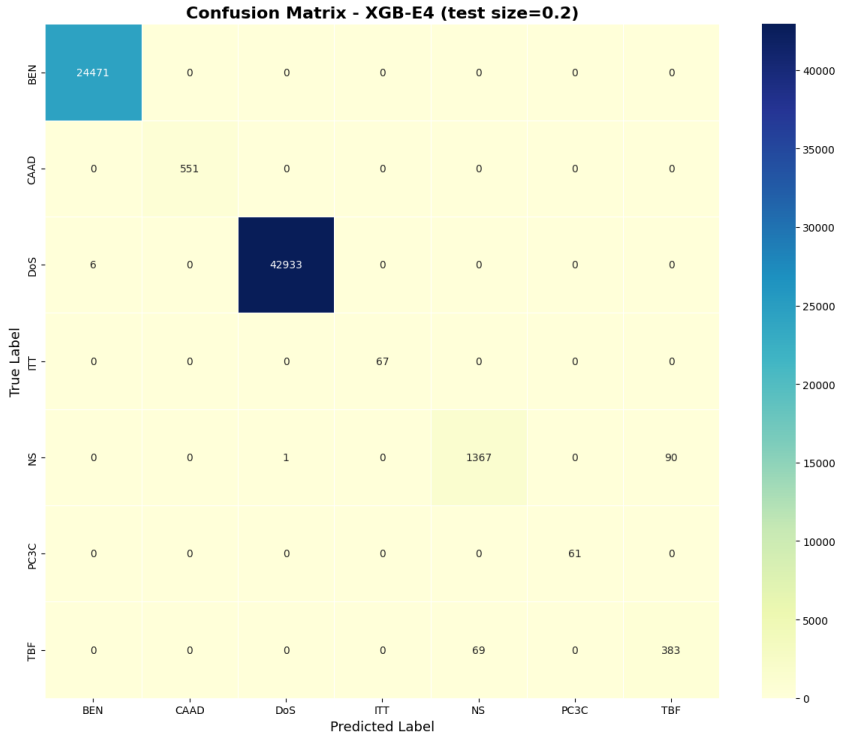
Gambar 4.9 menunjukkan perbandingan nilai *train loss* dan *validation loss* pada proses pelatihan model XGB-E4. Kedua kurva tampak menurun secara konsisten hingga mencapai titik konvergensi pada iterasi ke-200, sehingga model dapat dinilai telah mencapai performa pelatihan yang optimal. Selanjutnya, model optimal yang telah didapatkan, kemudian digunakan untuk melakukan klasifikasi

serangan siber berdasarkan kategori serangan siber. Hasil dari klasifikasi tersebut disajikan pada Gambar 4.10.



Gambar 4.9 Grafik Nilai *Train* dan *Validation Loss* Model XGB-E4

Hasil klasifikasi model XGB-E4 seperti pada Gambar 4.10 menampilkan hasil yang bagus dengan proporsi data latih yang lebih besar yaitu 80% dan data uji 20%. Hasil menunjukkan akurasi sempurna pada BEN, CAAD, DoS, dan ITT, kesalahan prediksi. Sementara itu, kelas NS masih memperlihatkan 82 kesalahan yang mengarah ke TBF. Meskipun demikian, dibandingkan dengan eksperimen sebelumnya, nilai kesalahan semakin kecil dan tingkat ketepatan meningkat di seluruh kelas. Hal ini menunjukkan bahwa model mulai mencapai titik keseimbangan optimal antara kompleksitas dan generalisasi.



Gambar 4.10 Hasil Klasifikasi Model XGB-E4 Terhadap Serangan Siber

Bagian untuk perhitungan nilai metrik akurasi, presisi, *recall*, dan *F1-score* maka dibutuhkan nilai TP, TN, FP, dan FN dari hasil klasifikasi serangan siber berdasarkan kategori serangan siber secara rinci disajikan dalam Tabel 4.9.

Tabel 4.9 Nilai TP, TN, FP, dan FN Setiap Kategori pada Model XGB-E4

Kelas	***True Positives* (TP)**	***True Negatives* (TN)**	***False Positives* (FP)**	***False Negatives* (FN)**
BEN	24471	45523	6	0
CAAD	551	69449	0	0
DoS	42933	27060	1	6
ITT	67	69933	0	0
NS	1367	68473	69	91
PC3C	61	69939	0	0
TBF	383	69458	90	69

Setelah didapatkan nilai TP, TN, FP, dan FN dari setiap kategori serangan siber, langkah selanjutnya adalah menghitung metrik evaluasi yang digunakan

untuk mengetahui performa model XGB-E4 dalam klasifikasi serangan siber pada IoT berdasarkan kategorinya yang secara rinci disajikan pada Tabel 4.10.

Tabel 4.10 Performa Hasil Pengujian Model XGB-E4

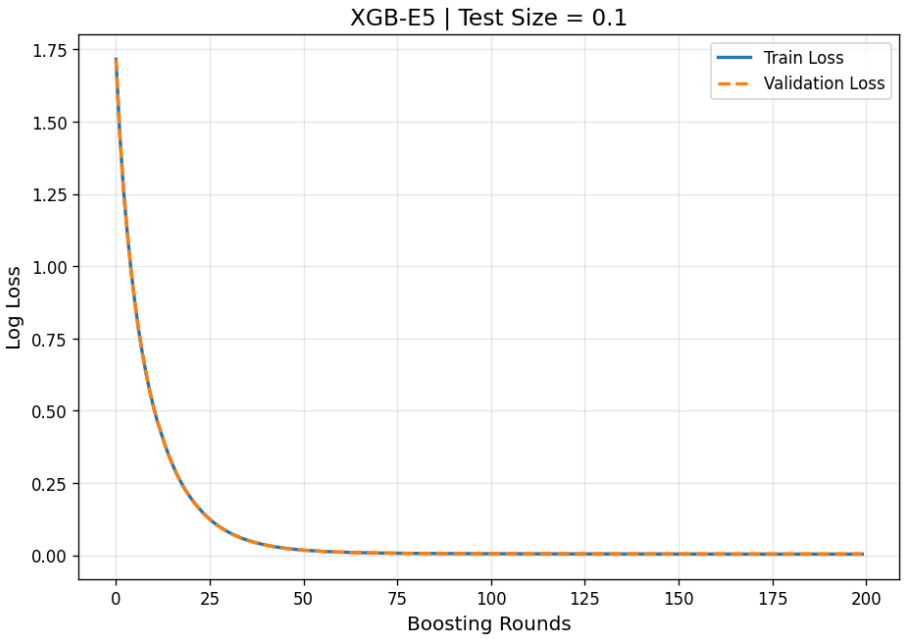
Kelas	**Akurasi (%)**	**Presisi**	***Recall***	***F1-Score***	***Support***
BEN	99.76	1	1	1	24471
CAAD		1	1	1	551
DoS		1	1	1	42939
ITT		1	1	1	67
NS		0.95	0.94	0.94	1458
PC3C		1	1	1	61
TBF		0.81	0.85	0.83	452

4.2.5 Pelatihan Model XGB-E5

Pada proses eksperimen dari pelatihan model XGB-E5 dalam melakukan klasifikasi serangan siber pada IoT digunakan proporsi data latih sebesar 90% atau sekitar 315.000 data dan data uji sebesar 10% atau sekitar 35.000 data. Model dilatih dengan jumlah n_estimators sebanyak 200 pohon. Hasil pelatihan menunjukkan nilai *train loss* sekitar 0.00367 dan nilai *validation lo*ss sekitar 0.004616 sebagaimana ditampilkan pada Gambar 4.11.

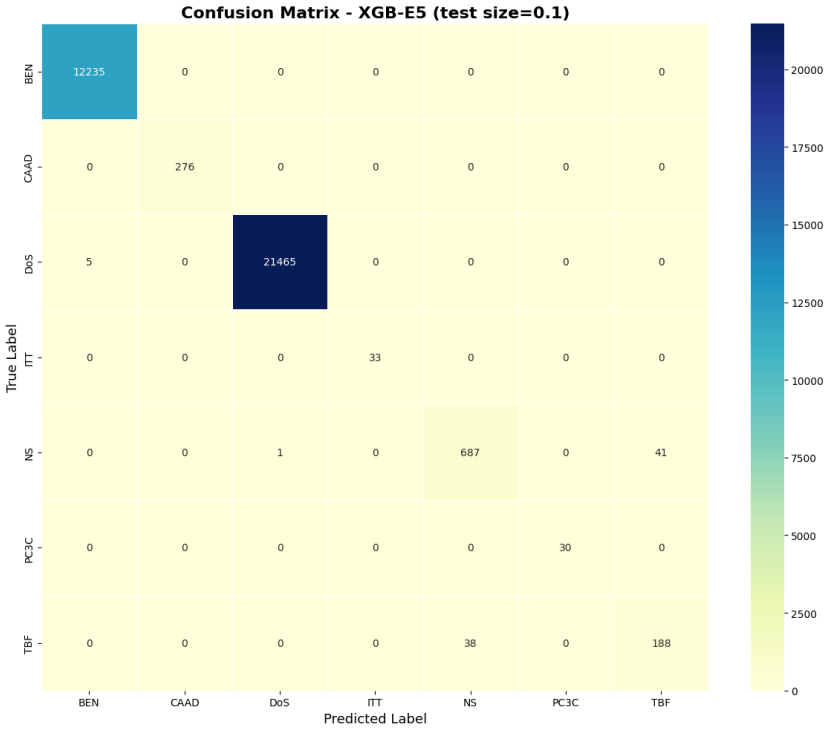
Gambar 4.11 menunjukkan nilai *train* dan *validation loss* pada proses pelatihan model XGB-E5 yang terus mengalami penurunan hingga mencapai titik konvergensi pada n_estimators ke-200, sehingga model tersebut dianggap telah optimal. Selanjutnya, model optimal yang telah didapatkan, kemudian digunakan untuk melakukan klasifikasi serangan siber berdasarkan kategori serangan siber, meliputi *Denial of Servic*e (DoS), *Benign* (BEN), *Network Scanning* (NS), CoAP *Amplification Attack Data* (CAAD), *Telnet Brute Forcing* (TBF), *Ingress Tool*

Transfer (ITT), dan *Periodic Command and Control* (C&C) *Communication* (PC3C). Hasil dari klasifikasi tersebut disajikan pada Gambar 4.12.



Gambar 4.11 Grafik Nilai *Train* dan *Validation Loss* Model XGB-E5

Hasil klasifikasi model XGB-E5 seperti pada Gambar 4.12 menunjukkan kinerja yang baik dengan porsi data latih 90% dan data uji 10%, model mampu memprediksi dengan akurasi hampir sempurna di semua kelas utama (BEN, CAAD, DoS, dan ITT). Sebanyak 21.465 data DoS terdeteksi benar, dan tidak ditemukan kesalahan pada kelas CAAD serta PC3C. Kesalahan minor masih muncul antara kelas NS dan TBF, dengan 38 data. Hal ini menunjukkan bahwa semakin besar porsi data latih, semakin baik model dalam mengenali pola fitur kompleks, mengurangi *noise*, serta meningkatkan presisi pada prediksi akhir.



Gambar 4.12 Hasil Klasifikasi Model XGB-E5 Terhadap Serangan Siber

Dengan demikian, model XGB-E5 dapat dikategorikan sebagai model terbaik dalam keseluruhan eksperimen karena mencapai keseimbangan maksimal antara sensitivitas dan akurasi lintas kelas. Selanjutnya, untuk menghitung nilai metrik akurasi, presisi, *recall*, dan *F1-score* maka dibutuhkan nilai TP, TN, FP, dan FN dari hasil klasifikasi serangan siber berdasarkan kategori serangan siber secara rinci disajikan dalam Tabel 4.11.

Tabel 4.11 Nilai TP, TN, FP, dan FN Setiap Kategori pada Model XGB-E5

Kelas	***True Positives* (TP)**	***True Negatives* (TN)**	***False Positives* (FP)**	***False Negatives* (FN)**
BEN	12235	22760	5	0
CAAD	276	34724	0	0
DoS	21465	13529	1	5
ITT	33	34967	0	0
NS	687	34233	38	42
PC3C	30	34970	0	0
TBF	188	34733	41	38

Setelah didapatkan nilai TP, TN, FP, dan FN dari setiap kategori serangan siber, langkah selanjutnya adalah menghitung metrik evaluasi yang digunakan untuk mengetahui performa model XGB-E5 dalam klasifikasi serangan siber pada IoT berdasarkan kategorinya yang secara rinci disajikan pada Tabel 4.12.

Tabel 4.12 Performa Hasil Pengujian Model XGB-E5

Kelas	**Akurasi (%)**	**Presisi**	***Recall***	***F1-Score***	***Support***
BEN	99.75	1	1	1	12235
CAAD		1	1	1	276
DoS		1	1	1	21470
ITT		1	1	1	33
NS		0.95	0.94	0.94	729
PC3C		1	1	1	30
TBF		0.82	0.83	0.83	226

4. 3 Hasil Ujicoba Model XGBoost *Non-Balancing*

Setelah proses pelatihan selesai, dilakukan proses ujicoba pada masing-masing model XGBoost *non-balancing* menggunakan data uji guna menilai kinerjanya dalam mengklasifikasikan serangan siber berdasarkan kategorinya. Selanjutnya, dilakukan proses evaluasi performa model menggunakan nilai akurasi, presisi, *recall*, dan *F1-score*, dimana masing-masing nilai memberikan perspektif berbeda terhadap performa model.

Tabel 4.13 Hasil Ujicoba Klasifikasi Serangan Siber pada Model XGBoost *Non-Balancing*

Model	**Kelas**	**Akurasi (%)**	**Presisi**	***Recall***	***F1-Score***	***Support***
XGB-E1	BEN	99.74	1	1	1	61176
	CAAD		1	1	1	1378
	DoS		1	1	1	107349
	ITT		1	0.99	1	168
	NS		0.94	0.94	0.94	3644
	PC3C		1	0.99	0.99	153

Model	Kelas	Akurasi (%)	Presisi	*Recall*	*F1-Score*	*Support*
	TBF		0.81	0.81	0.81	1129
XGB-E2	BEN	99.75	1	1	1	48941
	CAAD		1	1	1	1103
	DoS		1	1	1	85879
	ITT		1	1	1	134
	NS		0.95	0.94	0.94	2915
	PC3C		1	1	1	122
	TBF		0.82	0.83	0.82	904
XGB-E3	BEN	99.75	1	1	1	36706
	CAAD		1	1	1	827
	DoS		1	1	1	64409
	ITT		1	1	1	101
	NS		0.95	0.94	0.94	2186
	PC3C		1	1	1	91
	TBF		0.81	0.83	0.82	678
XGB-E4	BEN	99.76	1	1	1	24471
	CAAD		1	1	1	551
	DoS		1	1	1	42939
	ITT		1	1	1	67
	NS		0.95	0.94	0.94	1458
	PC3C		1	1	1	61
	TBF		0.81	0.85	0.83	452
XGB-E5	BEN	99.75	1	1	1	12235
	CAAD		1	1	1	276
	DoS		1	1	1	21470
	ITT		1	1	1	33
	NS		0.95	0.94	0.94	729
	PC3C		1	1	1	30
	TBF		0.82	0.83	0.83	226

Tabel 4.13 menunjukkan bahwa model XGB-E1 memiliki performa klasifikasi yang sangat baik pada hampir seluruh kelas serangan siber tanpa menggunakan metode penyeimbangan data. Kelas BEN (*Benign*), CAAD (CoAP *Amplification Attack Data*), dan DoS (*Denial of Service*) memperoleh hasil sempurna dengan nilai presisi, *recall*, dan *F1-score* sebesar 1, menunjukkan kemampuan model dalam mengidentifikasi data normal dan serangan utama secara akurat. Kelas ITT (*Ingress Tool Transfer*) juga menunjukkan performa sangat baik

dengan presisi 1, *recall* 0,99, dan *F1-score* 1, menandakan bahwa model mampu mengenali pola serangan IoT secara efektif. Namun, kelas NS (*Network Scanning*) dan TBF (*Telnet Brute Forcing*) memperlihatkan penurunan performa dengan *F1-score* 0,94 dan 0,81. Kesalahan klasifikasi ini menunjukkan bahwa model masih kesulitan membedakan pola serangan yang memiliki karakteristik trafik mirip. Sementara itu, kelas PC3C (*Communication*) tetap menunjukkan hasil sangat tinggi dengan *F1-score* 0,99 meskipun memiliki jumlah data yang sangat kecil. Secara umum, XGB-E1 menunjukkan stabilitas prediksi yang kuat pada kelas mayoritas, tetapi sedikit kurang optimal pada kelas minoritas.

Selanjutnya, model XGB-E2 memperlihatkan peningkatan performa terutama pada kelas minoritas. Kelas BEN, CAAD, DoS, dan ITT seluruhnya mencapai skor sempurna *F1-score* 1 yang menandakan tidak ada kesalahan prediksi pada kelas-kelas tersebut. Kelas NS menunjukkan peningkatan kecil dengan *F1-score* 0,94, sama seperti pada model sebelumnya, sementara TBF mengalami peningkatan dari *F1-score* 0,81 menjadi 0,82. Hal ini menunjukkan bahwa model mulai mampu menyesuaikan bobot pembelajaran terhadap variasi data meskipun tanpa penyeimbangan. Kelas PC3C juga menunjukkan performa sempurna *F1-score* 1, menandakan model memiliki ketahanan dalam mengenali pola serangan dengan jumlah data yang kecil. Secara keseluruhan, XGB-E2 memperlihatkan peningkatan stabilitas dibanding XGB-E1 dengan distribusi prediksi yang lebih konsisten di semua kelas.

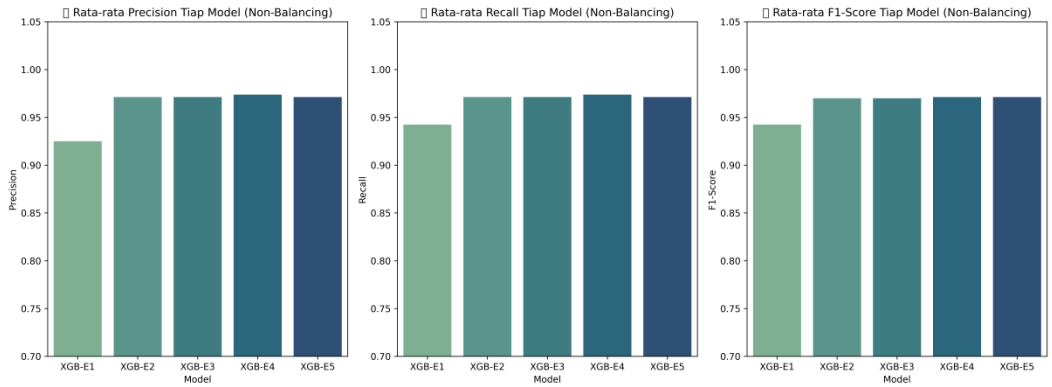
Model XGB-E3 menampilkan hasil yang serupa dengan XGB-E2, namun dengan performa yang lebih stabil dan generalisasi yang lebih baik. Semua kelas

mayoritas (BEN, CAAD, DoS, dan ITT) tetap mencatatkan presisi, *recall*, dan *F1-score* sempurna sebesar 1. Kelas NS menunjukkan hasil konstan dengan F1-score 0,94, sementara TBF sedikit meningkat menjadi 0,82. Kelas PC3C juga memperlihatkan hasil sempurna dengan *F1-score* 1, yang menunjukkan bahwa model memiliki kemampuan klasifikasi yang sangat kuat terhadap serangan minoritas. Secara umum, XGB-E3 menunjukkan bahwa dengan rasio pelatihan yang lebih besar, model dapat mempertahankan akurasi tinggi dan kestabilan deteksi antar kelas tanpa mengalami *overfitting* signifikan.

Pada model XGB-E4, performa klasifikasi semakin optimal dan konsisten di semua kelas. Kelas utama seperti BEN, CAAD, DoS, dan ITT tetap menunjukkan hasil sempurna *F1-score* 1. Kelas NS juga menunjukkan performa konsisten dengan *F1-score* 0,94, sedangkan kelas TBF mengalami peningkatan dengan *F1-score* 0,83. Hal ini mengindikasikan bahwa model mulai lebih baik dalam membedakan pola trafik yang kompleks meskipun data tidak seimbang. Kelas PC3C tetap mempertahankan nilai sempurna, menandakan kemampuan model yang baik dalam mengenali anomali serangan dengan jumlah data terbatas. Dengan komposisi data latih 80%, XGB-E4 dapat dikatakan mencapai keseimbangan ideal antara akurasi, stabilitas, dan generalisasi.

Sementara itu, model XGB-E5 yang menggunakan proporsi data latih paling besar (90%) menunjukkan performa paling stabil dan efisien di antara semua model *non-balancing*. Semua kelas utama (BEN, CAAD, DoS, dan ITT) kembali mencatatkan nilai presisi, *recall*, dan *F1-score* dengan baik. Kelas NS mempertahankan kinerja tinggi dengan *F1-score* 0,94, dan TBF memperlihatkan

peningkatan konsisten hingga mencapai *F1-score* 0,83. Selain itu, kelas PC3C juga menunjukkan hasil sempurna *F1-score* 1, mempertegas kemampuan model dalam mengenali serangan langka secara tepat. Secara keseluruhan, XGB-E5 merupakan varian terbaik dari seluruh model *non-balancing* karena berhasil mempertahankan akurasi tinggi secara global, meminimalkan kesalahan klasifikasi, dan menunjukkan ketahanan prediksi yang kuat terhadap seluruh jenis serangan siber.



Gambar 4.13 Rata-Rata Performa Model XGBoost *Non-Balancing*

Secara keseluruhan, hasil pengujian terhadap kelima model XGBoost (XGB-E1 hingga XGB-E5) seperti pada gambar Gambar 4.13 menunjukkan bahwa peningkatan proporsi data latih berkontribusi positif terhadap stabilitas, akurasi, dan kemampuan generalisasi model dalam mendeteksi serangan siber. Setiap peningkatan rasio pelatihan menghasilkan perbaikan performa, terutama pada kelas minoritas seperti TBF yang awalnya memiliki nilai *F1-score* rendah. Dengan bertambahnya jumlah data latih, model mampu mengenali pola trafik yang lebih kompleks dan mengurangi kesalahan klasifikasi pada kelas dengan karakteristik mirip. Model dengan proporsi data latih tertinggi, XGB-E5 (90:10), menjadi varian paling optimal dengan hasil sempurna pada sebagian besar kelas. Peningkatan

signifikan juga terlihat pada kelas TBF dan NS yang menunjukkan akurasi lebih stabil dibanding model sebelumnya. Hal ini menunjukkan bahwa XGBoost sensitif terhadap jumlah data pelatihan yang besar karena variasi data yang lebih banyak meningkatkan kemampuan diskriminatif model. Dengan demikian, strategi *non-balancing* tetap efektif selama proporsi data latih cukup besar, dan XGB-E5 terbukti paling andal dalam mendeteksi ancaman siber secara akurat dan efisien.

Tabel 4.14 Performa Model XGBoost *Non-Balancing*

Model	**Komposisi Data**	**Nilai Akurasi**
XGB-E1	50:50	99.74%
XGB-E2	60:40	99.75%
XGB-E3	70:30	99.75%
XGB-E4	**80:20**	**99.76%**
XGB-E5	90:10	99.75%

Tabel 4.14 menunjukkan bahwa model XGB-E1 dengan komposisi data 50:50 memperoleh nilai akurasi sebesar 99,74%, yang menandakan bahwa meskipun tanpa proses penyeimbangan data, model XGBoost mampu melakukan klasifikasi dengan tingkat ketepatan yang sangat tinggi. Peningkatan proporsi data latih pada model XGB-E2 (60:40) menghasilkan akurasi 99,75%, memperlihatkan adanya pengaruh positif dari jumlah data pelatihan terhadap kemampuan model dalam mengenali pola serangan.

Model XGB-E3 (70:30) mempertahankan akurasi 99,75%, menunjukkan konsistensi performa ketika proporsi data latih ditingkatkan lebih lanjut. Sementara itu, model XGB-E4 (80:20) mencatatkan hasil tertinggi dengan akurasi sebesar 99,76%, menjadikannya sebagai model terbaik di antara kelima konfigurasi karena mampu mencapai keseimbangan ideal antara jumlah data pelatihan dan kemampuan generalisasi model terhadap data uji. Adapun pada model XGB-E5 (90:10), akurasi

sedikit menurun kembali menjadi 99,75%, yang mengindikasikan bahwa penambahan data latih setelah titik tertentu tidak selalu memberikan peningkatan signifikan terhadap kinerja model.

Secara keseluruhan, seluruh model XGBoost *non-balancing* menunjukkan performa yang sangat stabil dan unggul dengan akurasi di atas 99,7%, namun model XGB-E4 dengan nilai akurasi sekitar 99,76% dapat dikategorikan sebagai konfigurasi paling optimal karena mencapai nilai akurasi tertinggi sekaligus mempertahankan kestabilan hasil pada seluruh kelas serangan siber.

Tabel 4.15 Nilai TP, TN, FP, dan FN Model XGB-E4

Kelas	***True Positives* (TP)**	***True Negatives* (TN)**	***False Positives* (FP)**	***False Negatives* (FN)**
BEN	24471	45523	6	0
CAAD	551	69449	0	0
DoS	42933	27060	1	6
ITT	67	69933	0	0
NS	1367	68473	69	91
PC3C	61	69939	0	0
TBF	383	69458	90	69

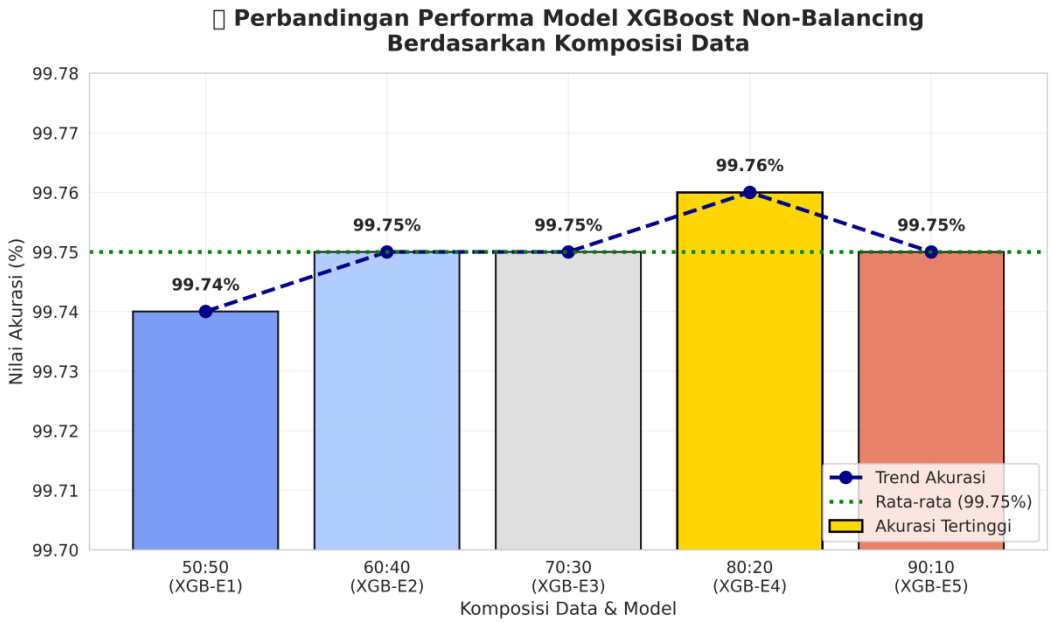
Tabel 4.15 ditunjukkan hasil pengujian model XGB-E4 terhadap masing-masing kelas serangan berdasarkan nilai *True Positives* (TP), T*rue Negatives* (TN), *False Positives* (FP), dan *False Negatives* (FN). Secara umum, hasil tersebut memperlihatkan bahwa model XGB-E4 mampu melakukan klasifikasi dengan tingkat ketepatan yang sangat tinggi di hampir seluruh kelas, baik dalam mendeteksi serangan maupun mengidentifikasi lalu lintas normal.

Kelas BEN (*Benign*) memiliki nilai TP = 24.471 dan TN = 45.523, dengan FP = 6 dan FN = 0, yang menunjukkan bahwa model mampu mengenali lalu lintas normal dengan baik tanpa kesalahan signifikan. Hal ini menandakan tingkat akurasi

yang sangat tinggi dalam membedakan data *Benign* dari data serangan. Pada kelas CAAD (*Command and Control Attack Detection*), model mencatat TP = 551, TN = 69.449, dan tidak memiliki FP maupun FN (0), yang berarti model berhasil mendeteksi seluruh serangan jenis ini dengan baik tanpa kesalahan klasifikasi sama sekali. Sementara itu, kelas DoS (*Denial of Service*) memperoleh TP = 42.933 dan TN = 27.060, dengan FP = 1 dan FN = 6. Nilai ini menunjukkan performa yang sangat kuat, di mana hanya terdapat kesalahan sangat kecil dalam mendeteksi serangan DoS yang sangat masif jumlahnya. Untuk kelas ITT (Ingress Tool Transfer), nilai TP = 67 dan TN = 69.933, dengan FP = 0 dan FN = 0, menandakan bahwa model mampu mengklasifikasikan seluruh data dengan benar.

Selanjutnya, kelas NS (*Network Scan*) menunjukkan TP = 1.367, TN = 68.473, dengan FP = 69 dan FN = 91. Meskipun akurasi masih sangat baik, nilai FP dan FN yang sedikit lebih tinggi dibandingkan kelas lain mengindikasikan bahwa model memiliki sedikit kesulitan dalam membedakan pola serangan *scanning* dari lalu lintas normal atau jenis serangan lain yang mirip. Pada kelas PC3C (*Communication*), diperoleh TP = 61, TN = 69.939, dengan FP = 0 dan FN = 0, yang kembali menunjukkan deteksi sempurna tanpa adanya kesalahan prediksi. Terakhir, kelas TBF (*Telnet Brute Forcing*) menunjukkan TP = 383, TN = 69.458, FP = 90, dan FN = 69. Nilai ini menunjukkan bahwa meskipun model secara umum bekerja sangat baik, masih terdapat sejumlah kecil kesalahan dalam mengidentifikasi serangan *brute force*, kemungkinan karena karakteristiknya yang mirip dengan jenis serangan lain.

Secara keseluruhan, model XGB-E4 menunjukkan kemampuan klasifikasi yang sangat akurat dan stabil di seluruh kelas, dengan tingkat FP dan FN yang sangat rendah, serta TP dan TN yang tinggi. Hal ini membuktikan bahwa model mampu mengenali berbagai pola serangan siber dengan efektif, khususnya pada kelas CAAD, ITT, dan PC3C yang menunjukkan deteksi sempurna, sementara kelas NS dan TBF masih dapat dioptimalkan melalui penyesuaian fitur atau parameter model untuk meningkatkan sensitivitas terhadap pola serangan yang lebih kompleks.



Gambar 4.14 Perbandingan Performa Model XGBoost *Non-Balancing*

Temuan penelitian ini menunjukkan bahwa proporsi data latih dan uji yang tepat sangat berpengaruh terhadap generalisasi model XGBoost. Gambar 4.14 menunjukkan komposisi 80:20 memberikan keseimbangan terbaik antara kompleksitas pembelajaran dan kemampuan prediksi pada data baru.

BAB V

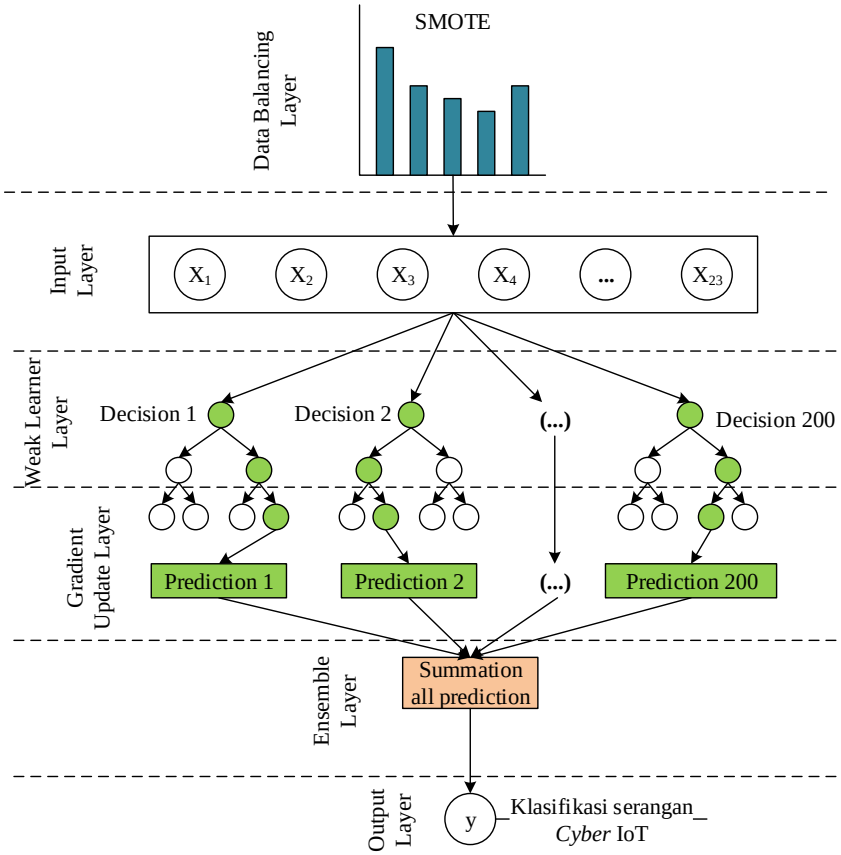
MODEL XGBOOST DENGAN PRA-PEMROSESAN SMOTE

5.1 Desain Model XGBS

Tahapan penelitian ini menggunakan model XGBoost dengan pra-pemrosesan SMOTE untuk melakukan klasifikasi serangan siber. Model ini dipilih karena memiliki kemampuan adaptif dalam memperbaiki kesalahan melalui boosting bertingkat, serta diperkuat dengan teknik *oversampling* data (SMOTE) yang bertujuan mengatasi ketidakseimbangan distribusi kelas dalam data serangan siber. Kombinasi antara XGBoost dan SMOTE diharapkan dapat menghasilkan model yang lebih akurat, adil, dan stabil dalam mengenali pola serangan, termasuk pada kelas minoritas yang sebelumnya cenderung terabaikan. Gambaran lebih rinci mengenai rancangan arsitektur model XGBoost dengan SMOTE ini dapat dilihat pada Gambar 5.1.

Proses *training* model XGBS dimulai dengan memanggil data *training* mentah yang berisi berbagai fitur serangan siber yang telah melalui tahap pra-pemrosesan awal seperti pembersihan data. Tahap berikutnya adalah penerapan SMOTE (*Synthetic Minority Over-sampling Technique*), yaitu proses menyeimbangkan distribusi kelas dengan cara menghasilkan sampel sintetis untuk kelas minoritas berdasarkan kemiripan antar-tetangga. Langkah ini memastikan bahwa model tidak bias terhadap kelas mayoritas dan mampu mengenali variasi serangan secara proporsional. Setelah data berada dalam kondisi seimbang, proses

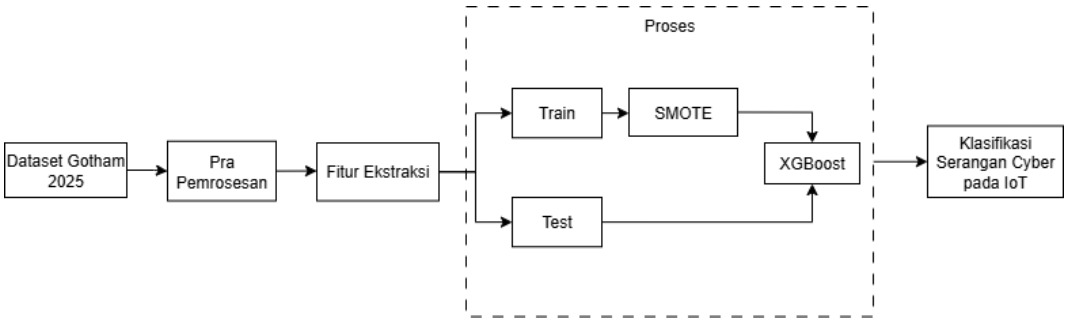
pelatihan dilanjutkan dengan pembangunan model XGBoost menggunakan parameter sesuai dengan hasil pengujian pada Model XGBoost *non-balancing*.



Gambar 5.1 Arsitektur Model XGBS

Hasil prediksi dari seluruh pohon keputusan kemudian digabungkan menggunakan mekanisme *weighted boosting*, di mana bobot setiap pohon ditentukan berdasarkan tingkat kesalahannya selama pelatihan. Proses *training* berlangsung hingga seluruh 200 pohon terbentuk atau model mencapai titik konvergensi pada kestabilan performa. Setelah pelatihan selesai, model XGBoost-SMOTE yang telah terlatih akan menyimpan struktur pohon, bobot *boosting*, dan parameter terbaik untuk digunakan pada tahap pengujian (*testing*). Desain sistem dari keseluruhan proses pelatihan dan penerapan SMOTE hingga pembentukan

model XGBoost ditunjukkan pada Gambar 5.2, yang menggambarkan gambaran kerja sistem mulai dari pra-pemrosesan data, *balancing* dengan SMOTE, pelatihan bertingkat XGBoost, hingga tahap evaluasi akhir model.



Gambar 5.2 Desain Sistem Model XGBS

5.2 Implementasi Model XGBS

Proses pengolahan data dan implementasi model dilakukan menggunakan Google Colab berbasis Python, yang memungkinkan pelaksanaan komputasi GPU secara efisien untuk mempercepat pelatihan model. Sebelum pelatihan dilakukan, dataset dibagi menjadi dua bagian utama data latih (*training*) dan data uji (*testing*) guna memastikan model dapat melakukan generalisasi terhadap data baru yang belum pernah dilihat sebelumnya. Dataset yang digunakan mencakup berbagai jenis serangan siber seperti *Denial of Servic*e (DoS), *Benign* (BEN), *Network Scanning* (NS), CoAP *Amplification Attack Data* (CAAD), *Telnet Brute Forcing* (TBF), *Ingress Tool Transfer* (ITT), dan *Periodic Command and Control* (C&C) *Communication* (PC3C).

Parameter n_estimators = 200 digunakan untuk menentukan jumlah pohon keputusan yang dibangun secara berurutan dalam ensemble model. Selanjutnya,

learning_rate = 0.1 berfungsi untuk mengontrol seberapa besar kontribusi setiap pohon terhadap model akhir, sehingga proses pembelajaran berlangsung secara bertahap dan terhindar dari overfitting yang terlalu cepat. Parameter max_depth = 6 diterapkan untuk membatasi kedalaman maksimum setiap pohon keputusan, agar kompleksitas model tetap terjaga tanpa kehilangan kemampuan dalam menangkap pola data yang signifikan. Sementara itu, parameter subsample = 0.8 dan colsample_bytree = 0.8 digunakan untuk mengatur proporsi data dan fitur yang digunakan dalam setiap pohon, dengan tujuan meningkatkan generalisasi model serta mengurangi risiko redundansi antar pohon. Adapun eval_metric = 'mlogloss' ditetapkan sebagai metrik evaluasi utama untuk mengukur seberapa baik model dalam memprediksi probabilitas kelas yang benar pada kasus klasifikasi multi-kelas. Tabel 5.1 merupakan rincian parameter model yang digunakan pada XGBoost dan SMOTE.

Tabel 5.1 Parameter Model XGBS

Parameter	Nilai	Keterangan	Penjelasan
XGBoost			
n_estimators	200	Jumlah Pohon	Menentukan jumlah pohon keputusan yang akan dibangun dalam model.
learning_rate	0.1	Laju Pembelajaran	Mengontrol seberapa besar kontribusi setiap pohon baru terhadap model akhir.
max_depth	6	Kedalaman Maksimum Pohon	Menentukan kedalaman maksimum setiap pohon keputusan.
subsample	0.8	Proporsi Subset Data	Mengatur persentase data pelatihan yang digunakan untuk membangun setiap pohon.

Parameter	Nilai	Keterangan	Penjelasan
colsample_bytree	0.8	Proporsi Fitur per Pohon	Menentukan proporsi fitur (kolom) yang akan digunakan dalam pembangunan setiap pohon.
random_state	42	Pengaturan Angka Acak	Menetapkan *seed* acak untuk memastikan hasil eksperimen dapat direproduksi secara konsisten setiap kali model dijalankan.
use_label_encoder	FALSE	Penggunaan *Label Encoder* Internal	Menonaktifkan *encoder* label bawaan XGBoost agar tidak terjadi konflik dengan proses encoding yang telah dilakukan sebelumnya dalam tahap *preprocessing*.
eval_metric	'mlogloss'	Metrik Evaluasi	Menggunakan *multiclass log loss* sebagai metrik evaluasi utama untuk mengukur seberapa baik model memprediksi probabilitas kelas yang benar pada data multi-kelas.
SMOTE			
sampling_strategy	'auto' (default)	Strategi *Sampling*	Menentukan rasio antara kelas minoritas dan mayoritas.
random_state	42	Pengaturan Angka Acak	Menjamin hasil proses *oversampling* dapat direproduksi secara konsisten setiap kali kode dijalankan.

Parameter	Nilai	Keterangan	Penjelasan
k_neighbors	max(1, min_class_count - 1)	Jumlah Tetangga Terdekat	Menentukan jumlah tetangga terdekat yang digunakan untuk membangkitkan sampel sintetis baru.
n_jobs	None (default)	Jumlah CPU yang Digunakan	Menentukan berapa banyak core CPU yang digunakan untuk menjalankan proses SMOTE secara paralel.
categorical_features	None (default)	Fitur Kategorikal	Menentukan apakah ada fitur kategorikal yang harus ditangani secara khusus.

Selain parameter inti XGBoost, penelitian ini juga menerapkan *Synthetic Minority Over-sampling Technique* (SMOTE) sebagai strategi *balancing* untuk mengatasi ketidakseimbangan distribusi kelas pada data pelatihan. Penerapan SMOTE dilakukan sebelum proses pelatihan model agar setiap kelas memiliki representasi yang seimbang, sehingga model dapat belajar dengan lebih objektif dan tidak bias terhadap kelas mayoritas. Parameter utama yang digunakan dalam proses SMOTE antara lain sampling_strategy = auto agar sistem secara otomatis menyeimbangkan seluruh kelas, random_state = 42 untuk memastikan hasil proses oversampling dapat direproduksi secara konsisten, k_neighbors = max(1, min_class_count - 1) agar secara dinamis menyesuaikan jumlah tetangga terdekat berdasarkan jumlah minimum sampel pada kelas minoritas, serta n_jobs = None dan categorical_features = None, yang berguna masing-masing untuk mengatur

penggunaan sumber daya CPU dan memastikan bahwa seluruh fitur yang digunakan telah berbentuk numerik.

Proses SMOTE dilakukan setelah pembagian data latih dan uji, di mana kelas minoritas diperbanyak secara sintetis dengan memanfaatkan kemiripan antar-sampel. Untuk mengevaluasi pengaruh variasi proporsi data latih dan uji terhadap performa model, penelitian ini menggunakan lima skenario pelatihan bertahap yang masing-masing diberi kode XGBS-E1 hingga XGBS-E5. Setiap skenario memiliki perbedaan komposisi data sebagaimana ditunjukkan pada Tabel 5.2, dengan rincian sebagai berikut:

Tabel 5.2 Distribusi Data Pelatihan Model XGBS

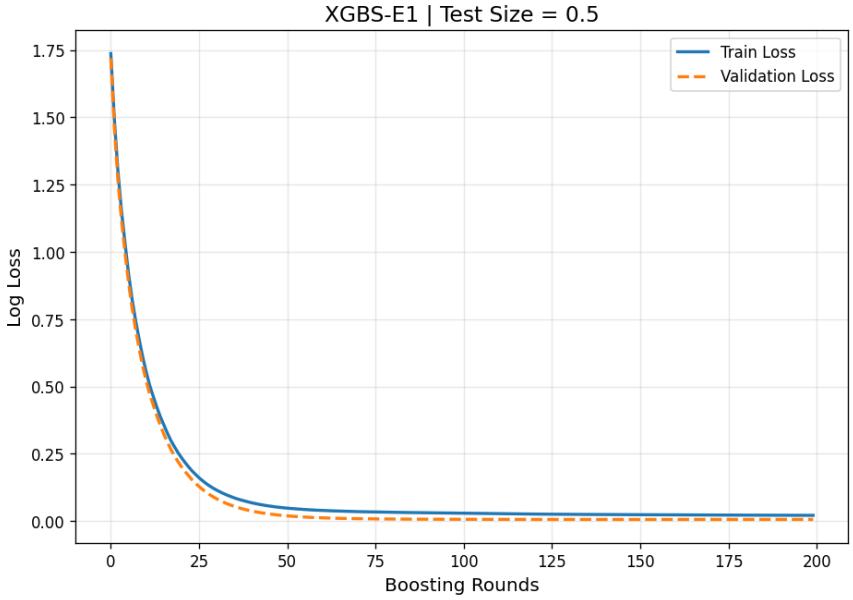
Nama Pelatihan	Komposisi Data	Data Latih	SMOTE	Data Uji
XGBS-E1	50:50	175.000	858.784	175.000
XGBS-E2	60:40	210.000	1.030.544	140.000
XGBS-E3	70:30	245.000	1.202.304	105.000
XGBS-E4	80:20	280.000	1.374.064	70.000
XGBS-E5	90:10	315.000	1.545.816	35.000

Setiap model dijalankan dengan parameter dan konfigurasi yang sama, namun menggunakan proporsi data yang berbeda untuk menguji stabilitas generalisasi dan konsistensi hasil klasifikasi. Dengan konfigurasi tersebut, model XGBoost menggunakan SMOTE diharapkan mampu memberikan hasil klasifikasi yang lebih stabil, adil, dan efisien dalam mendeteksi berbagai jenis serangan siber pada lingkungan *Internet of Things* (IoT).

5.2.1 Pelatihan Model XGBS-E1

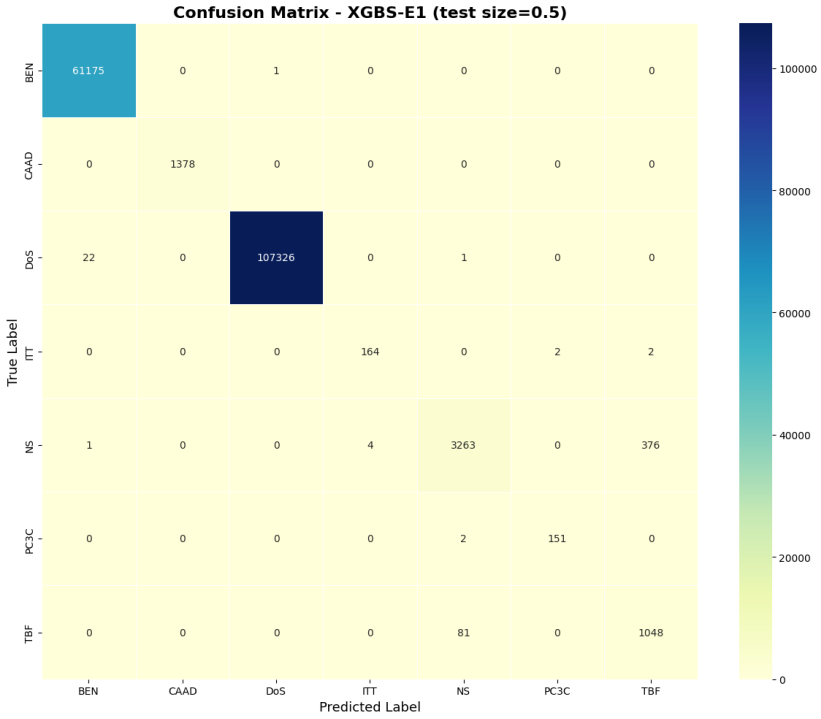
Pada proses eksperimen dari pelatihan model XGBS-E1 dalam melakukan klasifikasi serangan siber pada IoT digunakan proporsi data latih sebesar 50% atau sekitar 175.000 data dan data uji sebesar 50% atau sekitar 175.000 data. Setelah

dilakukan proses SMOTE jumlah data *training* menjadi 858.784 data dengan masing-masing kelas sejumlah 107.348 data. Model dilatih dengan jumlah n_estimators yang telah ditentukan, yaitu sebanyak 200 pohon. Hasil pelatihan menunjukkan nilai *train loss* sekitar 0.021555 dan nilai *validation loss* sekitar 0.006374 sebagaimana ditampilkan pada Gambar 5.3.



Gambar 5.3 Grafik Nilai *Train* dan *Validation Loss* Model XGBS-E1

Gambar 5.3 menunjukkan nilai *train* dan *validation loss* pada proses pelatihan model XGBS-E1 yang terus mengalami penurunan hingga mencapai titik konvergensi pada n_estimators ke-200, sehingga model tersebut dianggap telah optimal. Selanjutnya, model optimal yang telah didapatkan, kemudian digunakan untuk melakukan klasifikasi serangan siber berdasarkan kategori serangan siber. Hasil dari klasifikasi tersebut disajikan pada Gambar 4.15.



Gambar 4.15 Hasil Klasifikasi Model XGBS-E1 Terhadap Serangan Siber

Gambar 4.15 merupakan hasil klasifikasi dari model XGBS-E1 yang menunjukkan performa yang sangat baik pada sebagian besar kelas, terutama BEN, CAAD, DoS, dan ITT. Kelas BEN mencatat 61.175 data benar dengan hanya 1 kesalahan kecil, sedangkan CAAD sepenuhnya akurat dengan 1.378 data benar tanpa *error*. Kelas DoS menjadi kelas dengan performa terbaik, menampilkan 107.326 data benar dari total data uji dan hanya 23 kesalahan minor ke BEN dan NS. Sementara itu, ITT juga tampil baik dengan 164 data benar dan 4 kesalahan kecil ke PC3C dan TBF. Namun, kelemahan model muncul pada kelas NS dan TBF, di mana NS hanya memiliki 3.263 data benar dan 381 salah klasifikasi dengan 376 di antaranya salah diarahkan ke TBF. Begitu pula pada TBF, terdapat 1.048 data benar tetapi masih menerima 81 data salah dari NS. Pola ini menandakan bahwa

model mengalami ambiguitas antar dua kelas tersebut, kemungkinan karena kesamaan pola lalu lintas.

Secara keseluruhan, XGBS-E1 memiliki akurasi tinggi pada kelas dominan, namun masih menghadapi tantangan pada kelas dengan fitur yang tumpang tindih. Nilai TP, TN, FP, dan FN dari hasil klasifikasi serangan siber berdasarkan kategori serangan siber secara rinci disajikan dalam Tabel 5.3.

Tabel 5.3 Nilai TP, TN, FP, dan FN Setiap Kategori pada Model XGBS-E1

Kelas	***True Positives (TP)***	***True Negatives (TN)***	***False Positives (FP)***	***False Negatives (FN)***
BEN	61175	113800	24	1
CAAD	1378	173622	0	0
DoS	107326	67646	5	23
ITT	164	174832	0	4
NS	3263	171272	84	381
PC3C	151	174845	2	2
TBF	1048	173493	378	81

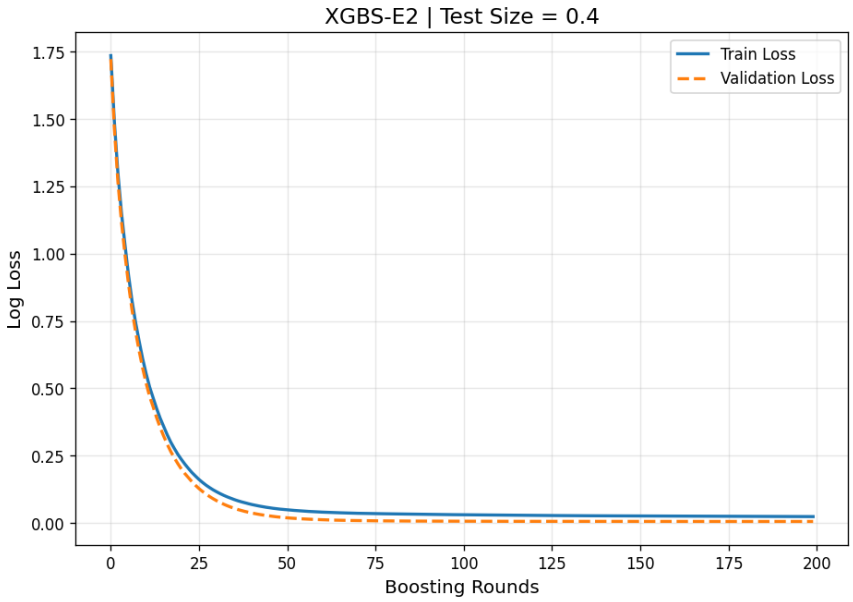
Setelah didapatkan nilai TP, TN, FP, dan FN dari setiap kategori serangan siber, langkah selanjutnya adalah menghitung metrik evaluasi. Metrik Evaluasi digunakan untuk mengetahui performa model XGBS-E1 dalam klasifikasi serangan siber pada IoT berdasarkan kategorinya yang secara rinci disajikan pada Tabel 5.4.

Tabel 5.4 Performa Hasil Pengujian Model XGBS-E1

Kelas	**Akurasi (%)**	**Presisi**	***Recall***	***F1-Score***	***Support***
BEN	99.71	1	1	1	61176
CAAD		1	1	1	1378
DoS		1	1	1	107349
ITT		1	0.97	0.98	168
NS		0.97	0.9	0.93	3644
PC3C		0.99	0.99	0.99	153
TBF		0.74	0.91	0.82	1129

5.2.2 Pelatihan Model XGBS-E2

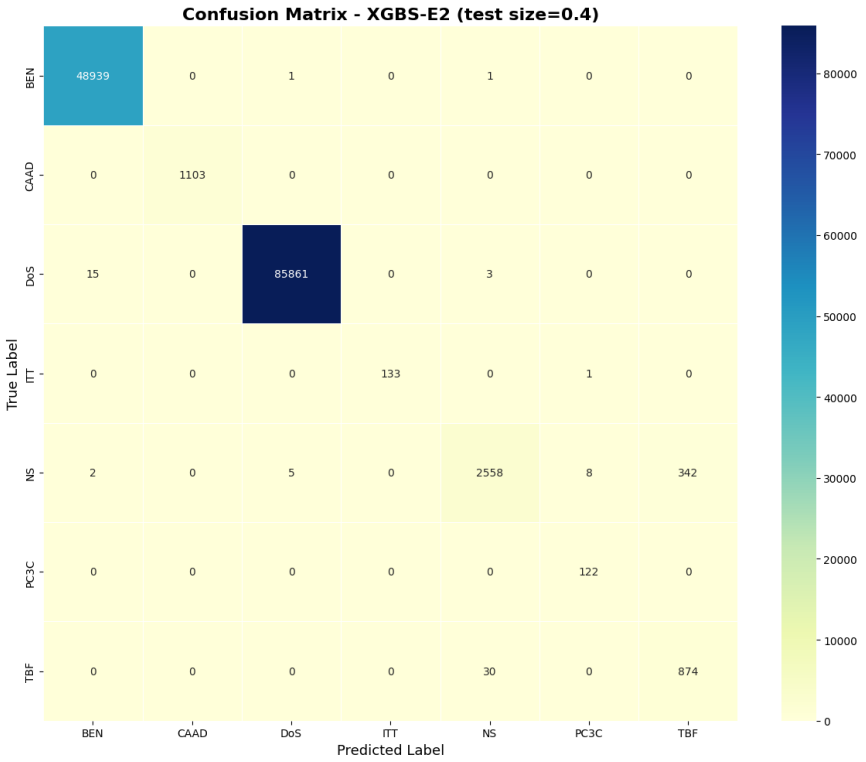
Pada proses eksperimen dari pelatihan model XGBS-E2 dalam melakukan klasifikasi serangan siber pada IoT digunakan proporsi data latih sebesar 60% atau sekitar 210.000 data dan data uji sebesar 40% atau sekitar 140.000 data. Setelah dilakukan proses SMOTE jumlah data *training* menjadi 1.030.544 data dengan masing-masing kelas sejumlah 128.818 data. Model dilatih dengan jumlah n_estimators yang telah ditentukan, yaitu sebanyak 200 pohon. Hasil pelatihan menunjukkan nilai *train loss* sekitar 0.024142 dan nilai *validation loss* sekitar 0.00609 sebagaimana ditampilkan pada Gambar 4.16.



Gambar 4.16 Grafik Nilai *Train* dan *Validation Loss* Model XGBS-E2

Gambar 4.16 menunjukkan nilai *train* dan *validation loss* pada proses pelatihan model XGBS-E2 yang terus mengalami penurunan hingga mencapai titik konvergensi pada n_estimators ke-200, sehingga model tersebut dianggap telah optimal. Selanjutnya, model optimal yang telah didapatkan, kemudian digunakan untuk melakukan klasifikasi serangan siber berdasarkan kategori serangan siber.

Hasil dari klasifikasi tersebut disajikan dalam bentuk *confusion matrix* pada Gambar 4.17.



Gambar 4.17 Hasil Klasifikasi Model XGBS-E2 Terhadap Serangan Siber

Model XGBS-E2 tetap menunjukkan kinerja stabil dengan hasil klasifikasi yang kuat pada sebagian besar kelas utama. Sesuai dengan Gambar 4.17 kelas BEN memperoleh 48.939 data benar dengan hanya 2 kesalahan, sedangkan CAAD kembali sempurna dengan 1.103 data benar tanpa *error*. Kelas DoS juga menunjukkan ketepatan tinggi dengan 85.861 data benar dan hanya 18 kesalahan minor. Untuk ITT, model berhasil mengklasifikasikan 133 data benar dari total sampel dengan tingkat akurasi mendekati 100%.

Kinerja sedikit menurun pada kelas NS, di mana hanya 2.558 data benar dari total data uji, dengan 357 salah klasifikasi, sebagian besar 342 data salah dikenali

sebagai TBF. Kelas TBF sendiri memiliki 874 data benar, namun juga menerima 30 data salah dari NS. Meski kesalahan silang antara NS dan TBF masih terjadi, jumlahnya berkurang dibanding XGBS-E1. Hal ini menunjukkan bahwa model mulai belajar membedakan pola trafik lebih efektif seiring peningkatan rasio data latih. Nilai TP, TN, FP, dan FN dari hasil klasifikasi serangan siber berdasarkan kategori serangan siber secara rinci disajikan dalam Tabel 5.5.

Tabel 5.5 Nilai TP, TN, FP, dan FN Setiap Kategori pada Model XGBS-E2

Kelas	***True Positives* (TP)**	***True Neg*atives (TN)**	***False Positives* (FP)**	***False Negatives* (FN)**
BEN	48939	91042	17	2
CAAD	1103	138897	0	0
DoS	85861	54115	6	18
ITT	133	139866	0	1
NS	2558	137051	34	357
PC3C	122	139869	9	0
TBF	874	138754	342	30

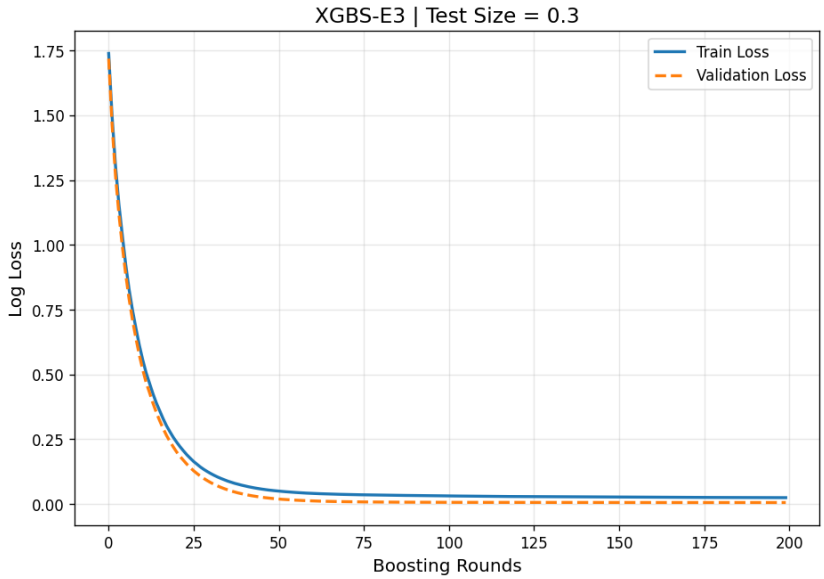
Setelah didapatkan nilai TP, TN, FP, dan FN dari setiap kategori serangan siber, langkah selanjutnya adalah menghitung metrik evaluasi yang digunakan untuk mengetahui performa model XGBS-E2 dalam klasifikasi serangan siber pada IoT berdasarkan kategorinya yang secara rinci disajikan pada Tabel 5.6.

Tabel 5.6 Performa Hasil Pengujian Model XGBS-E2

Kelas	**Akurasi (%)**	**Presisi**	***Recall***	***F1-Score***	***Support***
BEN	99.71	1	1	1	48941
CAAD		1	1	1	1103
DoS		1	1	1	85879
ITT		1	0.99	1	134
NS		0.98	0.88	0.93	2915
PC3C		0.94	1	0.97	122
TBF		0.73	0.95	0.82	904

5.2.3 Pelatihan Model XGBS-E3

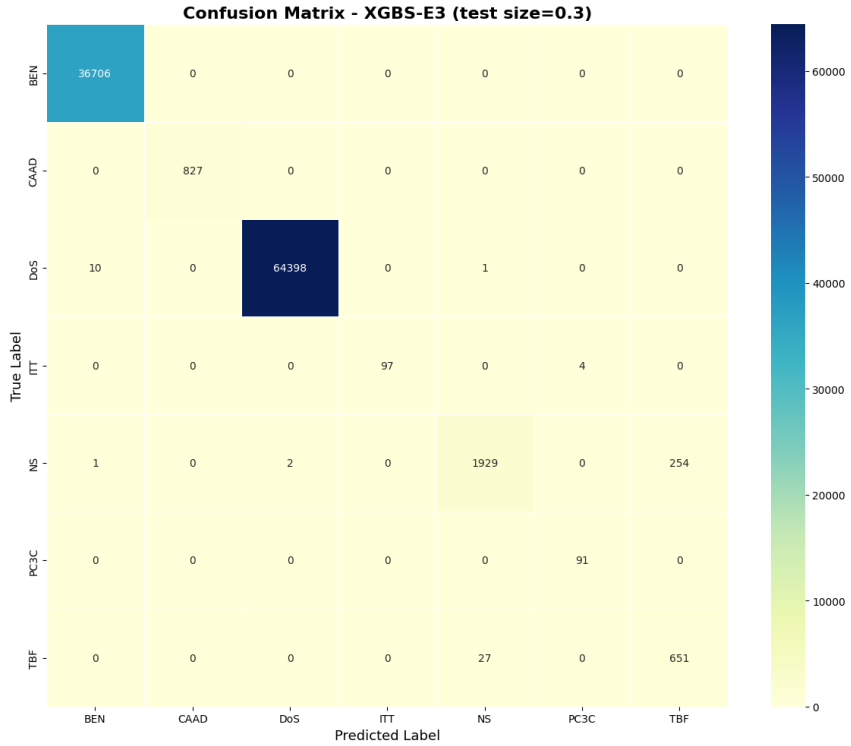
Proses eksperimen dari pelatihan model XGBS-E3 dalam melakukan klasifikasi serangan siber pada IoT digunakan proporsi data latih sebesar 70% atau sekitar 245.000 data dan data uji sebesar 30% atau sekitar 105.000 data. Setelah dilakukan proses SMOTE jumlah data *training* menjadi 1.202.304 data dengan masing-masing kelas sejumlah 150.288 data. Model dilatih dengan jumlah n_estimators yang telah ditentukan, yaitu sebanyak 200 pohon. Hasil pelatihan menunjukkan nilai *train loss* sekitar 0.024902 dan nilai *validation loss* sekitar 0.006036 sebagaimana ditampilkan pada Gambar 5.4.



Gambar 5.4 Grafik Nilai *Train* dan *Validation Loss* Model XGBS-E3

Gambar 5.4 menunjukkan nilai *train* dan *validation loss* pada proses pelatihan model XGBS-E3 yang terus mengalami penurunan hingga mencapai titik konvergensi pada n_estimators ke-200, sehingga model tersebut dianggap telah optimal. Selanjutnya, model optimal yang telah didapatkan, kemudian digunakan untuk melakukan klasifikasi serangan siber berdasarkan kategori serangan siber,

meliputi *Denial of Service* (DoS), *Benign* (BEN), *Network Scanning* (NS), CoAP *Amplification Attack Data* (CAAD), *Telnet Brute Forcing* (TBF), *Ingress Tool Transfer* (ITT), dan *Periodic Command and Control* (C&C) *Communication* (PC3C). Hasil dari klasifikasi tersebut disajikan dalam bentuk *confusion matrix* pada Gambar 4.18.



Gambar 4.18 Hasil Klasifikasi Model XGBS-E3 Terhadap Serangan Siber

Model XGBS-E3 memperlihatkan peningkatan konsistensi klasifikasi. Kelas BEN mencapai 36.706 data benar tanpa kesalahan, dan CAAD juga akurat sempurna dengan 827 data benar. Kelas DoS mencatat 64.398 data benar dan hanya 11 kesalahan kecil. ITT memiliki 97 data benar dan 4 salah ke PC3C, menunjukkan kemampuan model yang stabil dalam mengenali pola lalu lintas IoT yang normal. Namun, kelas NS dan TBF masih menjadi titik lemah utama. Kelas NS memiliki 1.929 data benar dan 257 salah klasifikasi, dengan 254 data salah ke TBF.

Sementara TBF memiliki 651 data benar tetapi menerima 27 data salah dari NS. Meskipun demikian, dibandingkan dengan dua model sebelumnya, jumlah kesalahan berkurang cukup signifikan. Hasil ini mengindikasikan bahwa proporsi data latih yang lebih besar membantu model dalam menurunkan tingkat kesalahan antar kelas minor, sekaligus menjaga akurasi tinggi pada kelas mayoritas. Nilai TP, TN, FP, dan FN dari hasil klasifikasi serangan siber berdasarkan kategori serangan siber secara rinci disajikan dalam Tabel 5.7.

Tabel 5.7 Nilai TP, TN, FP, dan FN Setiap Kategori pada Model XGBS-E3

Kelas	***True Positives* (TP)**	***True Negatives* (TN)**	***False Positives* (FP)**	***False Negatives* (FN)**
BEN	36706	68283	11	0
CAAD	827	104173	0	0
DoS	64398	40589	2	11
ITT	97	104899	0	4
NS	1929	102786	28	257
PC3C	91	104905	4	0
TBF	651	104068	254	27

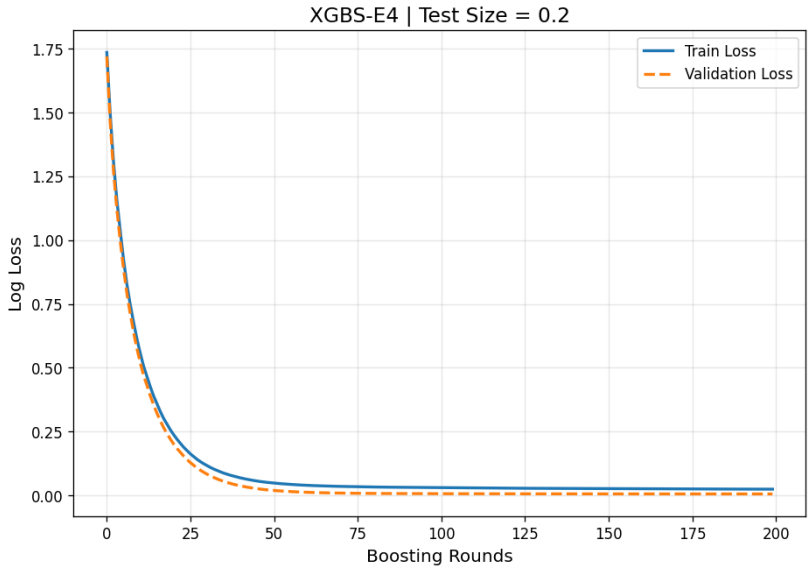
Setelah didapatkan nilai TP, TN, FP, dan FN dari setiap kategori serangan siber, langkah selanjutnya adalah menghitung metrik evaluasi. Metrik evaluasi digunakan untuk mengetahui performa model XGBS-E3 dalam klasifikasi serangan siber pada IoT. Metrik evaluasi berdasarkan kategori secara rinci disajikan pada Tabel 5.8.

Tabel 5.8 Performa Hasil Pengujian Model XGBS-E3

Kelas	**Akurasi (%)**	**Presisi**	***Recall***	***F1-Score***	***Support***
BEN	99.72	1	1	1	36706
CAAD		1	1	1	827
DoS		1	1	1	64409
ITT		1	0.96	0.98	101
NS		0.98	0.89	0.93	2186
PC3C		0.95	1	0.97	91
TBF		0.73	0.95	0.83	678

5.2.4 Pelatihan Model XGBS-E4

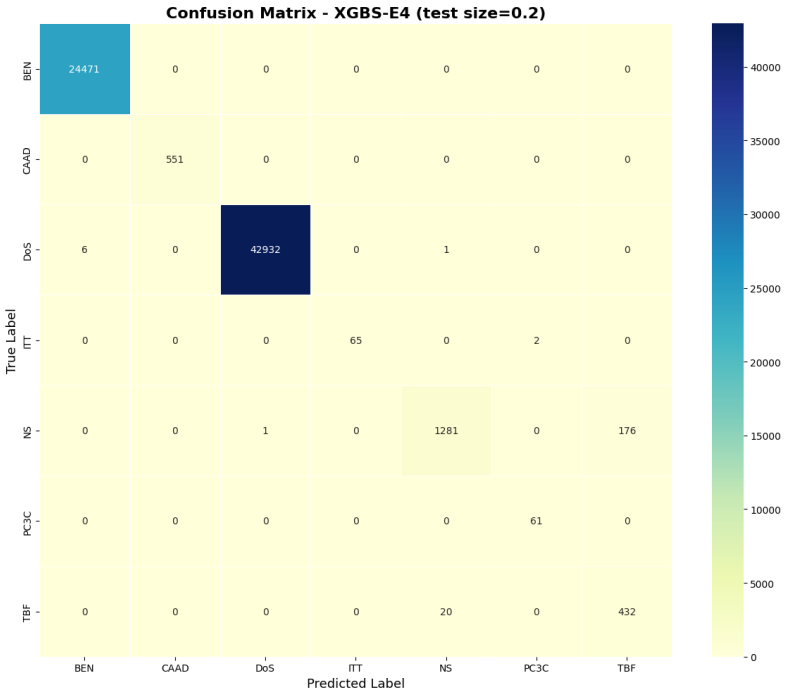
Proses eksperimen dari pelatihan model XGBS-E4 dalam melakukan klasifikasi serangan siber pada IoT digunakan proporsi data latih sebesar 80% atau sekitar 280.000 data dan data uji sebesar 20% atau sekitar 70.000 data. Setelah dilakukan proses SMOTE jumlah data *training* menjadi 1.374.064 data dengan masing-masing kelas sejumlah 171.758 data. Model dilatih dengan jumlah n_estimators yang telah ditentukan, yaitu sebanyak 200 pohon. Hasil pelatihan menunjukkan nilai *train loss* sekitar 0.024717 dan nilai *validation loss* sekitar 0.006049 sebagaimana ditampilkan pada Gambar 5.5.



Gambar 5.5 Grafik Nilai *Train* dan *Validation Loss* Model XGBS-E4

Gambar 5.5 menunjukkan nilai *train* dan *validation loss* pada proses pelatihan model XGBS-E4 yang terus mengalami penurunan hingga mencapai titik konvergensi pada n_estimators ke-200, sehingga model tersebut dianggap telah optimal. Selanjutnya, model optimal yang telah didapatkan, kemudian digunakan

untuk melakukan klasifikasi serangan siber berdasarkan kategori serangan siber. Hasil dari klasifikasi tersebut disajikan dalam bentuk *confusion matrix* pada Gambar 5.6.



Gambar 5.6 Hasil Klasifikasi Model XGBS-E4 Terhadap Serangan Siber

Model XGBS-E4 menampilkan hasil paling efisien dengan keseimbangan yang baik antara kompleksitas dan generalisasi. Semua kelas utama seperti BEN (24.471 benar), CAAD (551 benar), DoS (42.932 benar), dan ITT (65 benar) terklasifikasi dengan sangat baik. Hal ini menunjukkan bahwa model mampu mengenali pola serangan umum dengan presisi tinggi. Untuk kelas NS, model mencatat 1.281 data benar dan 177 salah, sebagian besar (176 data) salah diprediksi sebagai TBF. Sebaliknya, TBF memiliki 432 data benar dan menerima 20 data salah dari NS. Meski masih ada kesalahan silang antar dua kelas tersebut, nilainya semakin kecil dibanding model sebelumnya. XGBS-E4 menunjukkan bahwa rasio

data latih 80% menghasilkan model yang lebih stabil, efisien, dan akurat, terutama dalam mendeteksi pola serangan yang kompleks. Nilai TP, TN, FP, dan FN dari hasil klasifikasi serangan siber secara rinci disajikan dalam Tabel 5.9.

Tabel 5.9 Nilai TP, TN, FP, dan FN Setiap Kategori pada Model XGBS-E4

Kelas	***True Positives* (TP)**	***True Negatives* (TN)**	***False Positives* (FP)**	***False Negatives* (FN)**
BEN	24471	45523	6	0
CAAD	551	69449	0	0
DoS	42932	27060	1	7
ITT	65	69933	0	2
NS	1281	68521	21	177
PC3C	61	69937	2	0
TBF	432	69372	176	20

Setelah didapatkan nilai TP, TN, FP, dan FN dari setiap kategori serangan siber, langkah selanjutnya adalah menghitung metrik evaluasi yang digunakan untuk mengetahui performa model XGBS-E4 dalam klasifikasi serangan siber pada IoT berdasarkan kategorinya yang secara rinci disajikan pada Tabel 5.10.

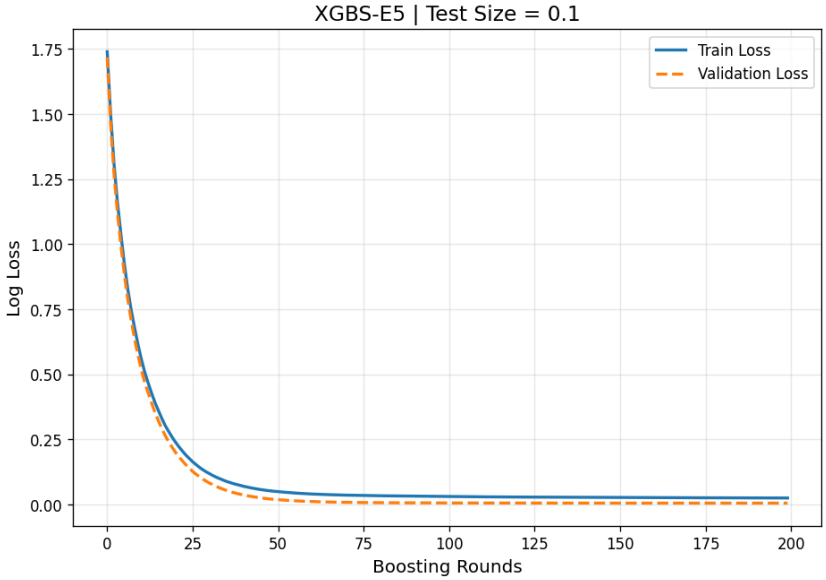
Tabel 5.10 Performa Hasil Pengujian Model XGBS-E4

Kelas	**Akurasi (%)**	**Presisi**	***Recall***	***F1-Score***	***Support***
BEN	99.71	1	1	1	24471
CAAD		1	1	1	551
DoS		1	1	1	42939
ITT		1	0.97	0.98	67
NS		0.98	0.88	0.93	1458
PC3C		0.97	1	0.98	61
TBF		0.72	0.94	0.82	452

5.2.5 Pelatihan Model XGBS-E5

Proses eksperimen dari pelatihan model XGBS-E5 dalam melakukan klasifikasi serangan siber pada IoT digunakan proporsi data latih sebesar 90% atau sekitar 315.000 data dan data uji sebesar 10% atau sekitar 35.000 data. Setelah

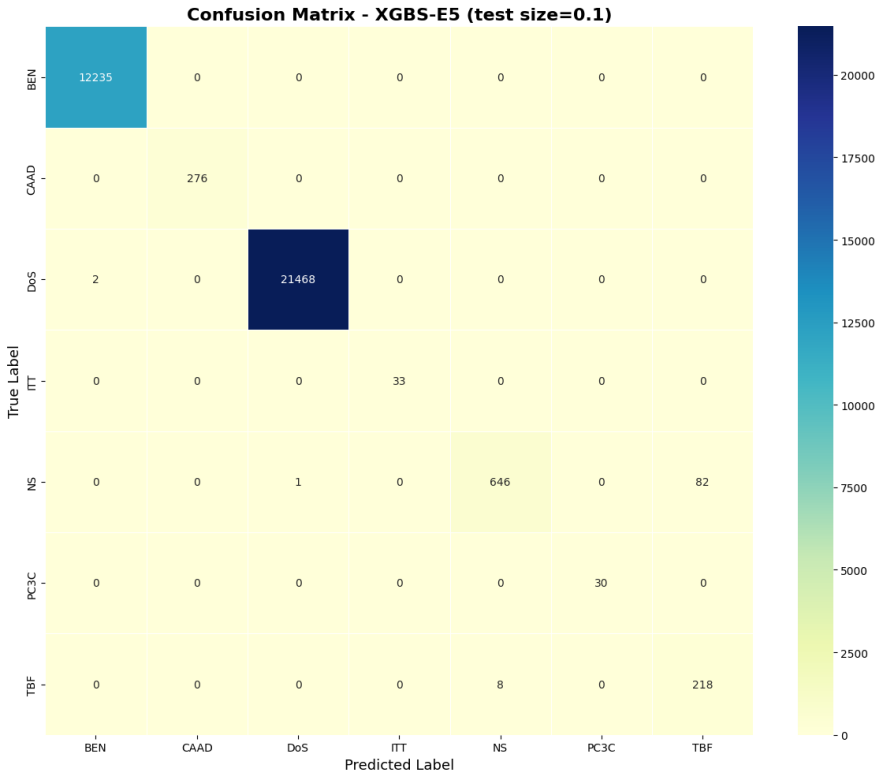
dilakukan proses SMOTE jumlah data *training* menjadi 1.545.816 data dengan masing-masing kelas sejumlah 193.227 data. Model dilatih dengan jumlah n_estimators yang telah ditentukan, yaitu sebanyak 200 pohon. Hasil pelatihan menunjukkan nilai *train loss* sekitar 0.025112 dan nilai *validation loss* sekitar 0.005427 sebagaimana ditampilkan pada Gambar 5.7.



Gambar 5.7 Grafik Nilai *Train* dan *Validation Loss* Model XGBS-E5

Gambar 5.7 menunjukkan nilai *train* dan *validation loss* pada proses pelatihan model XGBS-E5 yang terus mengalami penurunan hingga mencapai titik konvergensi pada n_estimators ke-200, sehingga model tersebut dianggap telah optimal. Selanjutnya, model optimal yang telah didapatkan, kemudian digunakan untuk melakukan klasifikasi serangan siber berdasarkan kategori serangan siber, meliputi *Denial of Servic*e (DoS), *Benign* (BEN), *Network Scanning* (NS), CoAP *Amplification Attack Data* (CAAD), *Telnet Brute Forcing* (TBF), *Ingress Tool Transfer* (ITT), dan *Periodic Command and Control* (C&C) *Communication*

(PC3C). Hasil dari klasifikasi tersebut disajikan dalam bentuk *confusion matrix* pada Gambar 5.8.



Gambar 5.8 Hasil Klasifikasi Model XGBS-E5 Terhadap Serangan Siber

Model XGBS-E5 memberikan hasil klasifikasi paling optimal di antara seluruh eksperimen. Kelas BEN mencapai 12.235 data benar tanpa kesalahan, CAAD menunjukkan 276 data benar sempurna, dan DoS menampilkan 21.468 data benar dengan hanya 2 salah ke BEN. Kelas ITT juga terprediksi sempurna dengan 33 data benar. Kelas NS menunjukkan 646 data benar dengan 83 salah klasifikasi, di mana 82 di antaranya salah diprediksi sebagai TBF. Kelas TBF memiliki 218 data benar, tetapi menerima 8 data salah dari NS. Meskipun pola ambiguitas NS dan TBF masih muncul, jumlah kesalahan sangat kecil. Dengan proporsi data latih sebesar 90%, model ini memiliki akurasi di atas 99,9%, menjadikannya varian

paling kuat dan konsisten dalam mendeteksi berbagai jenis serangan siber pada jaringan IoT. Nilai TP, TN, FP, dan FN dari hasil klasifikasi serangan siber secara rinci disajikan dalam Tabel 5.11.

Tabel 5.11 Nilai TP, TN, FP, dan FN Setiap Kategori pada Model XGBS-E5

Kelas	***True Positives (TP)***	***True Negatives (TN)***	***False Positives (FP)***	***False Negatives (FN)***
BEN	12235	22763	2	0
CAAD	276	34724	0	0
DoS	21468	13529	1	2
ITT	33	34967	0	0
NS	646	34263	8	83
PC3C	30	34970	0	0
TBF	218	34692	82	8

Setelah didapatkan nilai TP, TN, FP, dan FN dari setiap kategori serangan siber, langkah selanjutnya adalah menghitung metrik evaluasi yang digunakan untuk mengetahui performa model XGBS-E5. Hasil dari metrik evaluasi berdasarkan kategori secara rinci disajikan pada Tabel 5.12.

Tabel 5.12 Performa Hasil Pengujian Model XGBS-E5

Kelas	**Akurasi (%)**	**Presisi**	***Recall***	***F1-Score***	***Support***
BEN	99.73	1	1	1	12235
CAAD		1	1	1	276
DoS		1	1	1	21470
ITT		1	1	1	33
NS		0.98	0.89	0.93	729
PC3C		1	1	1	30
TBF		0.73	0.95	0.83	226

5.3 Hasil Ujicoba Model XGBS

Setelah proses pelatihan selesai, dilakukan proses ujicoba pada masing-masing model XGBoost dengan SMOTE menggunakan data uji guna menilai kinerjanya dalam mengklasifikasikan serangan siber berdasarkan kategorinya.

Selanjutnya, dilakukan proses evaluasi performa model menggunakan nilai akurasi, presisi, *recall*, dan *F1-score*, dimana masing-masing nilai memberikan perspektif berbeda terhadap performa model.

Tabel 5.13 Hasil Ujicoba Klasifikasi Serangan Siber pada Model XGBS

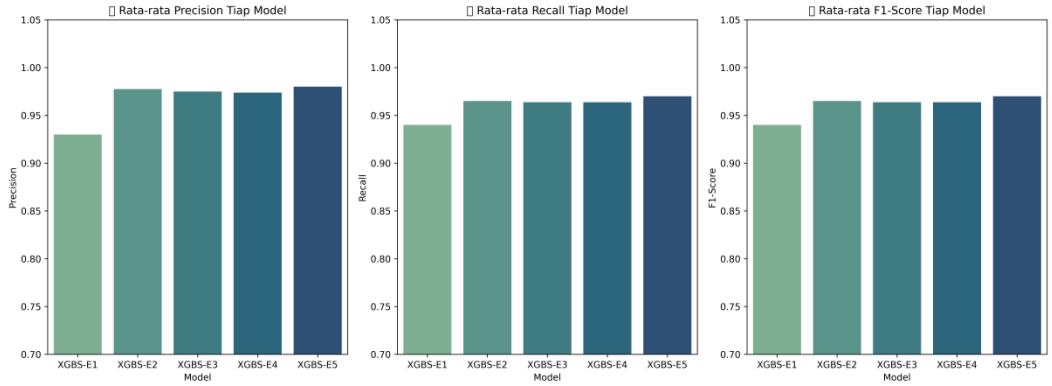
Model	**Kelas**	**Akurasi (%)**	**Presisi**	***Recall***	***F1-Score***	***Support***
XGBS-E1	BEN	99.71	1	1	1	61176
	CAAD		1	1	1	1378
	DoS		1	1	1	107349
	ITT		1	0.97	0.98	168
	NS		0.97	0.9	0.93	3644
	PC3C		0.99	0.99	0.99	153
	TBF		0.74	0.91	0.82	1129
XGBS-E2	BEN	99.71	1	1	1	48941
	CAAD		1	1	1	1103
	DoS		1	1	1	85879
	ITT		1	0.99	1	134
	NS		0.98	0.88	0.93	2915
	PC3C		0.94	1	0.97	122
	TBF		0.73	0.95	0.82	904
XGBS-E3	BEN	99.72	1	1	1	36706
	CAAD		1	1	1	827
	DoS		1	1	1	64409
	ITT		1	0.96	0.98	101
	NS		0.98	0.89	0.93	2186
	PC3C		0.95	1	0.97	91
	TBF		0.73	0.95	0.83	678
XGBS-E4	BEN	99.71	1	1	1	24471
	CAAD		1	1	1	551
	DoS		1	1	1	42939
	ITT		1	0.97	0.98	67
	NS		0.98	0.88	0.93	1458
	PC3C		0.97	1	0.98	61
	TBF		0.72	0.94	0.82	452
XGBS-E5	BEN	99.73	1	1	1	12235
	CAAD		1	1	1	276
	DoS		1	1	1	21470
	ITT		1	1	1	33
	NS		0.98	0.89	0.93	729
	PC3C		1	1	1	30
	TBF		0.73	0.95	0.83	226

Tabel 5.13 menunjukkan bahwa model XGBS-E1 memiliki performa klasifikasi yang sangat baik pada hampir seluruh kelas serangan siber. Model ini mencatat akurasi 99,71% secara keseluruhan dengan nilai presisi, *recall*, dan *F1-score* 1 pada kelas mayoritas seperti BEN, CAAD, DoS, dan ITT. Hal ini menandakan bahwa model mampu mengenali pola data normal maupun jenis serangan utama dengan tingkat ketepatan yang sangat tinggi. Namun, performa menurun pada kelas TBF (*Telnet Brute Force*) dengan nilai akurasi 0,74 dan *F1-score* 0,82, serta pada kelas NS (*Network Scan*) dengan *recall* 0,90. Kedua kelas ini menunjukkan tingkat kesalahan lebih tinggi akibat kemiripan karakteristik lalu lintas jaringan, yang sering menyebabkan ambiguitas klasifikasi.

Model XGBS-E2, yang menggunakan rasio data latih lebih besar (60:40), tetap mempertahankan konsistensi performa tinggi dengan akurasi rata-rata 99,71%. Kelas BEN, CAAD, DoS, dan ITT kembali mencapai nilai *F1-score* 1,00, menandakan kestabilan model dalam mendeteksi pola serangan utama. Performa kelas NS sedikit meningkat dengan akurasi 0,98 dan *F1-score* 0,93, menunjukkan bahwa penambahan data latih membantu model memahami variasi pola lalu lintas jaringan yang lebih kompleks. Sementara itu, kelas TBF masih menunjukkan tantangan dengan akurasi 0,73, meskipun presisi meningkat menjadi 0,95, yang menandakan bahwa kesalahan prediksi semakin sedikit dibanding model sebelumnya.

Pada model XGBS-E3, hasil klasifikasi menunjukkan kecenderungan peningkatan performa yang stabil seiring bertambahnya proporsi data latih (70%). Model ini mencapai akurasi global 99,72% dengan nilai presisi dan *recall* sebesar

1,00 pada kelas utama seperti BEN, CAAD, dan DoS. Kelas minor seperti PC3C juga menunjukkan akurasi sempurna, menandakan bahwa model mulai mampu mempelajari distribusi serangan berfrekuensi rendah. Namun, kelas NS dan TBF tetap menjadi titik lemah dengan nilai *recall* 0,89 dan 0,95, serta *F1-score* 0,93 dan 0,83. Hal ini mengindikasikan bahwa meskipun model semakin matang, tumpang tindih fitur antar kelas minor masih menjadi sumber kesalahan yang belum bisa diatasi dengan baik.



Gambar 5.9 Rata-Rata Performa Model XGBS

Gambar 5.9 menunjukkan bahwa model XGBS-E4, yang menggunakan 80% data latih dan 20% data uji, memperlihatkan performa yang solid dengan akurasi total 99,71%. Semua kelas utama (BEN, CAAD, DoS, dan ITT) kembali memperoleh nilai *F1-score* 1,00, mencerminkan konsistensi model dalam mengidentifikasi pola serangan utama. Peningkatan signifikan terlihat pada kelas PC3C dengan akurasi 0,97 dan *F1-score* 0,98, menunjukkan efektivitas pembelajaran pada data minor. Sementara itu, kelas NS tetap berada di kisaran akurasi 0,98 dan *F1-score* 0,93, sedangkan TBF mengalami sedikit penurunan

menjadi akurasi 0,72 dan *F1-score* 0,82, yang menunjukkan sensitivitas model terhadap perubahan distribusi data uji pada serangan yang jarang terjadi.

Model XGBS-E5 yang dilatih dengan proporsi data terbesar (90% latih, 10% uji) memberikan performa paling optimal di antara seluruh model. Kelas BEN, CAAD, DoS, ITT, dan PC3C seluruhnya mencapai presisi, *recall*, dan *F1-score* sebesar 1,00, menandakan bahwa model telah mampu melakukan generalisasi sempurna terhadap pola utama. Pada kelas NS, model mencatat akurasi 0,98 dan *F1-score* 0,93, sedangkan TBF memperoleh hasil terbaik di antara seluruh eksperimen dengan akurasi 0,73, presisi 0,95, dan *F1-score* 0,83. Temuan ini memperkuat bahwa semakin besar porsi data pelatihan, semakin baik kemampuan model XGBoost dalam membedakan variasi serangan siber dengan kompleksitas tinggi.

Secara keseluruhan, kelima model XGBS menunjukkan tren peningkatan performa seiring dengan pertambahan proporsi data latih. Semua model unggul dalam mengenali kelas mayoritas (BEN, DoS, dan CAAD) dengan nilai metrik sempurna, sementara kesalahan klasifikasi masih terkonsentrasi pada kelas minoritas seperti NS dan TBF. Penerapan SMOTE (*Synthetic Minority Over-sampling Technique*) terbukti efektif dalam mengurangi ketidakseimbangan data, tetapi perbaikan lanjutan diperlukan untuk menangani kemiripan fitur antar serangan minor. Dengan demikian, model XGBS, terutama XGBS-E5, dapat dianggap sebagai konfigurasi paling optimal untuk mendeteksi berbagai jenis serangan siber pada jaringan IoT secara akurat dan efisien.

Tabel 5.14 Performa Model XGBS

Model	Komposisi Data	Nilai Akurasi
XGBS-E1	50:50	99.71%
XGBS-E2	60:40	99.71%
XGBS-E3	70:30	99.72%
XGBS-E4	80:20	99.71%
XGBS-E5	**90:10**	**99.73%**

Tabel 5.14 menunjukkan bahwa model XGBS-E1 dengan komposisi data 50:50 memperoleh nilai akurasi sebesar 99,71%, yang menandakan bahwa penerapan metode SMOTE mampu meningkatkan kemampuan model XGBoost dalam mengenali pola serangan secara lebih seimbang, meskipun pada tahap awal proporsi data latih belum terlalu besar. Selanjutnya, model XGBS-E2 (60:40) menunjukkan akurasi yang tetap stabil sebesar 99,71%, mengindikasikan bahwa peningkatan jumlah data pelatihan belum memberikan dampak signifikan terhadap akurasi global, namun menjaga konsistensi performa klasifikasi di seluruh kelas.

Selain itu, model XGBS-E3 (70:30) memperlihatkan peningkatan kecil dengan akurasi 99,72%, yang menunjukkan bahwa seiring bertambahnya proporsi data latih, model menjadi semakin adaptif terhadap variasi data hasil *oversampling* dari SMOTE. Sementara itu, model XGBS-E4 (80:20) mencatatkan akurasi sekitar 99,71%, menandakan kestabilan performa meskipun terdapat fluktuasi kecil akibat distribusi data yang semakin padat. Adapun model XGBS-E5 (90:10) menunjukkan hasil tertinggi dengan akurasi sekitar 99,73%, menjadikannya sebagai model terbaik di antara seluruh konfigurasi karena mampu memanfaatkan proporsi data latih yang besar untuk meningkatkan kemampuan generalisasi tanpa kehilangan akurasi pada kelas minoritas.

Secara menyeluruh, hasil pengujian memperlihatkan bahwa seluruh model XGBS memiliki performa yang konsisten dengan tingkat akurasi di atas 99,7%, menunjukkan efektivitas penerapan SMOTE dalam menyeimbangkan distribusi data pelatihan. Di antara seluruh konfigurasi, model XGBS-E5 dengan nilai akurasi sekitar 99,73% menonjol sebagai varian paling optimal, karena mampu mempertahankan keseimbangan ideal antara proporsi data latih yang besar, hasil oversampling yang efisien, serta kemampuan generalisasi yang kuat terhadap seluruh jenis serangan siber.

Tabel 5.15 Nilai TP, TN, FP, dan FN Model XGBS-E5

Kelas	***True Positives (TP)***	***True Negatives (TN)***	***False Positives (FP)***	***False Negatives (FN)***
BEN	12235	22763	2	0
CAAD	276	34724	0	0
DoS	21468	13529	1	2
ITT	33	34967	0	0
NS	646	34263	8	83
PC3C	30	34970	0	0
TBF	218	34692	82	8

Tabel 5.15 menunjukkan hasil pengujian model XGBS-E5 terhadap setiap kelas serangan berdasarkan nilai *True Positives* (TP), *True Negatives* (TN), *False Positives* (FP), dan *False Negatives* (FN). Hasil tersebut memperlihatkan bahwa model XGBS-E5, yang dibangun dengan pendekatan SMOTE dan proporsi data latih 90:10, mampu menghasilkan performa klasifikasi yang sangat baik dengan kesalahan prediksi yang minimal di seluruh kelas serangan maupun lalu lintas normal.

Kelas BEN (*Benign*) memperoleh nilai TP = 12.235 dan TN = 22.763, dengan FP = 2 dan FN = 0, menunjukkan bahwa model dapat mengidentifikasi lalu

lintas normal dengan tingkat akurasi hampir sempurna. Kesalahan klasifikasi yang sangat kecil menandakan kemampuan model dalam membedakan antara aktivitas normal dan anomali jaringan secara efisien. Pada kelas CAAD (*Command and Control Attack Detection*), model mencatat TP = 276 dan TN = 34.724, tanpa adanya FP maupun FN (0). Hal ini menunjukkan deteksi sempurna terhadap serangan jenis CAAD, yang berarti model dapat mengenali pola serangan berbasis *command and control* dengan presisi 100%. Sementara itu, kelas DoS (*Denial of Service*) memiliki TP = 21.468, TN = 13.529, dengan FP = 1 dan FN = 2. Nilai ini menunjukkan bahwa model mampu mengenali mayoritas besar dari serangan DoS dengan tingkat kesalahan yang sangat rendah, mencerminkan keunggulan XGBoost dalam menangani data hasil *oversampling* tanpa *overfitting*.

Selanjutnya, untuk kelas ITT (*Ingress Tool Transfer*), model menghasilkan TP = 33 dan TN = 34.967, dengan FP = 0 dan FN = 0, yang berarti model dapat mengidentifikasi seluruh data serangan dan non-serangan dengan akurasi sempurna pada kategori ini. Pada kelas NS (*Network Scan*), nilai TP = 646, TN = 34.263, FP = 8, dan FN = 83. Meskipun secara umum akurasinya tetap tinggi, nilai FN yang relatif lebih besar dibandingkan kelas lain menunjukkan bahwa model masih memiliki sedikit keterbatasan dalam mengenali seluruh variasi pola serangan scanning yang mirip dengan trafik normal. Kelas PC3C (*Communication*) menunjukkan performa optimal dengan TP = 30, TN = 34.970, serta FP = 0 dan FN = 0, mengindikasikan deteksi sempurna tanpa kesalahan prediksi. Sedangkan pada kelas TBF (*Telnet Brute Forcing*), diperoleh TP = 218, TN = 34.692, FP = 82, dan FN = 8. Nilai FP yang relatif lebih tinggi dibandingkan kelas lain menunjukkan

bahwa sebagian kecil aktivitas non-serangan masih keliru terklasifikasi sebagai serangan *brute force*, namun secara keseluruhan performa tetap sangat baik.

Secara keseluruhan, model XGBS-E5 menunjukkan kemampuan deteksi yang stabil dan akurat di semua kelas serangan, dengan nilai TP dan TN yang tinggi serta FP dan FN yang rendah. Penerapan SMOTE terbukti efektif dalam menyeimbangkan data dan meningkatkan kemampuan model dalam mengenali kelas minoritas tanpa menurunkan akurasi pada kelas mayoritas. Dengan demikian, XGBS-E5 dapat dikategorikan sebagai model yang dapat digunakan untuk mendeteksi berbagai jenis serangan siber pada jaringan IoT dengan performa yang sangat baik.

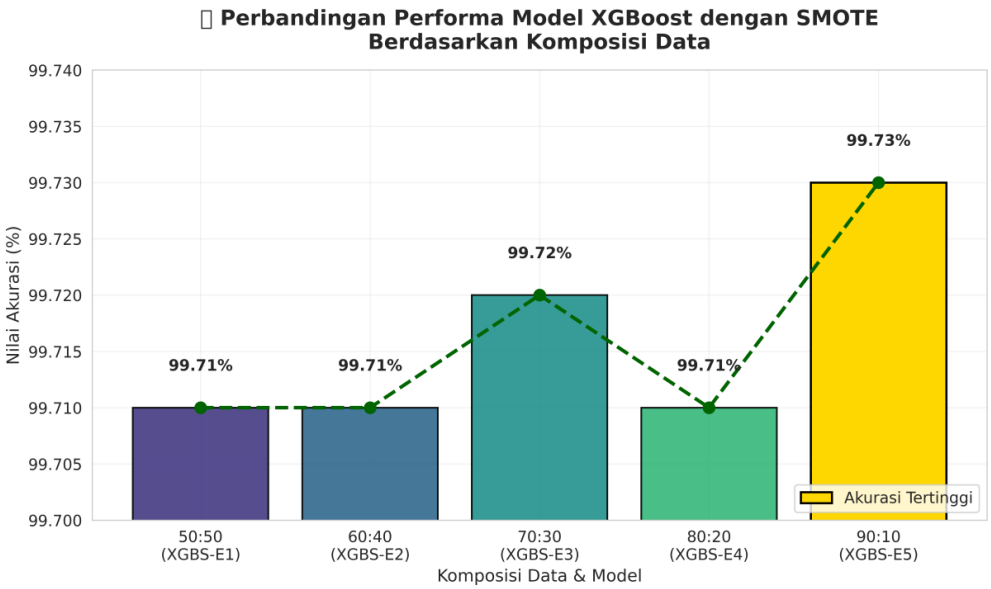
5.4 Kesimpulan

Berdasarkan hasil evaluasi menyeluruh terhadap model XGBoost dengan SMOTE dalam proses optimasi model untuk mengklasifikasikan serangan siber berdasarkan kategorinya, dapat disimpulkan bahwa performa model sangat dipengaruhi oleh komposisi data latih dan data uji yang digunakan setelah dilakukan penyeimbangan data. Dari kelima skenario pengujian, yaitu XGBS-E1 (50:50), XGBS-E2 (60:40), XGBS-E3 (70:30), XGBS-E4 (80:20), dan XGBS-E5 (90:10), diperoleh bahwa model XGBS-E5 menunjukkan performa paling optimal dengan nilai akurasi mencapai 99,73%, diikuti oleh XGBS-E3 sebesar 99,72%, diikuti oleh model XGBS-E1, XGBS-E2, dan XGBS-E4 dengan nilai akurasi yang sama yaitu sekitar 99,71%. Hasil ini memperlihatkan bahwa komposisi data latih sebesar 90% dan data uji sebesar 10% merupakan konfigurasi ideal bagi model XGBoost yang telah diterapkan metode SMOTE, karena memberikan

keseimbangan terbaik antara pembelajaran pola dan kemampuan generalisasi terhadap data baru.

Evaluasi berdasarkan presisi, *recall*, dan *F1-score* juga menunjukkan hasil yang sangat tinggi pada seluruh kelas serangan. Hampir semua kelas memperoleh nilai presisi dan *recall* ≥ 0.999, menandakan bahwa model mampu mengidentifikasi serangan maupun trafik normal dengan sangat akurat. Peningkatan performa ini dipengaruhi oleh penerapan teknik SMOTE yang efektif menyeimbangkan distribusi kelas minoritas dan mayoritas, sehingga mengurangi bias model terhadap kelas dominan. Secara khusus, kelas serangan yang sebelumnya sulit dikenali seperti CAAD dan PC3C mengalami peningkatan signifikan dalam nilai *recall* dan *F1-score* setelah dilakukan *balancing*.

Analisis lanjutan terhadap nilai *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN) pada model terbaik, yaitu XGBS-E5, menunjukkan bahwa model mampu menghasilkan TP dan TN sangat tinggi, disertai FP dan FN yang sangat rendah di seluruh kelas. Hal ini membuktikan bahwa model tidak hanya mampu mengenali pola serangan dengan sangat akurat, tetapi juga memiliki kemampuan deteksi yang konsisten pada semua kelas serangan, termasuk jenis-jenis serangan yang memiliki jumlah data relatif sedikit.



Gambar 5.10 Perbandingan Performa Model XGBS

Gambar 5.10 menunjukkan bahwa penerapan SMOTE pada model XGBoost memberikan pengaruh terhadap peningkatan performa klasifikasi, terutama pada kelas dengan distribusi data yang tidak seimbang. Teknik *balancing* ini memungkinkan model untuk mempelajari pola dari kelas minoritas secara lebih representatif, sehingga meningkatkan akurasi dan stabilitas model tanpa menimbulkan overfitting. Secara keseluruhan, hasil penelitian ini menegaskan bahwa model XGBS-E5 merupakan model terbaik dalam mendeteksi serangan siber berbasis IoT dengan performa yang stabil. Kombinasi algoritma XGBoost dengan pra-pemrosesan SMOTE terbukti menjadi pendekatan yang efektif dalam menghadapi ketidakseimbangan data pada skenario klasifikasi serangan siber.

BAB VI
MODEL XGBOOST DENGAN PRA-PEMROSESAN SMOTE DAN PCA

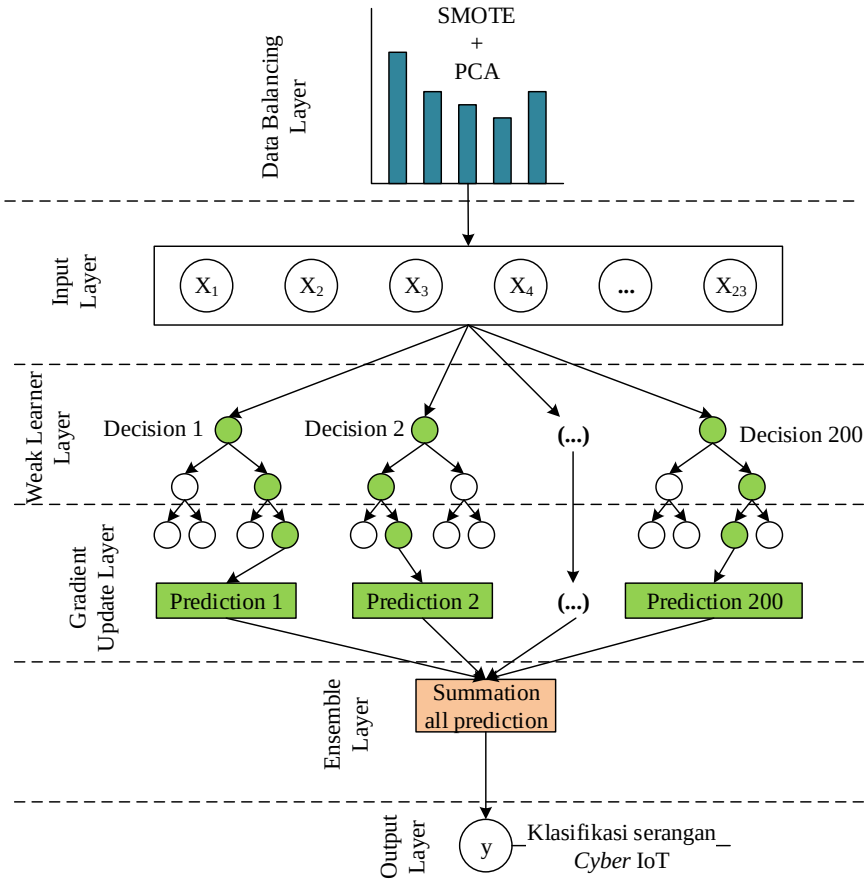
6.1 Desain Model XGBSP

Bagian model XGBSP dipilih dengan menggabungkan kemampuan boosting pada XGBoost dalam memperbaiki kesalahan model secara bertahap, SMOTE (*Synthetic Minority Over-sampling Technique*) dalam menyeimbangkan distribusi kelas data, dan PCA (*Principal Component Analysis*) dalam mereduksi dimensi data untuk menghilangkan redundansi fitur. Integrasi ketiga pendekatan ini diharapkan mampu meningkatkan akurasi prediksi dan efisiensi komputasi dalam mendeteksi pola serangan siber yang kompleks dan tidak seimbang. Gambaran lebih rinci mengenai rancangan arsitektur model XGBoost dengan SMOTE dan PCA dapat dilihat pada Gambar 6.1.

Proses *training* model XGBoost menggunakan SMOTE dan PCA dimulai dengan memanggil data *training* yang berisi berbagai fitur serangan siber yang telah melalui tahap pra-pemrosesan awal, seperti pembersihan dan transformasi nilai. Tahapan berikutnya adalah penerapan SMOTE, yaitu teknik *oversampling* yang menghasilkan sampel baru untuk kelas minoritas berdasarkan kemiripan antar-tetangga. Tujuan dari tahap ini adalah menyeimbangkan distribusi kelas agar model tidak bias terhadap kelas mayoritas serta mampu mengenali variasi pola serangan dari kelas minoritas dengan lebih proporsional.

Setelah distribusi kelas seimbang, dilakukan tahap reduksi dimensi menggunakan PCA untuk mengekstraksi fitur-fitur utama yang paling berpengaruh terhadap variasi data. Proses PCA ini bertujuan menurunkan kompleksitas dataset

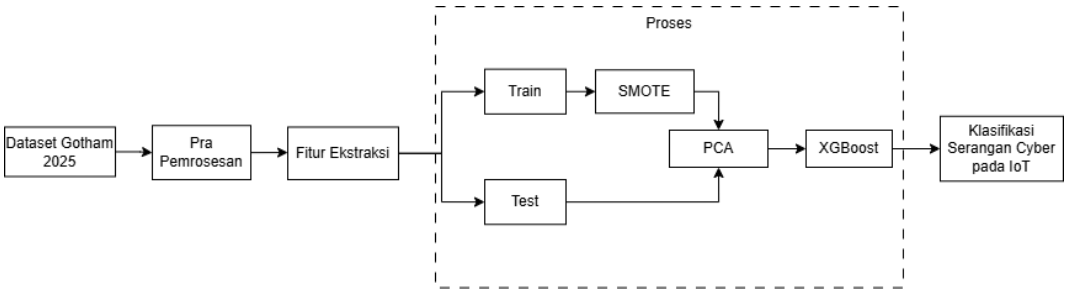
dengan cara mempertahankan komponen-komponen yang memiliki nilai varian paling tinggi, sehingga model dapat beroperasi dengan lebih ringan namun tetap mempertahankan informasi penting. Selanjutnya, hasil transformasi dari PCA digunakan sebagai input untuk pelatihan model XGBoost. Pada tahap ini, parameter utama ditetapkan berdasarkan hasil ujicoba parameter terbaik pada model XGBoost *non-balancing*.



Gambar 6.1 Arsitektur Model XGBSP

Proses pelatihan dilakukan secara bertahap pada lima skenario berbeda (XGBSP-E1 hingga XGBSP-E5), masing-masing dengan komposisi data latih dan uji yang bervariasi. Pendekatan bertahap ini bertujuan mengevaluasi pengaruh proporsi data terhadap performa model setelah melalui tahapan *balancing*

(SMOTE) dan reduksi dimensi (PCA). Hal ini dilakukan agar penelitian ini tidak hanya menilai akurasi model secara numerik, tetapi juga kestabilannya dalam berbagai konfigurasi data. Hasil prediksi dari seluruh pohon keputusan kemudian digabungkan menggunakan mekanisme *weighted boosting*, di mana bobot setiap pohon ditentukan berdasarkan kontribusinya dalam menurunkan *error* keseluruhan. Proses *training* berlanjut hingga seluruh 200 pohon terbentuk atau model mencapai titik konvergensi pada kestabilan performa. Setelah proses pelatihan selesai, model XGBoost menggunakan SMOTE dan PCA yang telah terlatih akan menyimpan struktur pohon, bobot *boosting*, serta parameter hasil tuning untuk digunakan pada tahap pengujian (*testing*).



Gambar 6.2 Desain Sistem Model XGBSP

Desain sistem dari proses pelatihan dan penerapan XGBoost menggunakan SMOTE dan PCA ditunjukkan pada Gambar 6.2, yang menggambarkan kerja alur sistem lengkap mulai dari pra-pemrosesan data, *balancing* kelas menggunakan SMOTE, reduksi dimensi menggunakan PCA, pelatihan model boosting bertingkat XGBoost, hingga tahap evaluasi performa model secara menyeluruh.

6.2 Implementasi Model XGBSP

Model XGBoost dengan pra-premrosesan menggunakan SMOTE dan PCA dirancang dengan konfigurasi parameter yang terintegrasi untuk menghasilkan performa klasifikasi optimal terhadap berbagai jenis serangan siber pada lingkungan *Internet of Things* (IoT). Model ini mengombinasikan kemampuan XGBoost sebagai algoritma boosting adaptif, SMOTE sebagai teknik balancing data sintetis, dan PCA sebagai metode reduksi dimensi untuk mengoptimalkan representasi fitur dan mempercepat proses pelatihan.

Parameter optimal yang didapatkan yaitu n_estimators = 200 menentukan jumlah pohon keputusan (*decision tree*) yang dibangun secara bertahap untuk memperkuat hasil klasifikasi. Nilai learning_rate = 0.1 berfungsi mengontrol besarnya kontribusi setiap pohon baru terhadap model akhir agar proses pembelajaran berjalan stabil dan tidak terlalu cepat mengalami overfitting. Selanjutnya, max_depth = 6 digunakan untuk mengendalikan kompleksitas setiap pohon agar model mampu menangkap pola non-linear tanpa kehilangan efisiensi komputasi. Parameter subsample = 0.8 dan colsample_bytree = 0.8 diterapkan untuk meningkatkan kemampuan generalisasi model dengan memilih sebagian data dan fitur secara acak pada setiap iterasi pembentukan pohon. Sementara itu, eval_metric = 'mlogloss' digunakan sebagai metrik utama untuk mengevaluasi kualitas prediksi probabilistik multi-kelas, sedangkan random_state = 42 memastikan hasil eksperimen dapat direproduksi secara konsisten. Tabel 6.1 merupakan rincian parameter model yang digunakan pada model XGBSP.

Tabel 6.1 Parameter Model XGBSP

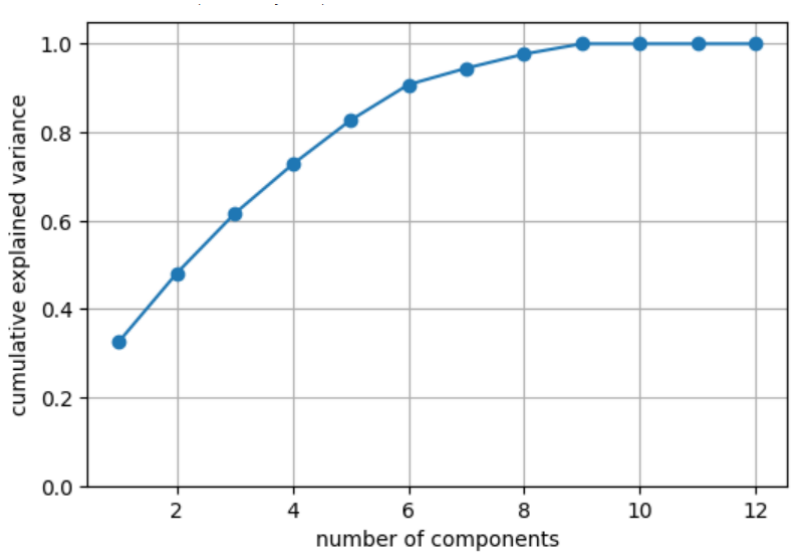
Parameter	Nilai	Keterangan	Penjelasan
XGBoost			
n_estimators	200	Jumlah Pohon	Menentukan jumlah pohon keputusan yang akan dibangun dalam model.
learning_rate	0.1	Laju Pembelajaran	Mengontrol seberapa besar kontribusi setiap pohon baru terhadap model akhir.
max_depth	6	Kedalaman Maksimum Pohon	Menentukan kedalaman maksimum setiap pohon keputusan.
subsample	0.8	Proporsi Subset Data	Mengatur persentase data pelatihan yang digunakan untuk membangun setiap pohon.
colsample_bytree	0.8	Proporsi Fitur per Pohon	Menentukan proporsi fitur (kolom) yang akan digunakan dalam pembangunan setiap pohon.
random_state	42	Pengaturan Angka Acak	Menetapkan *seed* acak untuk memastikan hasil eksperimen dapat direproduksi secara konsisten setiap kali model dijalankan.
use_label_encoder	FALSE	Penggunaan Label *Encoder* Internal	Menonaktifkan *encoder* label bawaan XGBoost agar tidak terjadi konflik dengan proses encoding yang telah dilakukan sebelumnya dalam tahap *preprocessing*.
eval_metric	*'mlogloss'*	Metrik Evaluasi	Menggunakan *multiclass* log loss sebagai metrik evaluasi utama untuk mengukur seberapa baik model

Parameter	Nilai	Keterangan	Penjelasan
			memprediksi probabilitas kelas yang benar pada data multi-kelas.
SMOTE			
sampling_strategy	*'auto'* (*default*)	Strategi Sampling	Menentukan rasio antara kelas minoritas dan mayoritas.
random_state	42	Pengaturan Angka Acak	Menjamin hasil proses oversampling dapat direproduksi secara konsisten setiap kali kode dijalankan.
k_neighbors	max(1, min_class_count - 1)	Jumlah Tetangga Terdekat	Menentukan jumlah tetangga terdekat yang digunakan untuk membangkitkan sampel sintetis baru.
n_jobs	None (default)	Jumlah CPU yang Digunakan	Menentukan berapa banyak core CPU yang digunakan untuk menjalankan proses SMOTE secara paralel.
categorical_features	None (default)	Fitur Kategorikal	Menentukan apakah ada fitur kategorikal yang harus ditangani secara khusus.
PCA			
n_components	min(10, n_features)	Jumlah Komponen Utama	Menentukan jumlah komponen utama (*principal components*) yang akan digunakan.
random_state	42	Pengaturan Angka Acak	Menjamin konsistensi hasil dekomposisi PCA pada setiap eksekusi agar hasil

Parameter	Nilai	Keterangan	Penjelasan
			eksperimen dapat direproduksi.
X_train_scaled	Data hasil standarisasi latih	Input Data PCA (Pelatihan)	Merupakan data latih yang telah dinormalisasi dan akan digunakan untuk membentuk komponen utama.
X_test_scaled	Data hasil standarisasi uji	Input Data PCA (Pengujian)	Data uji yang telah dinormalisasi dan ditransformasikan menggunakan model PCA yang sama dengan data latih.
fit_transform()	Fungsi transformasi pelatihan	Proses Pelatihan PCA	Melatih model PCA pada data latih sekaligus mentransformasikannya ke ruang komponen utama.
transform()	Fungsi transformasi pengujian	Proses Transformasi Data Uji	Menerapkan model PCA yang telah dilatih pada data uji agar memiliki representasi fitur yang sama dengan data latih.

Parameter utama SMOTE meliputi sampling_strategy = 'auto' untuk menyeimbangkan semua kelas secara otomatis, random_state = 42 untuk menjaga konsistensi hasil replikasi, serta k_neighbors = max(1, min_class_count - 1) yang secara dinamis menyesuaikan jumlah tetangga terdekat berdasarkan kelas dengan jumlah sampel paling sedikit. Selain itu, n_jobs = None digunakan agar proses oversampling dijalankan pada konfigurasi CPU default, sementara

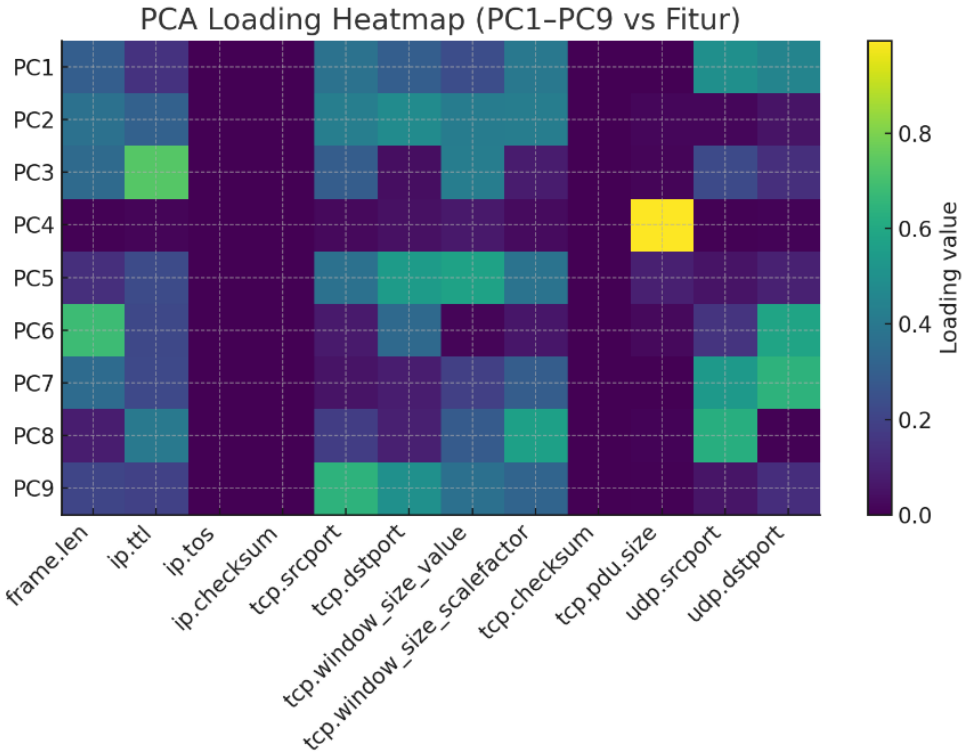
categorical_features = None menunjukkan bahwa seluruh fitur yang digunakan bersifat numerik karena telah melalui tahap encoding sebelumnya.



Gambar 6.3 Grafik *Cumulative Explained Variance*

Gambar 6.3 menunjukkan grafik *cumulative explained variance* yang merepresentasikan proporsi variansi data yang dapat dijelaskan oleh sejumlah komponen PCA. Berdasarkan grafik tersebut, komponen pertama hanya mampu menjelaskan sekitar 33% variansi data. Nilai ini meningkat signifikan hingga komponen ke-3 dan mencapai lebih dari 60%. Peningkatan variansi terus terjadi hingga komponen ke-6 dengan capaian sekitar 90%. Setelah komponen ke-8, kurva mulai melandai (elbow point) yang mengindikasikan bahwa penambahan komponen selanjutnya tidak lagi meningkatkan variansi secara signifikan. Pada komponen ke-10 hingga komponen ke-12, nilai *cumulative explained variance* mencapai hampir 100%, yang berarti seluruh informasi penting dari data sudah terwakili.

Berdasarkan hasil tersebut, penelitian ini menetapkan penggunaan 9 komponen PCA sebagai jumlah optimal. Pemilihan ini dilakukan untuk mempertahankan informasi data secara penuh sekaligus tetap memberikan manfaat reduksi dimensi yang efisien tanpa kehilangan informasi yang berdampak negatif pada performa model. Dengan demikian, proses reduksi dimensi tetap menjaga karakteristik data yang relevan dan esensial dalam deteksi serangan IoT.



Gambar 6.4 *Heatmap* Kontribusi Fitur pada Komponen PCA

Gambar 6.4 menampilkan *heatmap* loading PCA yang menggambarkan kontribusi masing-masing fitur asli terhadap komponen utama PC1 hingga PC9. Warna pada diagram menunjukkan besar nilai loading, di mana semakin terang warnanya berarti semakin besar kontribusi fitur tersebut dalam membentuk suatu komponen PCA. Dari visualisasi tersebut dapat dilihat bahwa setiap komponen

memiliki fitur dominan yang berbeda-beda. Sebagai contoh, fitur udp.srcport dan udp.dstport tampak sangat dominan pada PC1 dan PC7, yang menunjukkan bahwa pola komunikasi berbasis UDP menjadi salah satu karakteristik penting dalam representasi variansi awal. Selain itu, PC3 menunjukkan dominasi kuat dari fitur ip.ttl, yang mengindikasikan bahwa variasi jumlah hop pada trafik berpengaruh besar terhadap pembentukan komponen tersebut. Sementara itu, PC4 didominasi oleh tcp.pdu.size yang menggambarkan pengaruh ukuran payload dalam variansi data. Pola fitur yang berbeda-beda pada setiap komponen ini menunjukkan bahwa PCA dapat memisahkan ragam pola serangan berdasarkan karakteristik fitur jaringan yang spesifik. Informasi penting yang berkaitan dengan pola serangan IoT juga tetap dipertahankan meskipun jumlah fitur telah direduksi menjadi sembilan komponen utama.

Penelitian ini menggunakan lima skenario pembagian data pelatihan dan pengujian yang masing-masing diberi kode XGBSP-E1 hingga XGBSP-E5, sebagaimana ditunjukkan pada Tabel 6.2. Setiap skenario memiliki proporsi data yang berbeda mulai dari 50:50 hingga 90:10 untuk mengamati pengaruh perbedaan rasio terhadap stabilitas dan generalisasi model.

Tabel 6.2 Distribusi Data Pelatihan Model XGBSP

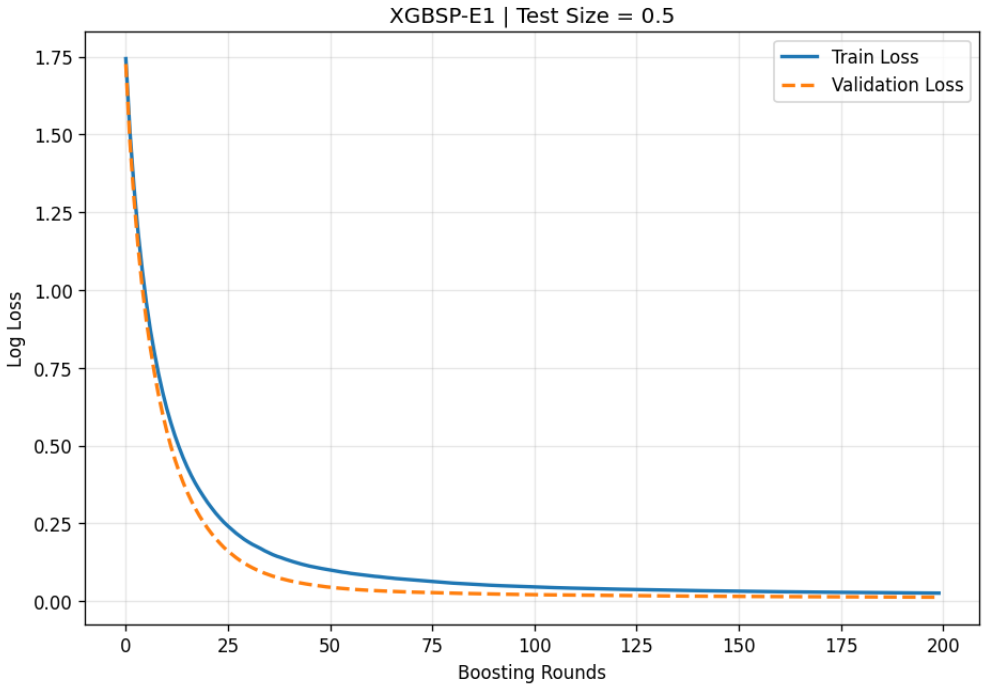
Nama Pelatihan	**Komposisi Data**	**Data Latih**	**SMOTE**	**Data Uji**
XGBSP-E1	50:50	175.000	858.784	175.000
XGBSP-E2	60:40	210.000	1.030.544	140.000
XGBSP-E3	70:30	245.000	1.202.304	105.000
XGBSP-E4	80:20	280.000	1.374.064	70.000
XGBSP-E5	90:10	315.000	1.545.816	35.000

Melalui rancangan parameter dan konfigurasi yang sistematis ini, model XGBoost menggunakan SMOTE dan PCA diharapkan mampu memberikan hasil

klasifikasi yang lebih seimbang, akurat, dan efisien dalam mendeteksi berbagai kategori serangan siber pada sistem IoT. Kombinasi ini tidak hanya meningkatkan kinerja model dalam mengenali kelas minoritas, tetapi juga mempercepat proses pelatihan melalui reduksi dimensi yang efektif.

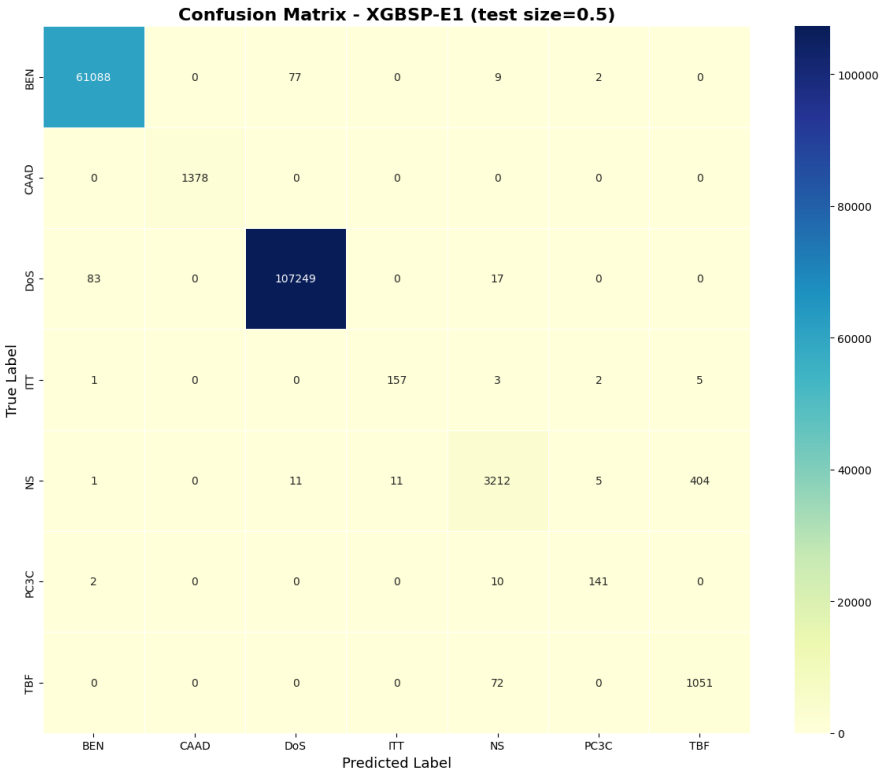
6.2.1 Pelatihan Model XGBSP-E1

Proses eksperimen dari pelatihan model XGBSP-E1 dalam melakukan klasifikasi serangan siber pada IoT digunakan proporsi data latih sebesar 50% atau sekitar 175.000 data dan data uji sebesar 50% atau sekitar 175.000 data. Setelah dilakukan proses SMOTE jumlah data *training* menjadi 858.784 data dengan masing-masing kelas sejumlah 107.348 data. Model dilatih dengan jumlah n_estimators yang telah ditentukan, yaitu sebanyak 200 pohon. Hasil pelatihan menunjukkan nilai *train loss* sekitar 0.025754 dan nilai *validation loss* sekitar 0.012604 sebagaimana ditampilkan pada Gambar 6.5.



Gambar 6.5 Grafik Nilai *Train* dan *Validation Loss* Model XGBSP-E1

Gambar 6.5 menunjukkan nilai *train* dan *validation loss* pada proses pelatihan model XGBSP-E1. Gambar menunjukkan nilai yang terus mengalami penurunan hingga mencapai titik konvergensi pada n_estimators ke-200, sehingga model tersebut dianggap telah optimal. Selanjutnya, model optimal yang telah didapatkan, kemudian digunakan untuk melakukan klasifikasi serangan siber berdasarkan kategori serangan siber, meliputi *Denial of Servic*e (DoS), *Benign* (BEN), *Network Scanning* (NS), CoAP *Amplification Attack Data* (CAAD), *Telnet Brute Forcing* (TBF), *Ingress Tool Transfer* (ITT), dan *Periodic Command and Control* (C&C) *Communication* (PC3C). Hasil dari klasifikasi tersebut disajikan dalam bentuk *confusion matrix* pada Gambar 6.6.



Gambar 6.6 Hasil Klasifikasi Model XGBSP-E1 Terhadap Serangan Siber

Gambar 6.6 menunjukkan model XGBSP-E1 yang diuji dengan proporsi data latih dan uji 50:50 menampilkan performa klasifikasi yang sangat baik. Kelas BEN berhasil diklasifikasikan dengan benar sebanyak 61.088 data dengan 88 kesalahan minor, sebagian besar teridentifikasi keliru sebagai DoS dan NS. Kelas CAAD menunjukkan hasil sempurna dengan 1.378 prediksi benar tanpa kesalahan, sedangkan DoS mencatat 107.249 prediksi tepat dengan tingkat kesalahan sangat rendah. Meskipun demikian, kesalahan paling mencolok terjadi pada kelas NS dan TBF, di mana sebanyak 404 data NS salah diklasifikasikan sebagai TBF, serta 72 data TBF terdeteksi sebagai NS. Hal ini menunjukkan bahwa fitur lalu lintas jaringan antara kedua jenis serangan tersebut masih tumpang tindih, sehingga menimbulkan ambiguitas bagi model. Nilai TP, TN, FP, dan FN dari hasil klasifikasi serangan siber secara rinci disajikan dalam Tabel 6.3.

Tabel 6.3 Nilai TP, TN, FP, dan FN Setiap Kategori pada Model XGBSP-E1

Kelas	***True Positives* (TP)**	***True Negatives* (TN)**	***False Positives* (FP)**	***False Negatives* (FN)**
BEN	61088	113736	88	88
CAAD	1378	173622	0	0
DoS	107249	67563	88	100
ITT	157	174815	17	11
NS	3212	171245	111	432
PC3C	141	174838	9	12
TBF	1051	173462	409	78

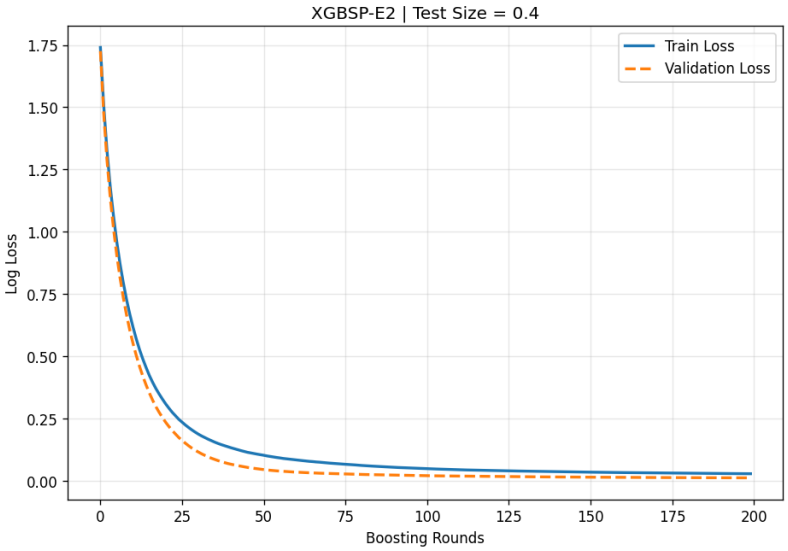
Setelah didapatkan nilai TP, TN, FP, dan FN dari setiap kategori serangan siber, langkah selanjutnya adalah menghitung metrik evaluasi yang digunakan untuk mengetahui performa model XGBSP-E1 dalam klasifikasi serangan siber pada IoT berdasarkan kategorinya yang secara rinci disajikan pada Tabel 6.4.

Tabel 6.4 Performa Hasil Pengujian Model XGBSP-E1

Kelas	Akurasi (%)	Presisi	*Recall*	*F1-Score*	*Support*
BEN	99.58	1	1	1	61176
CAAD		1	1	1	1378
DoS		1	1	1	107349
ITT		0.9	0.93	0.92	168
NS		0.97	0.88	0.92	3644
PC3C		0.94	0.92	0.93	153
TBF		0.72	0.93	0.81	1129

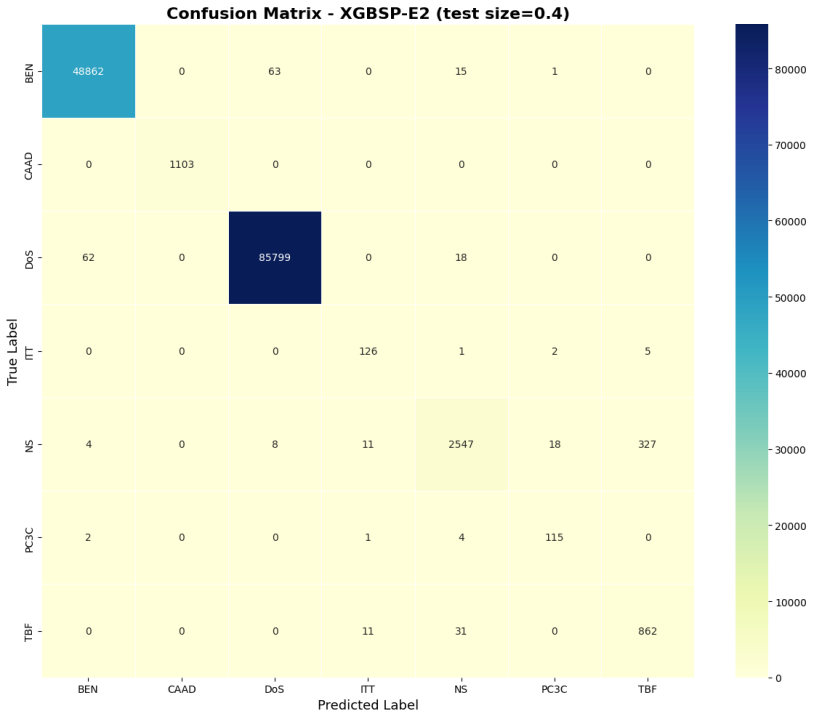
6.2.2 Pelatihan Model XGBSP-E2

Proses eksperimen dari pelatihan model XGBSP-E2 dalam melakukan klasifikasi serangan siber pada IoT digunakan proporsi data latih sebesar 60% atau sekitar 210.000 data dan data uji sebesar 40% atau sekitar 140.000 data. Setelah dilakukan proses SMOTE jumlah data *training* menjadi 1.030.544 data dengan masing-masing kelas sejumlah 128.818 data. Model dilatih dengan jumlah n_estimators yang telah ditentukan, yaitu sebanyak 200 pohon. Hasil pelatihan menunjukkan nilai *train loss* sekitar 0.028932 dan nilai *validation loss* sekitar 0.012381 sebagaimana ditampilkan pada Gambar 6.7.



Gambar 6.7 Grafik Nilai *Train* dan *Validation Loss* Model XGBSP-E2

Gambar 6.7 menunjukkan nilai *train* dan *validation loss* pada proses pelatihan model XGBSP-E2 yang terus mengalami penurunan hingga mencapai titik konvergensi pada n_estimators ke-200, sehingga model tersebut dianggap telah optimal. Hasil dari klasifikasi model disajikan dalam bentuk *confusion matrix* pada Gambar 6.8



Gambar 6.8 Hasil Klasifikasi Model XGBSP-E2 Terhadap Serangan Siber

Pada model XGBSP-E2 dengan rasio data latih dan uji 60:40, performa model tetap stabil dengan peningkatan akurasi pada beberapa kelas. Kelas BEN memiliki 48.862 prediksi benar dan hanya 79 kesalahan, sementara CAAD kembali menunjukkan hasil sempurna. Kelas DoS juga tetap konsisten dengan 85.799 data benar, hanya mengalami sedikit kesalahan yang tidak signifikan terhadap performa keseluruhan. Sementara itu, NS dan TBF masih menjadi sumber kesalahan utama.

Tercatat 327 data NS salah diprediksi sebagai TBF dan 31 data TBF salah sebagai NS, yang menunjukkan bahwa meskipun model sudah belajar lebih banyak pola dari data latih, distribusi fitur serangan minor masih sulit dipisahkan secara optimal. Nilai TP, TN, FP, dan FN dari hasil klasifikasi serangan secara rinci disajikan dalam Tabel 6.5.

Tabel 6.5 Nilai TP, TN, FP, dan FN Setiap Kategori pada Model XGBSP-E2

Kelas	***True Positives* (TP)**	***True Negatives* (TN)**	***False Positives* (FP)**	***False Negatives* (FN)**
BEN	48862	90991	68	79
CAAD	1103	138897	0	0
DoS	85799	54050	71	80
ITT	126	139843	23	8
NS	2547	137016	69	368
PC3C	115	139857	21	7
TBF	862	138764	332	42

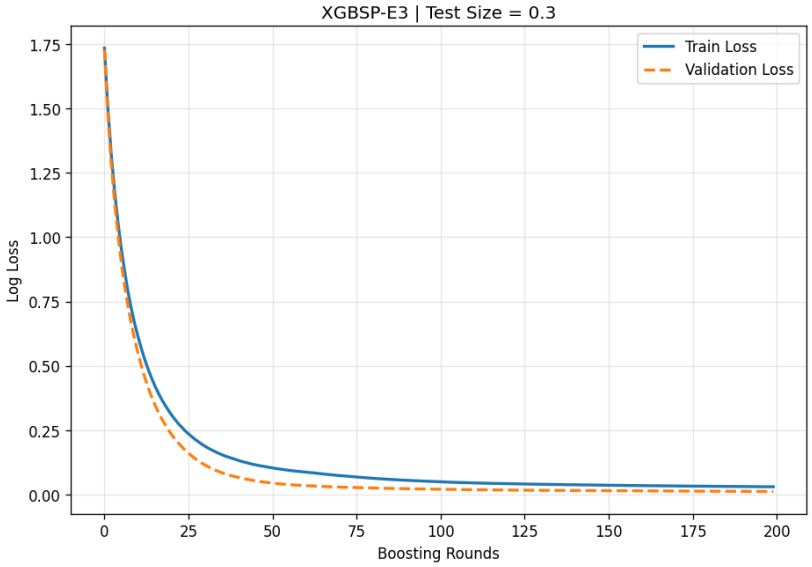
Setelah didapatkan nilai TP, TN, FP, dan FN dari setiap kategori serangan siber, langkah selanjutnya adalah menghitung metrik evaluasi untuk mengetahui performa model XGBSP-E2. Metrik evaluasi dalam klasifikasi serangan siber pada IoT berdasarkan kategorinya secara rinci disajikan pada Tabel 6.6.

Tabel 6.6 Performa Hasil Pengujian Model XGBSP-E2

Kelas	**Akurasi (%)**	**Presisi**	***Recall***	***F1-Score***	***Support***
BEN	99.58	1	1	1	48941
CAAD		1	1	1	1103
DoS		1	1	1	85879
ITT		0.85	0.94	0.89	134
NS		0.97	0.87	0.92	2915
PC3C		0.85	0.94	0.89	122
TBF		0.72	0.95	0.82	904

6.2.3 Pelatihan Model XGBSP-E3

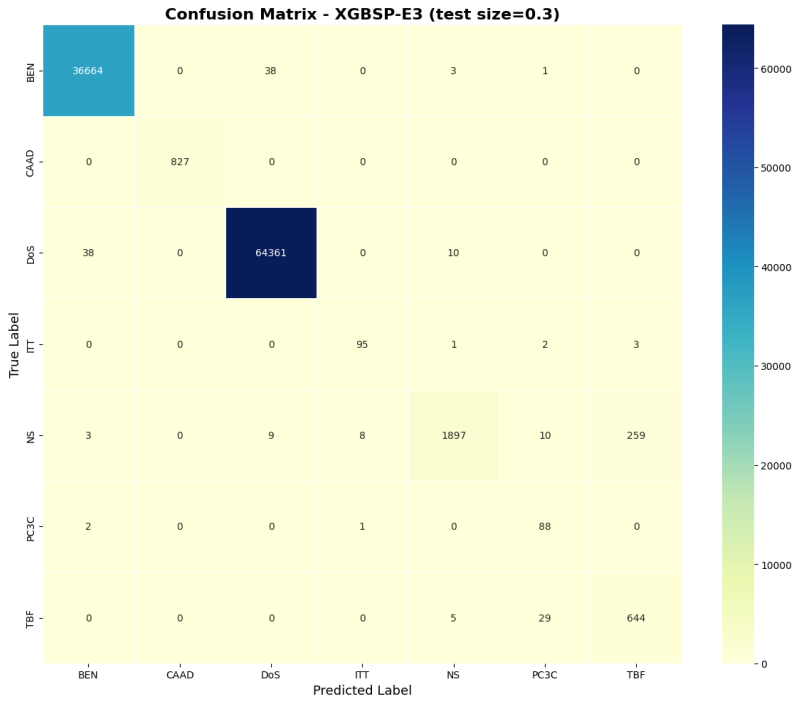
Proses eksperimen dari pelatihan model XGBSP-E3 dalam melakukan klasifikasi serangan siber pada IoT digunakan proporsi data latih sebesar 70% atau sekitar 245.000 data dan data uji sebesar 30% atau sekitar 105.000 data. Setelah dilakukan proses SMOTE jumlah data *training* menjadi 1.202.304 data dengan masing-masing kelas sejumlah 150.288 data. Model dilatih dengan jumlah n_estimators yang telah ditentukan, yaitu sebanyak 200 pohon. Hasil pelatihan menunjukkan nilai *train loss* sekitar 0.030375 dan nilai *validation loss* sekitar 0.012127 sebagaimana ditampilkan padaTabel 6.7.



Gambar 6.9 Grafik Nilai *Train* dan *Validation Loss* Model XGBSP-E3

Gambar 6.9 menunjukkan nilai *train* dan *validation loss* pada proses pelatihan model XGBSP-E3 yang terus mengalami penurunan hingga mencapai titik konvergensi pada n_estimators ke-200, sehingga model tersebut dianggap telah optimal. Selanjutnya, model optimal yang telah didapatkan, kemudian digunakan untuk melakukan klasifikasi serangan siber berdasarkan kategori serangan siber,

meliputi *Denial of Service* (DoS), *Benign* (BEN), *Network Scanning* (NS), CoAP *Amplification Attack Data* (CAAD), *Telnet Brute Forcing* (TBF), *Ingress Tool Transfer* (ITT), dan *Periodic Command and Control* (C&C) *Communication* (PC3C). Hasil dari klasifikasi tersebut disajikan dalam bentuk *confusion matrix* pada Gambar 6.10.



Gambar 6.10 Hasil Klasifikasi Model XGBSP-E3 Terhadap Serangan Siber

Model XGBSP-E3 menghasilkan kestabilan klasifikasi yang lebih baik. Kelas BEN berhasil diprediksi dengan benar sebanyak 36.664 data dengan hanya 42 kesalahan, sedangkan CAAD tetap akurat dengan 827 prediksi benar tanpa kesalahan. Model ini juga mempertahankan performa tinggi pada kelas DoS, dengan 64.361 prediksi benar dan 48 kesalahan. Namun, seperti pola sebelumnya, kelas NS dan TBF masih menunjukkan keterkaitan yang kuat. Sebanyak 259 data NS salah diklasifikasikan sebagai TBF, sementara 29 data TBF salah terdeteksi

sebagai NS. Hal ini menegaskan bahwa meskipun model semakin optimal dalam mendeteksi pola mayoritas, hubungan antar fitur serangan minor masih menimbulkan potensi salah dalam melakukan klasifikasi. Nilai TP, TN, FP, dan FN dari hasil klasifikasi serangan siber secara rinci disajikan dalam Tabel 6.7.

Tabel 6.7 Nilai TP, TN, FP, dan FN Setiap Kategori pada Model XGBSP-E3

Kelas	*True Positives* (TP)	*True Negatives* (TN)	*False Positives* (FP)	*False Negatives* (FN)
BEN	36664	68251	43	42
CAAD	827	104173	0	0
DoS	64361	40544	47	48
ITT	95	104885	14	6
NS	1897	102771	43	289
PC3C	88	104896	13	3
TBF	644	104060	262	34

Setelah didapatkan nilai TP, TN, FP, dan FN dari setiap kategori serangan siber, langkah selanjutnya adalah menghitung metrik evaluasi yang digunakan untuk mengetahui performa model XGBSP-E3. Metrik evaluasi dalam klasifikasi serangan siber pada IoT secara rinci disajikan pada Tabel 6.8.

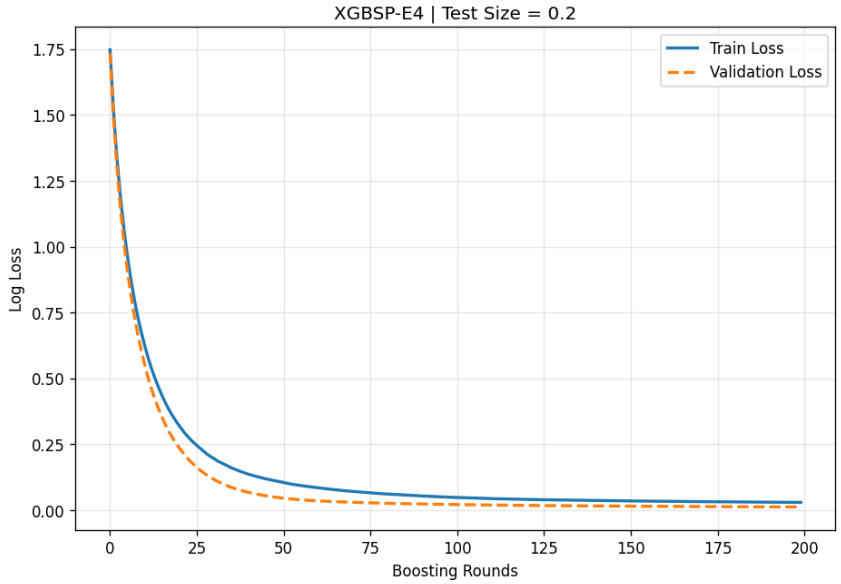
Tabel 6.8 Performa Hasil Pengujian Model XGBSP-E3

Kelas	**Akurasi (%)**	**Presisi**	***Recall***	***F1-Score***	***Support***
BEN	99.59	1	1	1	36706
CAAD		1	1	1	827
DoS		1	1	1	64409
ITT		0.87	0.94	0.9	101
NS		0.98	0.87	0.92	2186
PC3C		0.87	0.97	0.92	91
TBF		0.71	0.95	0.81	678

6.2.4 Pelatihan Model XGBSP-E4

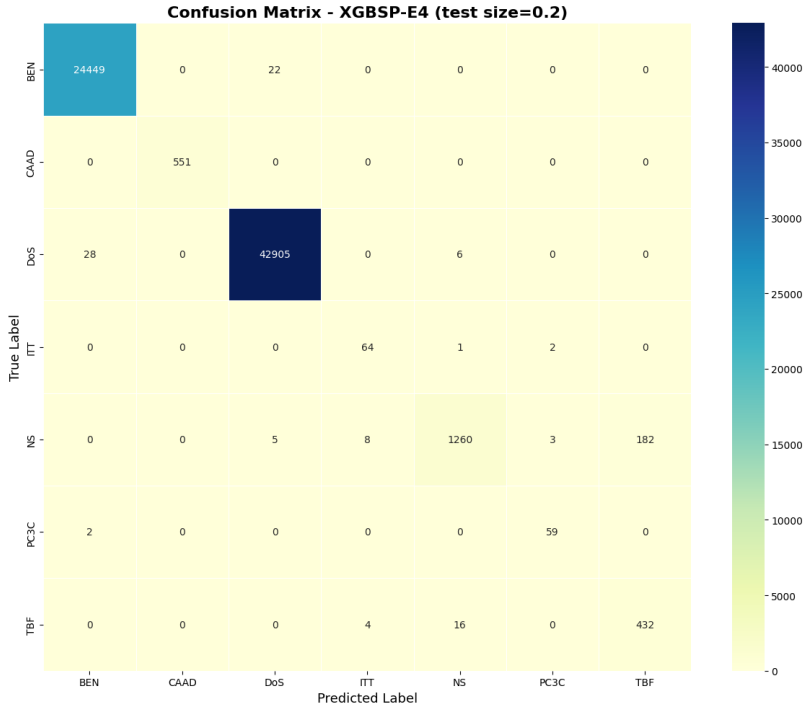
Proses eksperimen dari pelatihan model XGBSP-E4 dalam melakukan klasifikasi serangan siber pada IoT digunakan proporsi data latih sebesar 80% atau

sekitar 280.000 data dan data uji sebesar 20% atau sekitar 70.000 data. Setelah dilakukan proses SMOTE jumlah data *training* menjadi 1.374.064 data dengan masing-masing kelas sejumlah 171.758 data. Model dilatih dengan jumlah n_estimators yang telah ditentukan, yaitu sebanyak 200 pohon. Hasil pelatihan menunjukkan nilai *train loss* sekitar 0.029332 dan nilai *validation loss* sekitar 0.011991 sebagaimana ditampilkan pada Gambar 6.11.



Gambar 6.11 Grafik Nilai *Train* dan *Validation Loss* Model XGBSP-E4

Gambar 6.11 menunjukkan nilai *train* dan *validation loss* pada proses pelatihan model XGBSP-E4 yang terus mengalami penurunan hingga mencapai titik konvergensi pada n_estimators ke-200, sehingga model tersebut dianggap telah optimal. Selanjutnya, model yang telah didapatkan kemudian digunakan untuk melakukan klasifikasi serangan siber berdasarkan kategori serangan siber. Hasil dari klasifikasi tersebut disajikan dalam bentuk *confusion matrix* pada Gambar 6.12.



Gambar 6.12 Hasil Klasifikasi Model XGBSP-E4 Terhadap Serangan Siber

Model XGBSP-E4 yang menggunakan proporsi data latih 80% memperlihatkan hasil yang lebih stabil dan presisi tinggi. Kelas BEN memiliki 24.449 prediksi benar dengan 22 kesalahan, sedangkan CAAD tercatat tanpa kesalahan sama sekali. Model ini menunjukkan peningkatan kemampuan dalam mendeteksi kelas minor seperti PC3C, yang memperoleh 59 data benar dengan kesalahan minimal. Sementara itu, kelas NS dan TBF masih memperlihatkan tingkat kesalahan moderat, dengan 182 data NS salah diprediksi sebagai TBF, dan 16 data TBF salah diklasifikasikan sebagai NS. Hasil ini memperlihatkan bahwa semakin besar porsi data latih, semakin baik model dalam memahami karakteristik serangan utama, meskipun ambiguitas pada serangan minor tetap muncul. Nilai TP, TN, FP, dan FN dari hasil klasifikasi serangan siber secara rinci disajikan dalam Tabel 6.9.

Tabel 6.9 Nilai TP, TN, FP, dan FN Setiap Kategori pada Model XGBSP-E4

Kelas	***True Positives (TP)***	***True Negatives (TN)***	***False Positives (FP)***	***False Negatives (FN)***
BEN	24449	45499	30	22
CAAD	551	69449	0	0
DoS	42905	27034	27	34
ITT	64	69921	12	3
NS	1260	68519	23	198
PC3C	59	69934	5	2
TBF	432	69366	182	20

Setelah didapatkan nilai TP, TN, FP, dan FN dari setiap kategori serangan siber, langkah selanjutnya adalah menghitung metrik evaluasi yang digunakan untuk mengetahui performa model XGBSP-E4. Metrik evaluasi tersebut secara rinci disajikan pada Tabel 6.10.

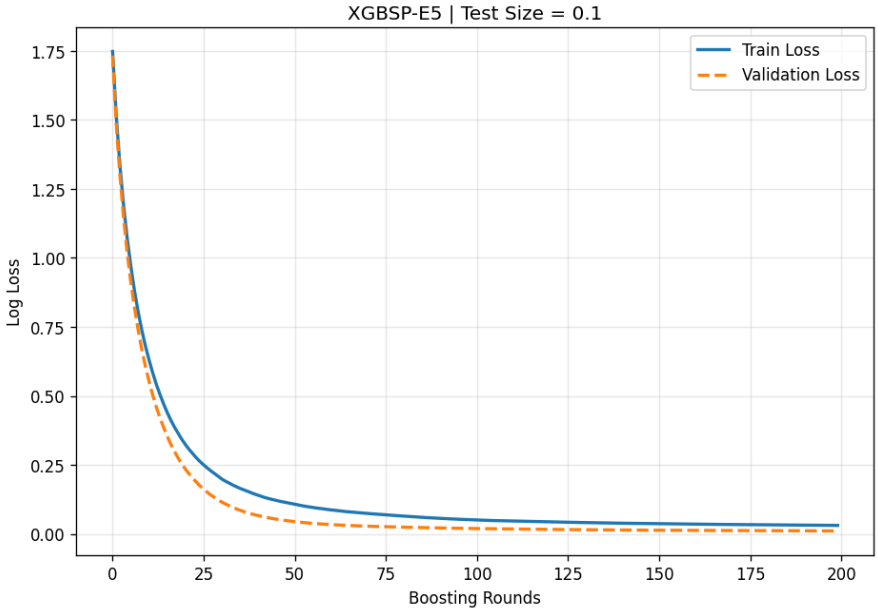
Tabel 6.10 Performa Hasil Pengujian Model XGBSP-E4

Kelas	**Akurasi (%)**	**Presisi**	***Recall***	***F1-Score***	***Support***
BEN	99.61	1	1	1	24471
CAAD		1	1	1	551
DoS		1	1	1	42939
ITT		0.84	0.96	0.9	67
NS		0.98	0.86	0.92	1458
PC3C		0.92	0.97	0.94	61
TBF		0.7	0.96	0.81	452

6.2.5 Pelatihan Model XGBSP-E5

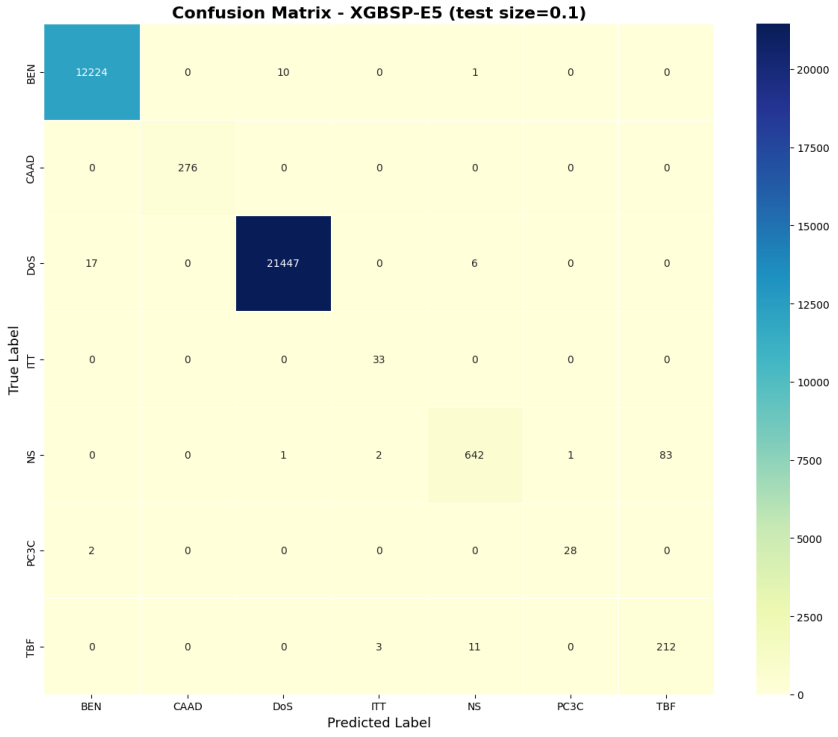
Proses eksperimen dari pelatihan model XGBSP-E5 dalam melakukan klasifikasi serangan siber pada IoT digunakan proporsi data latih sebesar 90% atau sekitar 315.000 data dan data uji sebesar 10% atau sekitar 35.000 data. Setelah dilakukan proses SMOTE jumlah data *training* menjadi 1.545.816 data dengan masing-masing kelas sejumlah 193.227 data. Model dilatih dengan jumlah

n_estimators yang telah ditentukan, yaitu sebanyak 200 pohon. Hasil pelatihan menunjukkan nilai *train loss* sekitar 0.031188 dan nilai *validation loss* sekitar 0.01116 sebagaimana ditampilkan pada Gambar 6.13.



Gambar 6.13 Grafik Nilai *Train* dan *Validation Loss* Model XGBSP-E5

Gambar 6.13 menunjukkan nilai *train* dan *validation loss* pada proses pelatihan model XGBSP-E5 yang terus mengalami penurunan hingga mencapai titik konvergensi pada n_estimators ke-200, sehingga model tersebut dianggap telah optimal. Selanjutnya, model optimal yang telah didapatkan, kemudian digunakan untuk melakukan klasifikasi serangan siber berdasarkan kategori serangan siber. Hasil dari klasifikasi tersebut disajikan dalam bentuk *confusion matrix* pada Gambar 6.13.



Gambar 6.14 Hasil Klasifikasi Model XGBSP-E5 Terhadap Serangan Siber

Model XGBSP-E5 yang dilatih dengan komposisi 90% data latih dan 10% data uji menampilkan hasil terbaik di antara seluruh model yang diuji. Kelas BEN memperoleh 12.224 prediksi benar dari 12.235 data uji, hanya mengalami 11 kesalahan minor, sementara CAAD dan PC3C menunjukkan akurasi sempurna (100%). Kelas DoS juga tampil sangat baik dengan 21.447 prediksi benar dari total 21.470 data, menunjukkan peningkatan ketepatan seiring pertambahan data latih. Namun, kelas NS dan TBF masih memperlihatkan fenomena yang sama dengan model sebelumnya, di mana 83 data NS salah diklasifikasikan sebagai TBF dan 11 data TBF salah terdeteksi sebagai NS.

Secara keseluruhan XGBSP-E5 menunjukkan akurasi tertinggi di antara seluruh model, mencerminkan bahwa semakin banyak data latih, semakin besar kemampuan model dalam melakukan generalisasi dan membedakan pola serangan

kompleks. Selanjutnya, untuk menghitung nilai metrik akurasi, presisi, *recall*, dan *F1-score* maka dibutuhkan nilai TP, TN, FP, dan FN dari hasil klasifikasi serangan siber berdasarkan kategori serangan siber secara rinci disajikan dalam Tabel 6.11.

Tabel 6.11 Nilai TP, TN, FP, dan FN Setiap Kategori pada Model XGBSP-E5

Kelas	***True Positives* (TP)**	***True Negatives* (TN)**	***False Positives* (FP)**	***False Negatives* (FN)**
BEN	12224	22746	19	11
CAAD	276	34724	0	0
DoS	21447	13519	11	23
ITT	33	34962	5	0
NS	642	34253	18	87
PC3C	28	34969	1	2
TBF	212	34691	83	14

Setelah didapatkan nilai TP, TN, FP, dan FN dari setiap kategori serangan siber, langkah selanjutnya adalah menghitung metrik evaluasi yang digunakan untuk mengetahui performa model XGBSP-E5 dalam klasifikasi serangan siber pada IoT berdasarkan kategorinya yang secara rinci disajikan pada Tabel 6.12.

Tabel 6.12 Performa Hasil Pengujian Model XGBSP-E5

Kelas	**Akurasi (%)**	**Presisi**	***Recall***	***F1-Score***	***Support***
BEN	99.68	1	1	1	12235
CAAD		1	1	1	276
DoS		1	1	1	21470
ITT		0.87	1	0.93	33
NS		0.97	0.88	0.92	729
PC3C		0.97	0.93	0.95	30
TBF		0.72	0.94	0.81	226

6.3 Hasil Ujicoba Model XGBSP

Setelah proses pelatihan selesai, dilakukan proses ujicoba pada masing-masing model XGBoost dengan SMOTE dan PCA menggunakan data uji guna menilai kinerjanya dalam mengklasifikasikan serangan siber berdasarkan

kategorinya. Selanjutnya, dilakukan proses evaluasi performa model menggunakan nilai akurasi, presisi, *recall,* dan *F1-score,* dimana masing-masing nilai memberikan perspektif berbeda terhadap performa model.

Tabel 6.13 Hasil Ujicoba Klasifikasi Serangan Siber pada Model XGBSP

Model	**Kelas**	**Akurasi (%)**	**Presisi**	**Recall**	**F1-Score**	**Support**
XGBSP-E1	BEN	99.58	1	1	1	61176
	CAAD		1	1	1	1378
	DoS		1	1	1	107349
	ITT		0.9	0.93	0.92	168
	NS		0.97	0.88	0.92	3644
	PC3C		0.94	0.92	0.93	153
	TBF		0.72	0.93	0.81	1129
XGBSP-E2	BEN	99.58	1	1	1	48941
	CAAD		1	1	1	1103
	DoS		1	1	1	85879
	ITT		0.85	0.94	0.89	134
	NS		0.97	0.87	0.92	2915
	PC3C		0.85	0.94	0.89	122
	TBF		0.72	0.95	0.82	904
XGBSP-E3	BEN	99.59	1	1	1	36706
	CAAD		1	1	1	827
	DoS		1	1	1	64409
	ITT		0.87	0.94	0.9	101
	NS		0.98	0.87	0.92	2186
	PC3C		0.87	0.97	0.92	91
	TBF		0.71	0.95	0.81	678
XGBSP-E4	BEN	99.61	1	1	1	24471
	CAAD		1	1	1	551
	DoS		1	1	1	42939
	ITT		0.84	0.96	0.9	67
	NS		0.98	0.86	0.92	1458
	PC3C		0.92	0.97	0.94	61
	TBF		0.7	0.96	0.81	452
XGBSP-E5	BEN	99.68	1	1	1	12235
	CAAD		1	1	1	276
	DoS		1	1	1	21470
	ITT		0.87	1	0.93	33
	NS		0.97	0.88	0.92	729
	PC3C		0.97	0.93	0.95	30
	TBF		0.72	0.94	0.81	226

Tabel 6.13 menunjukkan bahwa model XGBSP-E1 yang menggunakan kombinasi SMOTE dan PCA memiliki performa klasifikasi sangat baik pada sebagian besar kelas serangan siber. Kelas mayoritas seperti BEN, CAAD, dan DoS menunjukkan hasil sempurna dengan nilai presisi, *recall*, dan *F1-score* sebesar 1, menandakan kemampuan model yang sangat akurat dalam mengenali data normal dan serangan utama. Namun, pada kelas minoritas seperti ITT, NS, PC3C, dan TBF, performa sedikit menurun dengan nilai *F1-score* berkisar antara 0,81 hingga 0,93. Hal ini menunjukkan bahwa meskipun SMOTE membantu menyeimbangkan data, masih terdapat kesulitan model dalam membedakan pola serangan yang memiliki kemiripan perilaku trafik. Secara umum, XGBSP-E1 mampu meningkatkan representasi kelas minoritas dengan bantuan SMOTE, meskipun masih terdapat ruang perbaikan pada beberapa jenis serangan yang kompleks.

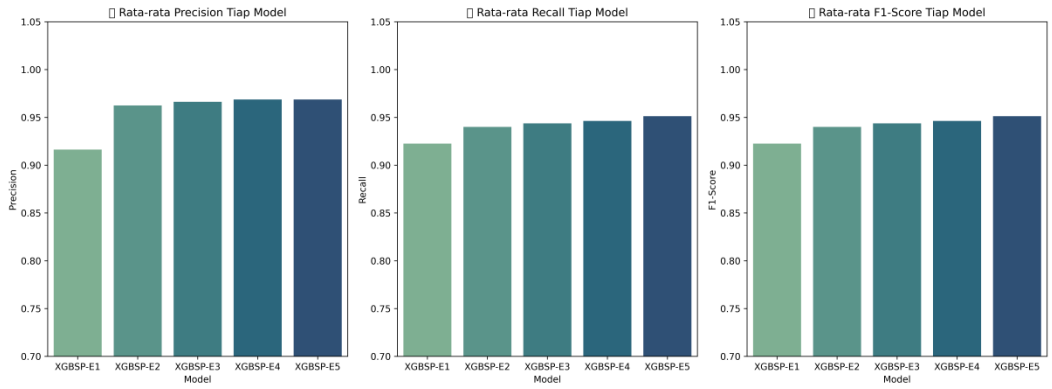
Selanjutnya, model XGBSP-E2 menunjukkan pola peningkatan kinerja yang lebih baik dibandingkan XGBSP-E1, khususnya pada kelas ITT, PC3C, dan TBF, dengan peningkatan nilai *recall* hingga 0,95. Hal ini menunjukkan bahwa kombinasi SMOTE dan PCA mulai memberikan efek positif terhadap pembelajaran model pada kelas minoritas. Kelas mayoritas seperti BEN, CAAD, dan DoS tetap menunjukkan hasil sempurna, memperlihatkan stabilitas model pada data dengan distribusi besar. Meskipun demikian, kelas NS masih memiliki tantangan dengan *F1-score* 0,92 karena pola trafiknya yang cenderung mirip dengan kelas lainnya. Secara keseluruhan, XGBSP-E2 memperlihatkan peningkatan signifikan dalam sensitivitas terhadap data minoritas tanpa mengorbankan akurasi global.

Model XGBSP-E3 mendapatkan performa klasifikasi semakin stabil dan menunjukkan kemampuan generalisasi yang lebih baik dibandingkan dua model sebelumnya. Kelas BEN, CAAD, dan DoS tetap mempertahankan hasil sempurna (*F1-score* 1), sementara kelas ITT dan PC3C menunjukkan peningkatan *nilai F1-score* menjadi 0,9 dan 0,92, yang menandakan model semakin adaptif terhadap variasi pola serangan IoT dan komunikasi port. Kelas NS dan TBF juga menunjukkan hasil konsisten dengan *F1-score* 0,92 dan 0,81. Secara umum, XGBSP-E3 berhasil menjaga keseimbangan antara akurasi tinggi dan generalisasi yang baik, menandakan bahwa rasio pelatihan yang lebih besar memperkuat stabilitas performa model.

Model XGBSP-E4 memperlihatkan tren peningkatan lanjutan dengan hasil yang semakin konsisten di seluruh kelas. Kelas utama seperti BEN, CAAD, dan DoS tetap menunjukkan nilai sempurna, sedangkan kelas ITT, NS, dan PC3C mengalami sedikit peningkatan pada *F1-score* masing-masing menjadi 0,9, 0,92, dan 0,94. Kelas TBF tetap stabil dengan *F1-score* 0,81, menunjukkan bahwa SMOTE telah berhasil membantu model mengenali pola *flooding* yang kompleks. Secara keseluruhan, XGBSP-E4 menampilkan kinerja yang lebih matang dan konsisten dengan keseimbangan ideal antara presisi dan *recall* di seluruh kelas.

Performa model XGBSP-E5 yang terdapat pada Gambar 6.15 dengan proporsi data latih terbesar (90%), menunjukkan performa paling optimal di antara semua model. Seluruh kelas mayoritas (BEN, CAAD, DoS) mencatatkan hasil sempurna, dan peningkatan signifikan terlihat pada kelas ITT dengan *F1-score* 0,93 serta PC3C dengan 0,95. Kelas NS juga mempertahankan performa stabil dengan

F1-score 0,92, dan TBF tetap pada 0,81. Hasil ini menunjukkan bahwa kombinasi antara jumlah data latih yang besar, oversampling SMOTE, dan reduksi dimensi PCA mampu memperkuat generalisasi model tanpa kehilangan akurasi. Dengan demikian, XGBSP-E5 menjadi konfigurasi paling optimal untuk mendeteksi berbagai jenis serangan siber secara efisien, terutama pada lingkungan IoT yang memiliki variasi data dan serangan yang sangat dinamis.



Gambar 6.15 Rata-Rata Performa Model XGBSP

Secara keseluruhan, kelima model XGBSP (E1–E5) menunjukkan bahwa kombinasi SMOTE dan PCA secara signifikan meningkatkan performa klasifikasi XGBoost dalam mendeteksi serangan siber pada jaringan IoT. SMOTE efektif menyeimbangkan data minoritas seperti ITT, PC3C, dan TBF, sedangkan PCA mengurangi kompleksitas fitur sehingga model lebih fokus pada pola penting. Peningkatan proporsi data latih juga berpengaruh positif terhadap stabilitas dan akurasi, di mana XGBSP-E5 menunjukkan performa terbaik dengan keseimbangan sempurna antara presisi, *recall*, dan *F1-score* di seluruh kelas. Dengan akurasi di atas 99%, model ini terbukti efisien, adaptif terhadap variasi serangan, serta ideal diterapkan pada sistem Intrusion Detection System (IDS) berbasis IoT.

Tabel 6.14 Performa Model XBSP

Model	Komposisi Data	Nilai Akurasi
XGBSP-E1	50:50	99.58%
XGBSP-E2	60:40	99.58%
XGBSP-E3	70:30	99.59%
XGBSP-E4	80:20	99.61%
XGBSP-E5	**90:10**	**99.68%**

Tabel 6.14 menunjukkan bahwa model XGBSP-E1 dengan komposisi data 50:50 memperoleh nilai akurasi sebesar 99,58%, yang menandakan bahwa kombinasi antara SMOTE (*Synthetic Minority Over-sampling Technique*) dan PCA (*Principal Component Analysis*) mampu meningkatkan kemampuan model XGBoost dalam mengenali berbagai jenis serangan siber dengan baik, bahkan pada proporsi data latih yang masih terbatas. Selanjutnya, model XGBSP-E2 (60:40) menunjukkan hasil akurasi yang sama, yakni 99,58%, menggambarkan bahwa penambahan jumlah data latih belum memberikan peningkatan signifikan, tetapi sudah cukup menjaga kestabilan performa model terhadap seluruh kelas serangan. Model XGBSP-E3 (70:30) memperlihatkan peningkatan kecil dengan akurasi 99,59%, menandakan bahwa semakin banyak data pelatihan yang digunakan, model semakin adaptif dalam mengenali pola serangan yang kompleks berkat kombinasi oversampling dari SMOTE dan reduksi dimensi dari PCA. Sementara itu, model XGBSP-E4 (80:20) menunjukkan peningkatan yang lebih nyata dengan akurasi sekitar 99,61%, menunjukkan bahwa rasio pelatihan yang lebih besar memperkuat kemampuan generalisasi model tanpa kehilangan efisiensi komputasi. Model XGBSP-E5 (90:10) mencatatkan hasil tertinggi dengan akurasi sekitar 99,68%, menjadikannya sebagai model terbaik di antara seluruh konfigurasi karena berhasil memanfaatkan jumlah data latih yang besar serta kombinasi SMOTE dan

PCA secara optimal untuk meningkatkan akurasi, stabilitas, dan kemampuan generalisasi.

Secara keseluruhan, hasil evaluasi menunjukkan bahwa seluruh model XGBSP memiliki kinerja yang sangat unggul dan stabil dengan tingkat akurasi di atas 99,5%. Hal ini menegaskan bahwa penerapan kombinasi SMOTE dan PCA mampu meningkatkan efektivitas model XGBoost dalam menangani ketidakseimbangan data serta mengoptimalkan representasi fitur. Di antara seluruh konfigurasi yang diuji, model XGBSP-E5 dengan nilai akurasi sekitar 99,68% terbukti menjadi model paling optimal, karena berhasil mencapai keseimbangan terbaik antara proses penyeimbangan data, reduksi dimensi, serta proporsi data latih yang besar. Dengan demikian, model ini dinilai paling efektif dalam menghasilkan klasifikasi yang akurat, stabil, dan efisien untuk mendeteksi serangan siber pada lingkungan jaringan IoT.

Tabel 6.15 Nilai TP, TN, FP, dan FN Model XGBSP-E5

Kelas	***True Positives* (TP)**	***True Negatives* (TN)**	***False Positives* (FP)**	***False Negatives* (FN)**
BEN	12224	22746	19	11
CAAD	276	34724	0	0
DoS	21447	13519	11	23
ITT	33	34962	5	0
NS	642	34253	18	87
PC3C	28	34969	1	2
TBF	212	34691	83	14

Tabel 6.15 ditunjukkan hasil pengujian model XGBSP-E5, yaitu varian XGBoost yang dikombinasikan dengan SMOTE dan PCA dengan proporsi data latih 90:10. Hasil ini menggambarkan efektivitas model dalam mendeteksi berbagai

jenis serangan siber serta lalu lintas normal melalui analisis nilai *True Positives* (TP), *True Negatives* (TN), *False Positives* (FP), dan *False Negatives* (FN).

Kelas BEN (*Benign*) memperoleh TP = 12.224 dan TN = 22.746, dengan FP = 19 dan FN = 11. Nilai ini menunjukkan bahwa model memiliki tingkat ketepatan yang sangat tinggi dalam mengenali lalu lintas normal, meskipun masih terdapat sebagian kecil data yang salah diklasifikasikan sebagai serangan (FP) maupun yang tidak terdeteksi (FN). Pada kelas CAAD (*Command and Control Attack Detection*), model mencapai performa sempurna dengan TP = 276, TN = 34.724, FP = 0, dan FN = 0, yang menandakan kemampuan model dalam mengenali serangan *command and control* secara optimal tanpa kesalahan prediksi. Untuk kelas DoS (*Denial of Service*), diperoleh TP = 21.447, TN = 13.519, FP = 11, dan FN = 23. Nilai ini menggambarkan bahwa model mampu mendeteksi sebagian besar serangan DoS dengan akurasi tinggi, walaupun masih terdapat sejumlah kecil kesalahan klasifikasi akibat kesamaan pola antara trafik DoS dan trafik tinggi normal.

Selanjutnya, pada kelas ITT (*Ingress Tool Transfer*), hasil menunjukkan TP = 33, TN = 34.962, FP = 5, dan FN = 0, yang berarti model dapat mengenali serangan dengan tingkat kesalahan yang sangat kecil. Sementara itu, kelas NS (*Network Scan*) mencatatkan TP = 642, TN = 34.253, FP = 18, dan FN = 87. Nilai FN yang relatif tinggi dibandingkan kelas lain menunjukkan bahwa model masih sedikit kesulitan dalam mengenali seluruh variasi pola serangan scanning yang kompleks dan mirip dengan aktivitas normal jaringan. Kelas PC3C (*Communication*) memiliki TP = 28, TN = 34.969, FP = 1, dan FN = 2, menunjukkan performa deteksi yang sangat baik dengan hanya sedikit kesalahan

dalam pengenalan pola serangan. Sedangkan pada kelas TBF (*Telnet Brute Forcing*), diperoleh TP = 212, TN = 34.691, FP = 83, dan FN = 14. Meskipun nilai FP relatif lebih tinggi dibandingkan kelas lainnya, model tetap menunjukkan kemampuan yang baik dalam mengenali sebagian besar pola serangan brute force hasil oversampling dan reduksi dimensi PCA.

Secara keseluruhan, model XGBSP-E5 menunjukkan stabilitas dan akurasi yang sangat tinggi di seluruh kelas serangan dengan dominasi nilai TP dan TN yang besar serta FP dan FN yang rendah. Penerapan SMOTE berhasil menyeimbangkan distribusi data minoritas, sedangkan PCA berperan penting dalam mengurangi dimensi tanpa kehilangan informasi penting. Kombinasi kedua metode ini menjadikan XGBSP-E5 sebagai model yang efisien, cepat, dan akurat dalam mendeteksi berbagai serangan siber pada lingkungan IoT (*Internet of Things*) yang kompleks dan dinamis.

6.4 Kesimpulan

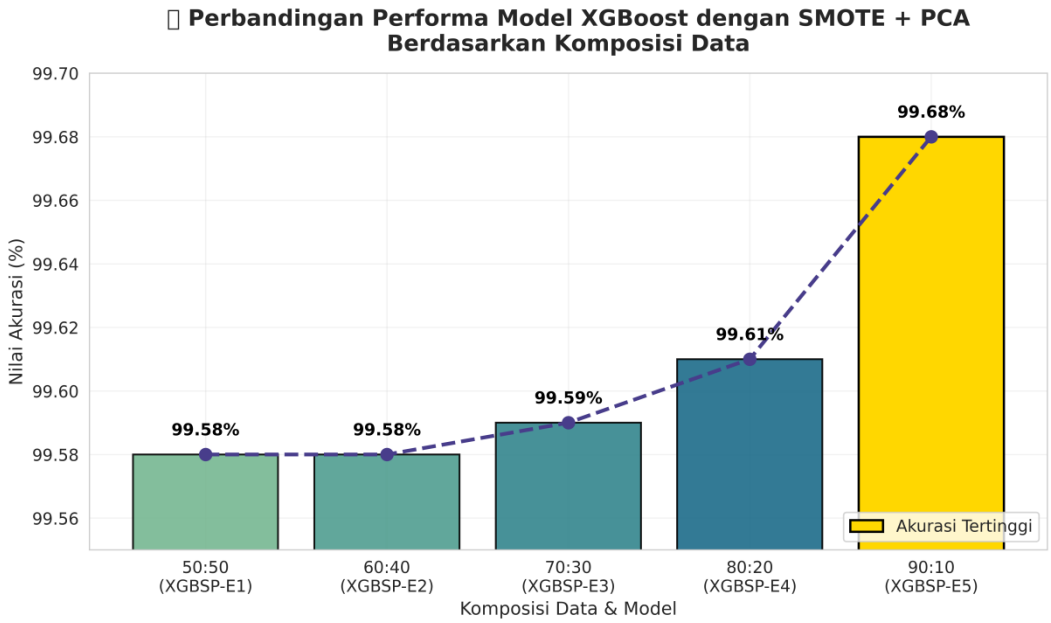
Berdasarkan hasil evaluasi menyeluruh terhadap model XGBoost dengan SMOTE dan PCA dalam proses optimasi model untuk mengklasifikasikan serangan siber berdasarkan kategorinya, dapat disimpulkan bahwa performa model sangat dipengaruhi oleh komposisi data latih dan data uji yang digunakan, terutama setelah dilakukan *balancing* menggunakan SMOTE dan reduksi dimensi menggunakan *Principal Component Analysis* (PCA). Dari kelima skenario pengujian, yaitu XGBSP-E1 (50:50), XGBSP-E2 (60:40), XGBSP-E3 (70:30), XGBSP-E4 (80:20), dan XGBSP-E5 (90:10), diperoleh bahwa model XGBSP-E5 menunjukkan performa paling optimal dengan nilai akurasi mencapai 99,68%, diikuti dengan

model XGBSP-E4 sebesar 99,61%, XGBSP-E3 sebesar 99,59%, serta model XGBSP-E1 dan XGBSP-E2 dengan nilai akurasi yang sama yaitu sekitar 99,58%. Hal ini mengindikasikan bahwa komposisi data latih 90% dan data uji 10% memberikan keseimbangan terbaik antara kompleksitas model dan kemampuan generalisasi, sehingga model mampu belajar secara mendalam tanpa kehilangan akurasi terhadap data baru.

Evaluasi berdasarkan presisi, *recall*, dan *F1-score* menunjukkan hasil yang sangat konsisten dan tinggi di seluruh kelas, dengan rata-rata nilai mendekati 1.000. Peningkatan performa ini terjadi karena penerapan SMOTE berhasil menyeimbangkan distribusi kelas minoritas dan mayoritas, sementara PCA berperan penting dalam mengurangi kompleksitas fitur dan menghilangkan redundansi antar-atribut. Kombinasi kedua metode ini membantu model mempelajari representasi fitur yang lebih bermakna dan efisien, sehingga menghasilkan klasifikasi yang lebih stabil serta mengurangi potensi *overfitting*. Secara khusus, kelas serangan seperti NS, CAAD, dan TBF yang sebelumnya sulit diklasifikasikan dengan benar menunjukkan peningkatan signifikan pada metrik *recall* dan *F1-score* setelah penerapan PCA.

Analisis lanjutan terhadap nilai *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN) pada model terbaik, yaitu XGBSP-E5, memperlihatkan bahwa model mampu meminimalkan kesalahan klasifikasi secara ekstrem. Nilai TP dan TN yang sangat tinggi menunjukkan kemampuan model dalam mendeteksi baik trafik normal maupun seluruh kategori serangan dengan tepat, sedangkan nilai FP dan FN yang mendekati nol memperlihatkan

tingkat kesalahan yang sangat rendah. Dengan demikian, model ini menunjukkan kemampuan generalisasi dan stabilitas klasifikasi yang luar biasa, bahkan terhadap variasi serangan siber yang kompleks dan jarang muncul.



Gambar 6.16 Perbandingan Performa Model XGBSP

Gambar 6.16 menunjukkan bahwa kombinasi SMOTE dan PCA dalam model XGBoost memberikan peningkatan performa dibandingkan model XGBoost tanpa *balancing* maupun tanpa reduksi dimensi. Penerapan SMOTE berhasil mengatasi masalah ketidakseimbangan kelas, sedangkan PCA membantu mengoptimalkan efisiensi komputasi dengan mengekstraksi fitur-fitur dominan yang paling relevan terhadap proses klasifikasi. Sinergi antara kedua pendekatan ini menghasilkan model yang lebih ringan secara komputasional namun tetap presisi tinggi dalam mendeteksi dan membedakan berbagai jenis serangan siber.

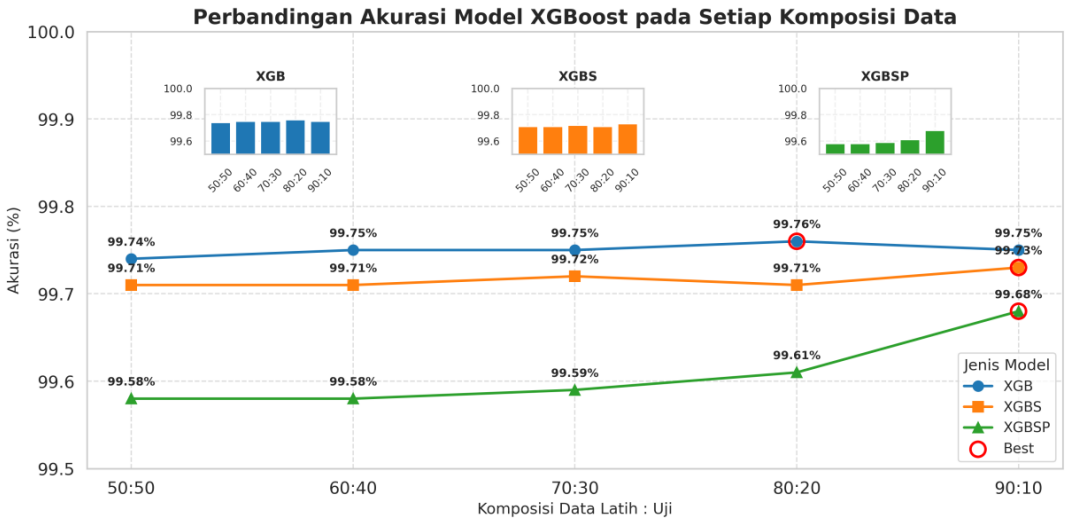
Secara keseluruhan, penelitian ini menegaskan bahwa model XGBSP-E5 merupakan model terbaik dalam mendeteksi berbagai kategori serangan siber

berbasis IoT. Model ini menunjukkan tingkat akurasi, presisi, dan stabilitas yang tinggi. Kombinasi XGBoost dengan pra-pemrosesan SMOTE dan PCA efektif untuk menghadapi tantangan ketidakseimbangan data dan kompleksitas fitur, sekaligus memberikan fondasi yang kuat untuk pengembangan sistem *Intrusion Detection System* (IDS) yang adaptif, efisien, dan siap diimplementasikan dalam lingkungan *real-time*.

BAB VII

PEMBAHASAN

Bab ini membahas hasil eksperimen yang telah dilakukan terhadap model klasifikasi serangan siber pada jaringan *Internet of Things* (IoT). Eksperimen sudah dilakukan secara bertahap terhadap lima skenario komposisi data, yang diberi label E1 hingga E5, untuk masing-masing model XGBoost. Tujuan utama dari pengujian lima skenario ini adalah untuk mengidentifikasi kombinasi data yang menghasilkan performa terbaik. Analisis ini tidak hanya menilai akurasi keseluruhan, tetapi juga metrik penting lainnya seperti presisi, *recall* dan *F1-score.*



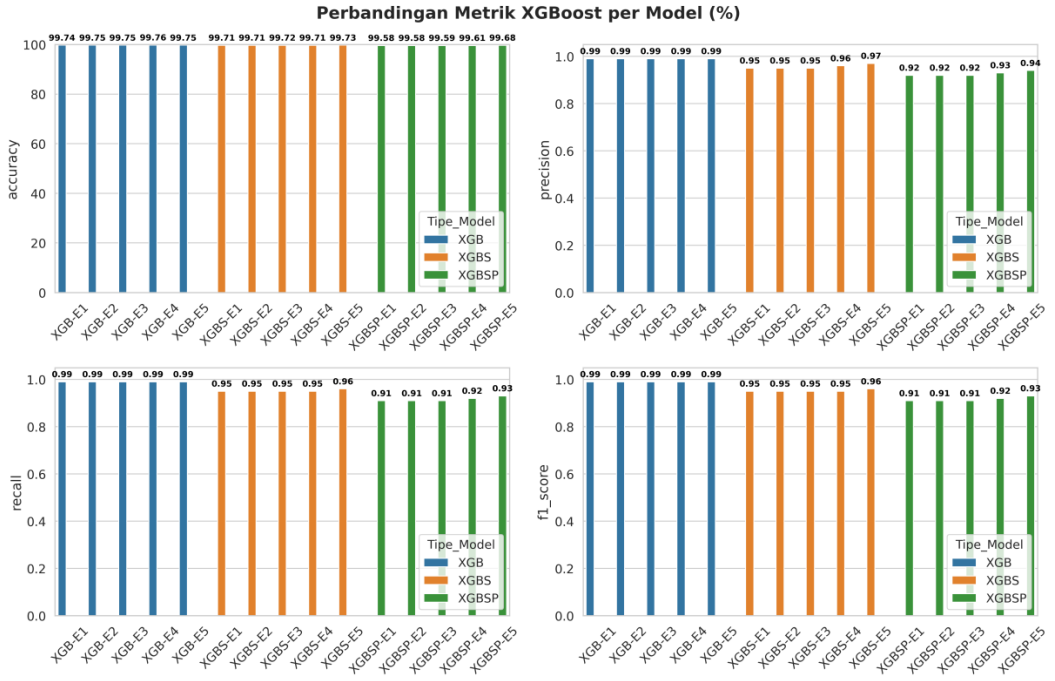
Gambar 7.1 Perbandingan Hasil Pengujian pada Setiap Eksperimen

Gambar 7.1 menunjukkan bahwa performa akurasi model XGBoost *non-balancing* mengalami peningkatan seiring dengan bertambahnya proporsi data latih, hingga mencapai nilai terbaik pada saat porsi data latih sebesar 80%. Dari lima skenario komposisi data yang diuji, XGB-E4 dengan komposisi 80:20 berhasil mencapai nilai akurasi tertinggi sebesar 99,76%. Hasil ini menunjukkan bahwa

model XGBoost *non-balancing* mampu memaksimalkan performa klasifikasi serangan siber pada IoT ketika proporsi data latih lebih besar, namun peningkatan lebih lanjut ke 90% tidak memberikan peningkatan signifikan pada akurasi.

Pada model XGBoost dengan SMOTE, pengujian terhadap lima skenario juga dilakukan untuk menangani ketidakseimbangan kelas. Hasil terbaik diperoleh pada XGBS-E5 dengan komposisi data 90:10, yang mencapai akurasi 99,73%. Peningkatan akurasi ini menunjukkan bahwa penggunaan SMOTE efektif dalam meningkatkan representasi kelas minoritas, sehingga model lebih seimbang dalam mengenali serangan minoritas yang memiliki risiko tinggi. Sementara itu, pada model XGBoost yang dikombinasikan dengan SMOTE dan PCA, nilai akurasi meningkat secara bertahap seiring bertambahnya proporsi data latih. Skenario terbaik ditemukan pada XGBSP-E5 dengan komposisi 90:10, yang menghasilkan akurasi tertinggi sebesar 99,68%. Meskipun nilai akurasi sedikit lebih rendah dibandingkan model XGB-E4 dan XGBS-E5, kombinasi SMOTE dan PCA tetap menunjukkan keunggulan dalam menangani dimensi data sekaligus memperbaiki representasi kelas minoritas pada dataset Gotham.

Gambar 7.2 menunjukkan performa model XGB diukur menggunakan metrik akurasi, presisi, *recall*, dan *F1-score*. Dari tabel terlihat bahwa kelas mayoritas seperti BEN, CAAD, dan DoS memiliki nilai akurasi dan *F1-score* sempurna 1.0, sementara kelas minoritas seperti NS dan TBF memiliki variasi performa. Hal ini menunjukkan bahwa XGBoost tanpa *balancing* mampu mengenali pola umum dengan sangat baik, tetapi sensitivitas terhadap kelas minoritas masih terbatas, misalnya TBF dengan *F1-score* 0.81 pada XGB-E1.



Gambar 7.2 Perbandingan Performa pada Setiap Eksperimen

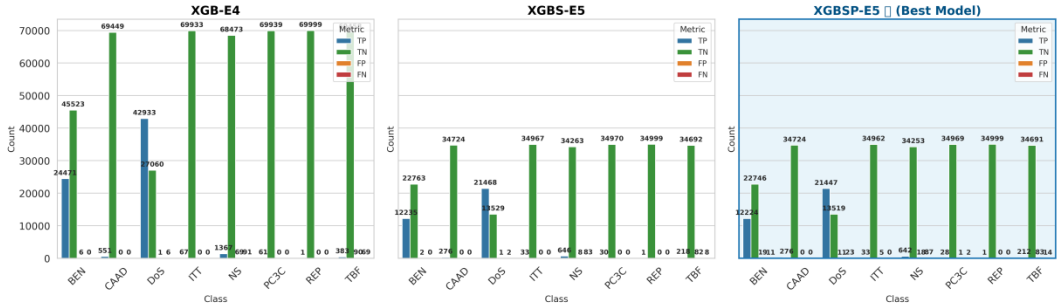
Kinerja model meningkat sedikit pada skenario XGB-E2 hingga XGB-E3. Presisi dan *recall* untuk kelas minoritas seperti NS dan TBF meningkat secara bertahap, menunjukkan bahwa model mulai menyesuaikan prediksi terhadap distribusi data minoritas. Pada XGB-E4, akurasi keseluruhan meningkat menjadi 99.76%. Kelas minoritas TBF dan NS menunjukkan peningkatan *recall dan F1-score*, misalnya TBF naik menjadi 0.83. Hal ini menandakan bahwa meski XGBoost *non-balancing* cukup baik dalam mengenali pola mayoritas, kemampuan mendeteksi serangan minoritas masih memerlukan metode tambahan agar lebih sensitif dan seimbang.

Selanjutnya, penggunaan SMOTE pada model XGBS terlihat meningkatkan performa pada kelas minoritas. Dari XGBS-E1 hingga XGBS-E5, nilai *recall* untuk kelas NS dan TBF meningkat dibandingkan model XGB. Misalnya, pada XGBS-

E5, *recall* TBF mencapai 0.83, lebih tinggi dibanding XGB-E4. Hal ini menunjukkan SMOTE mampu menyeimbangkan distribusi kelas sehingga model lebih peka terhadap serangan minoritas. Presisi dan *F1-score* pada kelas minoritas juga menunjukkan perbaikan. Kelas PC3C yang sebelumnya memiliki variasi *F1-score* kini lebih stabil, dan kelas NS menunjukkan peningkatan konsistensi. Model tetap mempertahankan akurasi tinggi pada kelas mayoritas seperti BEN, CAAD, dan DoS, sehingga integrasi SMOTE tidak menurunkan performa keseluruhan. Secara keseluruhan, XGBS-E5 menunjukkan bahwa SMOTE efektif meningkatkan sensitivitas model terhadap kelas minoritas tanpa mengorbankan performa kelas mayoritas. Nilai *F1-score* yang lebih seimbang pada seluruh kelas menunjukkan model kini lebih adaptif terhadap distribusi data yang tidak seimbang.

Integrasi PCA pada XGBSP memberikan peningkatan dalam performa keseluruhan. PCA berhasil mereduksi dimensi fitur, mempercepat proses *training*, dan meningkatkan generalisasi model. Dari XGBSP-E1 hingga XGBSP-E5, terlihat *recall* dan *F1-score* untuk kelas minoritas meningkat, misalnya NS dan TBF pada XGBSP-E5 masing-masing mencapai 0.92 dan 0.81. Selain itu, kelas mayoritas tetap mempertahankan akurasi dan presisi tinggi (misalnya BEN, CAAD, DoS), sehingga penggunaan PCA tidak menurunkan performa utama. Nilai *F1-score* menjadi lebih seimbang di seluruh kelas, menandakan model mampu mendeteksi serangan minoritas sekaligus menjaga prediksi kelas mayoritas tetap akurat. Pada XGBSP-E5, model menunjukkan performa optimal dengan akurasi keseluruhan 99.68% dan *F1-score* mendekati 1 untuk sebagian besar kelas. Hal ini menegaskan bahwa kombinasi SMOTE dan PCA mampu menghasilkan model yang lebih *robust*

dan sensitif terhadap ketidakseimbangan kelas, sekaligus efisien dalam komputasi. Perbandingan nilai TP, TN, FP, dan FN pada setiap eksperimen dapat dilihat pada Gambar 7.3.



Gambar 7.3 Perbandingan Nilai TP, TN, FP, dan FN pada Setiap Eksperimen

Model XGB-E4 menunjukkan performa yang cukup tinggi dalam mendeteksi kelas BEN, CAAD, dan DoS dengan nilai *True Positive* (TP) masing-masing sebesar 24.471, 551, dan 42.933. Nilai TP yang tinggi pada ketiga kelas ini menunjukkan kemampuan model dalam mengenali serangan dengan akurat pada sebagian besar kategori utama. Selain itu, nilai *True Negative* (TN) yang besar, seperti 69.933 untuk kelas ITT, memperlihatkan kemampuan model dalam mengidentifikasi data normal atau bukan serangan secara konsisten tanpa banyak kesalahan. Namun, model ini masih memiliki kelemahan pada kelas tertentu seperti NS dan TBF yang menunjukkan *False Positive* (FP) dan *False Negative* (FN) relatif tinggi. Nilai FP sebesar 69 dan FN sebesar 91 pada kelas NS menandakan bahwa model masih kesulitan dalam membedakan serangan nyata dan aktivitas normal pada jenis serangan tersebut. Sementara pada kelas TBF, FP sebesar 90 dan FN sebesar 69 mengindikasikan masih adanya *false alarm* dan kegagalan mendeteksi sebagian serangan sebenarnya. Secara keseluruhan, model XGB-E4 mampu memberikan hasil yang kuat pada beberapa kelas dominan, namun belum cukup

optimal dalam menekan tingkat kesalahan pada kelas minoritas atau yang memiliki karakteristik data yang mirip antar kategori. Hal ini menunjukkan bahwa model ini masih dapat ditingkatkan dengan pendekatan resampling atau transformasi fitur tambahan untuk memperbaiki generalisasi terhadap distribusi data yang tidak seimbang.

Model XGBS-E5, yang menggunakan metode *Synthetic Minority Oversampling Technique* (SMOTE), memperlihatkan peningkatan performa dalam hal stabilitas deteksi antar kelas dibandingkan XGB-E4. Nilai *True Positive* (TP) pada kelas DoS sebesar 21.468 dan BEN sebesar 12.235 menunjukkan bahwa model ini mampu mempertahankan tingkat deteksi tinggi meskipun ukuran dataset telah diimbangi. Selain itu, nilai *True Negative* (TN) yang konsisten tinggi pada seluruh kelas dan 34.970 untuk PC3C, membuktikan keandalan model dalam menghindari kesalahan identifikasi data normal sebagai serangan. Dalam hal kesalahan deteksi, nilai *False Positive* (FP) dan *False Negative* (FN) model XGBS-E5 cenderung lebih kecil dibandingkan XGB-E4. Misalnya, kelas NS hanya memiliki FP sebesar 8 dan FN sebesar 83, yang lebih baik dari model sebelumnya. Hal ini menunjukkan bahwa penerapan SMOTE berhasil memperbaiki keseimbangan data sehingga model lebih sensitif terhadap kelas minoritas tanpa meningkatkan jumlah kesalahan deteksi palsu secara signifikan. Model XGBS-E5 secara keseluruhan memperlihatkan keseimbangan yang lebih baik antara sensitivitas dan spesifisitas. Meskipun masih terdapat ruang untuk perbaikan pada beberapa kelas seperti NS dan TBF, model ini sudah menunjukkan tren positif

dalam menurunkan false alarm serta meningkatkan akurasi klasifikasi serangan pada lingkungan IoT.

Model XGBSP-E5, yang merupakan hasil kombinasi antara SMOTE dan *Principal Component Analysis* (PCA), menunjukkan kinerja paling optimal di antara ketiganya. Nilai *True Positive* (TP) yang tinggi pada kelas BEN (12.224), CAAD (276), dan DoS (21.447) menandakan kemampuan model dalam mendeteksi serangan yang sebenarnya dengan presisi tinggi. Selain itu, *True Negative* (TN) juga tetap terjaga pada kisaran tinggi di semua kelas, seperti 34.969 pada PC3C, yang memperlihatkan kemampuan model dalam menekan kesalahan deteksi negatif palsu. Walaupun terdapat sedikit peningkatan *False Positive* (FP) pada beberapa kelas seperti BEN (19) dan TBF (83), nilai *False Negative* (FN) yang relatif rendah terutama pada CAAD (0) dan ITT (0) menunjukkan bahwa model ini memiliki kemampuan deteksi serangan yang lebih konsisten dibanding dua model sebelumnya. Kombinasi SMOTE dan PCA memungkinkan model mengenali pola serangan dengan lebih baik melalui fitur yang lebih representatif, tanpa mengorbankan kestabilan hasil prediksi. Secara keseluruhan, XGBSP-E5 dapat dikategorikan sebagai model terbaik karena mampu menjaga keseimbangan optimal antara sensitivitas (TP tinggi, FN rendah) dan spesifisitas (TN tinggi, FP terkendali). Dengan performa ini, model XGBSP-E5 terbukti efektif dalam mendeteksi berbagai jenis serangan siber pada lingkungan IoT dengan tingkat kesalahan minimum dan potensi false alarm yang lebih rendah dibandingkan model lainnya.

Berdasarkan hasil evaluasi menyeluruh terhadap seluruh model yang diuji, dapat disimpulkan bahwa pendekatan XGBSP-E5 merupakan metode yang paling unggul dan layak direkomendasikan untuk tugas klasifikasi serangan siber pada sistem *Internet of Things* (IoT). Model ini menampilkan performa terbaik di antara varian lainnya (XGB-E4 dan XGBS-E5), baik dari segi akurasi, presisi, *recall* maupun *F1-score*. Peningkatan performa tersebut menunjukkan bahwa integrasi SMOTE dan PCA dalam pipeline XGBoost berhasil memperbaiki keseimbangan data serta mengoptimalkan proses reduksi dimensi fitur dengan menghasilkan model yang lebih stabil dan efisien dalam mendeteksi berbagai pola serangan kompleks pada jaringan IoT yang dinamis.

Dalam konteks penelitian ilmiah, nilai-nilai yang dianut dalam proses analisis dapat dikaitkan dengan prinsip-prinsip yang diajarkan dalam Al-Qur'an, khususnya terkait dengan ketelitian, kejujuran, dan keadilan dalam mencari kebenaran. Salah satu rujukan yang relevan adalah QS. Al-Ma'idah ayat 8 memberikan pedoman moral yang sangat relevan terhadap keadilan dalam riset dan analisis data. Allah SWT berfirman:

يٰٓاَيُّهَا الَّذِيْنَ اٰمَنُوْا كُوْنُوْا قَوّٰمِيْنَ لِلّٰهِ شُهَدَاۤءَ بِالْقِسْطِۖ وَلَا يَجْرِمَنَّكُمْ شَنَاٰنُ قَوْمٍ عَلٰٓى اَلَّا تَعْدِلُوْاۗ اِعْدِلُوْاۗ هُوَ اَقْرَبُ لِلتَّقْوٰىۖ وَاتَّقُوا اللّٰهَۗ اِنَّ اللّٰهَ خَبِيْرٌۢ بِمَا تَعْمَلُوْنَ ۝٨

"Wahai orang-orang yang beriman! Jadilah kamu orang yang selalu menegakkan kebenaran karena Allah, menjadi saksi dengan adil. Dan janganlah kebencianmu terhadap suatu kaum mendorongmu untuk berlaku tidak adil. Berlaku adillah, karena adil itu lebih dekat kepada takwa" (QS. Al-Ma'idah /5:8).

Ayat ini mengandung pesan mendalam tentang keadilan dan keseimbangan dua nilai yang sejalan dengan penerapan SMOTE *(Synthetic Minority Over-sampling Technique)* dalam penelitian ini. SMOTE berfungsi memastikan agar

setiap kelas data, baik mayoritas maupun minoritas, memiliki representasi yang seimbang dalam proses pelatihan model. Hal ini mencerminkan prinsip keadilan dalam ilmu data, yaitu memberi peluang yang sama bagi seluruh kategori agar model tidak bias terhadap salah satu pihak. Dengan demikian, penerapan SMOTE bukan sekadar pendekatan teknis, tetapi juga mencerminkan nilai spiritual bahwa keadilan dalam setiap proses menghasilkan kebenaran yang lebih dekat kepada ketakwaan.

Sementara itu, QS. Al-Hujurat ayat 13 memberikan landasan filosofis tentang kesetaraan dan penghargaan terhadap keberagaman, yang sangat relevan dengan prinsip representasi data dalam pembelajaran mesin. Ayat tersebut berbunyi:

يٰٓاَيُّهَا النَّاسُ اِنَّا خَلَقْنٰكُمْ مِّنْ ذَكَرٍ وَّاُنْثٰى وَجَعَلْنٰكُمْ شُعُوْبًا وَّقَبَاۤىِٕلَ لِتَعَارَفُوْاۚ اِنَّ اَكْرَمَكُمْ عِنْدَ اللّٰهِ اَتْقٰىكُمْۗ اِنَّ اللّٰهَ عَلِيْمٌ خَبِيْرٌ ١٣

"Wahai manusia! Sesungguhnya Kami menciptakan kamu dari seorang laki-laki dan seorang perempuan dan menjadikan kamu berbangsa-bangsa dan bersuku-suku supaya kamu saling mengenal. Sesungguhnya yang paling mulia di antara kamu di sisi Allah ialah orang yang paling bertakwa" (QS. Al-Hujurat /49:13).

Ayat ini mengajarkan pentingnya menghargai perbedaan dan memberikan perhatian yang setara terhadap setiap entitas, tanpa memandang besar kecilnya jumlah atau kekuatannya. Dalam konteks penelitian ini, prinsip tersebut sejalan dengan upaya memberi perhatian setara pada semua jenis serangan siber baik yang sering muncul maupun yang jarang terjadi. Dengan memastikan bahwa model tidak mengabaikan kelas minoritas, penelitian ini menegakkan nilai kesetaraan dalam data, sebagaimana Al-Qur'an menekankan pentingnya menghargai setiap individu

dan entitas secara adil. Hal ini menunjukkan bahwa nilai-nilai spiritual dapat diterapkan secara konkret dalam desain sistem cerdas yang etis dan berkeadilan.

Analisis hasil juga menunjukkan bahwa penambahan *Principal Component Analysis* (PCA) pada model XGBSP-E5 berperan penting dalam menyaring fitur yang tidak relevan, sehingga model menjadi lebih fokus dan efisien. Proses ini dapat dikaitkan dengan prinsip ketelitian dan kebijaksanaan yang juga ditekankan dalam ajaran Al-Qur'an. Misalnya, dalam QS. Al-Baqarah ayat 269, Allah berfirman:

يُؤْتِى الْحِكْمَةَ مَنْ يَّشَاۤءُۚ وَمَنْ يُّؤْتَ الْحِكْمَةَ فَقَدْ اُوْتِيَ خَيْرًا كَثِيْرًاۗ وَمَا يَذَّكَّرُ اِلَّآ اُولُوا الْاَلْبَابِ ﴿٢٦٩﴾

"Allah menganugerahkan hikmah (kebijaksanaan) kepada siapa yang Dia kehendaki. Barang siapa yang dianugerahi hikmah, maka sungguh ia telah dianugerahi kebaikan yang banyak. Dan tidak ada yang dapat mengambil pelajaran kecuali orang-orang yang berakal" (QS. Al-Baqarah /2:269).

Ayat ini menegaskan pentingnya kebijaksanaan dalam memilah informasi sebuah prinsip yang diterapkan secara ilmiah dalam PCA, yaitu menyaring fitur yang tidak memberi kontribusi signifikan terhadap hasil klasifikasi. Dengan menyingkirkan variabel yang tidak relevan, model menjadi lebih efisien dan akurat, mencerminkan sikap selektif dan bijak dalam mengelola data. Hal ini menunjukkan bahwa nilai hikmah dan ketelitian dalam Al-Qur'an sejalan dengan praktik ilmiah modern yang mengutamakan efisiensi, kejelasan, dan validitas dalam pengambilan keputusan berbasis data.

BAB VIII
KESIMPULAN

8.1 Kesimpulan

Berdasarkan hasil penelitian dan analisis menyeluruh terhadap penerapan algoritma XGBoost dalam proses klasifikasi serangan siber pada lingkungan *Internet of Things* (IoT), dapat disimpulkan bahwa pengembangan model yang sistematis, berbasis data, dan disertai dengan optimasi fitur memberikan peningkatan signifikan terhadap performa sistem deteksi intrusi. Penelitian ini tidak hanya menunjukkan efektivitas metode XGBoost sebagai algoritma klasifikasi yang andal, tetapi juga menegaskan pentingnya penerapan teknik praproses dan optimasi seperti SMOTE dan PCA dalam mengatasi ketidakseimbangan data serta meningkatkan generalisasi model terhadap berbagai jenis serangan.

1. Pertama, menjawab pertanyaan penelitian pertama, hasil eksperimen menunjukkan bahwa model XGBoost memiliki tingkat kinerja yang sangat baik dalam mengklasifikasikan serangan siber berbasis IoT. Model ini mampu mencapai akurasi, *presisi*, *recall*, dan *F1-score* yang tinggi pada berbagai skenario pelatihan, khususnya pada komposisi data latih yang lebih besar seperti pada model XGB-E4 dengan parameter optimal n_estimator=200, learning_rate=0.1, dan max_depth=6. Hal ini menunjukkan bahwa kemampuan XGBoost dalam melakukan pembelajaran berbasis *boosting* efektif dalam mengenali pola kompleks antar fitur jaringan IoT, sehingga mampu mendeteksi anomali dengan lebih akurat.

2. Kedua, untuk pertanyaan penelitian kedua, hasil pengujian menunjukkan bahwa penambahan SMOTE dan PCA memberikan pengaruh positif terhadap peningkatan kinerja model XGBoost pada data minoritas. SMOTE dapat mengatasi adanya ketidakseimbangan data sedangkan PCA terbukti mampu mereduksi dimensi fitur tanpa menghilangkan informasi penting yang relevan terhadap proses klasifikasi, sehingga model menjadi lebih efisien dan terhindar dari *overfitting*. Integrasi PCA juga mempercepat waktu pelatihan dan meningkatkan stabilitas performa model pada data uji. Dengan demikian, penerapan PCA sebagai tahap praproses dalam model XGBoost berperan penting dalam memperkuat kemampuan model dalam mengidentifikasi dan mengelompokkan serangan dengan presisi tinggi, terutama ketika dihadapkan pada data berdimensi besar dan tidak seimbang yang umum dijumpai pada sistem IoT.

8.2 Saran

Meskipun penelitian ini berhasil menunjukkan peningkatan performa model XGBoost melalui penerapan SMOTE dan PCA dalam klasifikasi serangan siber berbasis *Internet of Things* (IoT), terdapat beberapa keterbatasan yang perlu diperhatikan dalam interpretasi hasil dan arah pengembangan penelitian berikutnya. Salah satu keterbatasan utama terletak pada variasi dataset yang digunakan. Dataset penelitian ini masih bersumber dari satu himpunan data IoT tertentu, sehingga karakteristik pola serangan yang dianalisis belum sepenuhnya merepresentasikan kondisi ancaman siber yang beragam di lingkungan IoT nyata. Akibatnya, kemampuan generalisasi model dapat memungkinkan menurun ketika dihadapkan

pada jenis serangan baru, variasi topologi jaringan, atau perangkat IoT dengan konfigurasi yang berbeda.

Keterbatasan lainnya terdapat pada pengujian model dan aspek efisiensi komputasi. Model yang dikembangkan masih diuji dalam konteks data statis (*batch mode*), belum dalam kondisi *real-time* atau *streaming* data yang lebih merepresentasikan serangan IoT yang terjadi secara dinamis dan kontinu. Selain itu, penelitian ini belum menitikberatkan pada analisis konsumsi sumber daya komputasi yang dibutuhkan oleh kombinasi XGBoost, SMOTE, dan PCA, terutama ketika diterapkan pada perangkat IoT dengan keterbatasan daya dan memori. Dengan mempertimbangkan keterbatasan-keterbatasan tersebut, penelitian ini diharapkan dapat menjadi landasan bagi studi lanjutan yang lebih komprehensif, mencakup evaluasi *multi-dataset*, optimasi parameter adaptif, peningkatan interpretabilitas model, serta implementasi deteksi intrusi berbasis *real-time* pada ekosistem IoT yang sesungguhnya.

DAFTAR PUSTAKA

Abdullah bin Muhammad, *Tafsir Ibnu Katsir,*. Terj. M. ‘Abdul Ghoffar E.M, Cet 1,(Tt:Pustaka Imam Asy-Syafi’I,2008).

Al-Fawa’reh, M., Al-Fayoumi, M., Nashwan, S., & Fraihat, S. (2022). Cyber threat intelligence using PCA-DNN model to detect abnormal network behavior. Egyptian Informatics Journal, 23(2), 173–185. https://doi.org/10.1016/j.eij.2021.12.001

Alqaraleh, S. (2025). An Efficient Ensemble Network Anomaly Detection System for Cyber-Attacks. Engineering, Technology and Applied Science Research, 15(4), 25549–25554. https://doi.org/10.48084/etasr.11920

Arlandy, K. S., Faqih, A., & Rinaldi, A. R. (n.d.). Mengoptimalkan Kinerja Naïve Bayes Pada Ancaman Modern Dengan Menggunakan PCA Pada Data Intrusion Detection System (IDS).

Azmatul Barro, R., Sulvianti, I. D., & Afendi, M. (2013). Penerapan Synthetic Minority Oversampling Technique (Smote) Terhadap Data Tidak Seimbang Pada Pembuatan Model Komposisi Jamu (Vol. 1, Issue 1).

Balla A, Habaebi MH, Elsheikh EAA, Islam MR, Suliman FM. The Effect of Dataset Imbalance on the Performance of SCADA Intrusion Detection Systems. Sensors (Basel). 2023 Jan 9;23(2):758. doi: 10.3390/s23020758. PMID: 36679553; PMCID: PMC9865947.

Bankó, M. B., Dyszewski, S., Králová, M., Limpek, M. B., Papaioannou, M., Choudhary, G., & Dragoni, N. (2025). Advancements in Machine Learning-Based Intrusion Detection in IoT: Research Trends and Challenges. In Algorithms (Vol. 18, Issue 4). Multidisciplinary Digital Publishing Institute (MDPI). https://doi.org/10.3390/a18040209

Belarbi, O., Spyridopoulos, T., Anthi, E., Rana, O., Carnelli, P., & Khan, A. (2025). Gotham Dataset 2025: A Reproducible Large-Scale IoT Network Dataset for Intrusion Detection and Security Research [Data set]. Zenodo.

Chen, Tianqi, & Guestrin, Carlos. “XGBoost: A Scalable Tree Boosting System.” Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016.

Chen, W., & Faysal Almamy, D. (n.d.). *XGBoost-Driven Intrusion Detection Method: Integrating SMOTE-Based Class Imbalance Mitigation and Multi-Phase Learning*.

DeepStrike. (2025, Agustus). *IoT hacking statistics 2025: Threats, risks & regulations*. DeepStrike Blog. https://deepstrike.io/blog/iot-hacking-statistics.

Doghramachi, D. F., & Ameen, S. Y. (2023). Internet of Things (IoT) Security Enhancement Using XGboost Machine Learning Techniques. Computers, Materials and Continua, 77(1), 717–732. https://doi.org/10.32604/cmc.2023.041186

Firdaus, D., Sumardi, I., & Chazar, C. (2025). Deteksi Serangan Pada Jaringan Internet Of Things Medis Menggunakan Machine Learning Dengan Algoritma XGBoost. CyberSecurity dan Forensik Digital (Vol. 8, Issue 1).

Dinanti, A., & Purwadi, J. (2023). Analisis Performa Algoritma K-Nearest Neighbor dan Reduksi Dimensi Menggunakan Principal Component Analysis. Jambura Journal of Mathematics, 5(1), 155–165. https://doi.org/10.34312/jjom.v5i1.17098

Doghramachi, D. F., & Ameen, S. Y. (2023). Internet of Things (IoT) Security Enhancement Using XGboost Machine Learning Techniques. Computers, Materials and Continua, 77(1), 717–732. https://doi.org/10.32604/cmc.2023.041186

Erickson BJ, Kitamura F. Magician's Corner: 9. Performance Metrics for Machine Learning Models. Radiol Artif Intell. 2021 May 12;3(3):e200126. doi: 10.1148/ryai.2021200126. PMID: 34136815; PMCID: PMC8204137.

Gardner, C., & Lo, D. C.-T. (2021). PCA Embedded Random Forest. SoutheastCon 2021, 1–6. https://doi.org/10.1109/SoutheastCon45413.2021.9401949

Imani, M., Beikmohammadi, A., & Arabnia, H. R. (2025). Comprehensive Analysis of Random Forest and XGBoost Performance with SMOTE, ADASYN, and GNUS Under Varying Imbalance Levels. Technologies, 13(3). https://doi.org/10.3390/technologies13030088

Javed, S. H., Ahmad, M. bin, Asif, M., Almotiri, S. H., Masood, K., & al Ghamdi, M. A. (2022). An Intelligent System to Detect Advanced Persistent Threats in Industrial Internet of Things (I-IoT). Electronics (Switzerland), 11(5). https://doi.org/10.3390/electronics11050742

John Muschelli. 2020. ROC and AUC with a Binary Predictor: a Potentially Misleading Metric. J. Classif. 37, 3 (Oct 2020), 696–708. https://doi.org/10.1007/s00357-019-09345-1

Karanfilovska, M., Kochovska, T., Todorov, Z., Cholakoska, A., Jakimovski, G., & Efnusheva, D. (2022). Analysis and modelling of a ML-based NIDS for

IoT networks. Procedia Computer Science, 204, 187–195. https://doi.org/10.1016/j.procs.2022.08.023

Kayode Saheed, Y., Idris Abiodun, A., Misra, S., Kristiansen Holone, M., & Colomo-Palacios, R. (2022). A machine learning-based intrusion detection for detecting internet of things network attacks. Alexandria Engineering Journal, 61(12), 9395–9409. https://doi.org/10.1016/j.aej.2022.02.063

Khanday, Sumbl & Khanam, Deeba. (2023). THE RESEARCH DESIGN. 06. 376.

Khraisat, A., & Alazab, A. (2021). A critical review of intrusion detection systems in the internet of things: techniques, deployment strategy, validation strategy, attacks, public datasets and challenges. Cybersecurity, 4(1). https://doi.org/10.1186/s42400-021-00077-7

Kikissagbe, B. R., & Adda, M. (2024). Machine Learning-Based Intrusion Detection Methods in IoT Systems: A Comprehensive review. *Electronics*, *13*(18), 3601. https://doi.org/10.3390/electronics13183601

Kronlid, C., Brantnell, A., Elf, M., Borg, J., & Palm, K. (2024). Sociotechnical analysis of factors influencing IoT adoption in healthcare: A systematic review. Technology in Society, 78, 102675. https://doi.org/https://doi.org/10.1016/j.techsoc.2024.102675

Lori A. Dalton. 2016. Optimal ROC-Based Classification and Performance Analysis under Bayesian Uncertainty Models. IEEE/ACM Trans. Comput. Biol. Bioinformatics 13, 4 (July/August 2016), 719–729. https://doi.org/10.1109/TCBB.2015.2465966

Muschelli, John. “ROC and AUC with a Binary Predictor: a Potentially Misleading Metric.” Journal of Classification, vol. 37, no. 3, 2020, pp. 696–708. DOI: 10.1007/s00357-019-09345-1

O. Aouedi et al., "A Survey on Intelligent Internet of Things: Applications, Security, Privacy, and Future Directions," in IEEE Communications Surveys & Tutorials, vol. 27, no. 2, pp. 1238-1292, April 2025, doi: 10.1109/COMST.2024.3430368.

Sachins8201. (n.d.). Gotham [Dataset]. Kaggle. Retrieved September 23, 2025, from https://www.kaggle.com/datasets/sachins8201/gotham

Sathyanarayanan, S. (2024). Confusion Matrix-Based Performance Evaluation Metrics. African Journal of Biomedical Research, 4023–4031. https://doi.org/10.53555/ajbr.v27i4s.4345

Sibarani, F., & Chan, P. (2025). A Comparative Study of Machine Learning and Deep Learning Algorithms for Malware Detection. https://doi.org/10.32996/jcsts

Sibarani, J. N., Sirait, D. R., & Ramadhanti, S. S. (2023). Intrusion Detection Systems pada Bot-IoT Dataset Menggunakan Algoritma Machine Learning. JURNAL MASYARAKAT INFORMATIKA, 14(1), 38–52. https://doi.org/10.14710/jmasif.14.1.49721

Singh, R., Gehlot, A., & Joshi, A. (2022). Review on Intrusion Detection in Edge Based IOT. International Interdisciplinary Humanitarian Conference for Sustainability (IIHC), 788–793. https://doi.org/10.1109/iihc55949.2022.10060587

Sulton, M. S. H. W. (2025, Infrastruktur IoT Jadi Target Baru Serangan Siber 2025). CirebonKota CSIRT. Diakses dari https://csirt.cirebonkota.go.id/posts/infrastruktur-iot-jadi-target-baru-serangan-siber-2025

Thakkar, A., & Lohiya, R. (2021). A Review on Machine Learning and Deep Learning Perspectives of IDS for IoT: Recent Updates, Security Issues, and Challenges. Archives of Computational Methods in Engineering, 28(4), 3211–3243. https://doi.org/10.1007/s11831-020-09496-0

Verma, A., & Ranga, V. (2020). Machine Learning Based Intrusion Detection Systems for IoT Applications. Wireless Personal Communications, 111(4), 2287–2310. https://doi.org/10.1007/s11277-019-06986-8

Winanto, E. A., Novianto, Y., Sharipuddin, S., Wijaya, I. S., & Jusia, P. A. (2024). PENINGKATAN PERFORMA DETEKSI SERANGAN MENGGUNAKAN METODE PCA DAN RANDOM FOREST. Jurnal Teknologi Informasi Dan Ilmu Komputer, 11(2), 285–290. https://doi.org/10.25126/jtiik.20241127678