ISSN (Online) 2747-0563
Volume 6 Tahun 2026

Deteksi Intrusi ToN-IoT Menggunakan Stacking Ensemble XGBoost–LightGBM

Albi Akhsanul Hakim[1], Firza Prima Aditiawan[2*], Achmad Junaidi[3]

[1,2,3] Informatika, Universitas Pembangunan Nasional "Veteran" Jawa Timur
[1]22081010194@student.upnjatim.ac.id
[3]achmadjunaidi.if@upnjatim.ac.id
*Corresponding author email: [2]firzaprima.if@upnjatim.ac.id

***Abstrak*— Peningkatan penggunaan Internet of Things (IoT) dan Industrial Internet of Things (IIoT) pada berbagai sektor memberikan kemudahan dalam otomasi, pemantauan, dan pertukaran data, tetapi juga memperluas permukaan serangan jaringan. Kondisi tersebut membuat sistem IoT/IIoT membutuhkan mekanisme deteksi intrusi yang mampu membedakan trafik normal dan trafik serangan secara akurat. Pendekatan berbasis machine learning banyak digunakan karena mampu mempelajari pola dari data dan lebih adaptif dibandingkan pendekatan deteksi berbasis aturan. Penelitian ini mengembangkan model deteksi intrusi berbasis stacking ensemble menggunakan XGBoost dan LightGBM sebagai base learner, serta Regresi Logistik sebagai meta learner pada dataset ToN-IoT subset Network. Tugas klasifikasi difokuskan pada klasifikasi biner, yaitu normal dan attack. Tahapan penelitian meliputi pemahaman data, pra-pemrosesan fitur numerik, boolean, dan kategorikal, pembagian data menggunakan stratified split, optimasi hiperparameter menggunakan Bayesian Optimization, pembentukan fitur meta melalui mekanisme Out-of-Fold Prediction, serta evaluasi model pada data uji. Hasil optimasi menunjukkan bahwa XGBoost, LightGBM, dan Regresi Logistik sebagai meta learner memperoleh nilai PR-AUC validasi di atas 0,999. Hasil evaluasi menunjukkan bahwa ketiga model memperoleh performa yang sangat tinggi dengan nilai PR-AUC di atas 0,9999 dan Macro-F1 di atas 0,995. LightGBM menghasilkan Recall tertinggi sebesar 0,998316 dan Macro-F1 tertinggi sebesar 0,996376, sedangkan stacking ensemble memperoleh PR-AUC tertinggi sebesar 0,999984. Temuan ini menunjukkan bahwa model berbasis gradient boosting sangat efektif untuk deteksi intrusi pada dataset ToN-IoT, sementara pendekatan stacking ensemble mampu memberikan peningkatan performa tambahan melalui penggabungan prediksi beberapa model. Hasil penelitian ini menunjukkan potensi penggunaan machine learning berbasis ensemble untuk mendukung pengembangan sistem deteksi intrusi pada lingkungan IoT/IIoT.**

Kata Kunci— **Deteksi Intrusi, Machine Learning, ToN-IoT, Stacking Ensemble, XGBoost, LightGBM.**

I. PENDAHULUAN

Internet of Things (IoT) dan Industrial Internet of Things (IIoT) semakin banyak digunakan untuk mendukung otomatisasi, pemantauan, dan pertukaran data pada berbagai sektor, tetapi peningkatan konektivitas tersebut juga memperluas permukaan serangan sehingga sistem IoT/IIoT menjadi lebih rentan terhadap ancaman keamanan jaringan. Untuk mengatasi hal tersebut [1], [2], Intrusion Detection System (IDS) digunakan sebagai mekanisme pemantauan untuk mengidentifikasi indikasi serangan, akses tidak sah, maupun perilaku anomali pada lalu lintas jaringan [3]. Pendekatan berbasis *machine learning* banyak dikembangkan karena dinilai lebih adaptif dalam mempelajari pola dari data dan mendukung proses deteksi intrusi secara lebih dinamis dibandingkan pendekatan berbasis aturan semata [4].

Dalam konteks IoT/IIoT, pendekatan *machine learning* banyak digunakan karena karakteristik lalu lintas jaringan dapat direpresentasikan sebagai data tabular yang memuat atribut koneksi, protokol, jumlah paket, byte, serta fitur lain yang berkaitan dengan aktivitas jaringan [5]. Dataset ToN-IoT menyediakan data trafik jaringan yang dilengkapi dengan *ground truth* berupa label normal dan *attack*, sehingga dapat dimanfaatkan sebagai dasar pengembangan model deteksi intrusi berbasis data. Melalui proses pelatihan pada data berlabel, model *machine learning* dapat mempelajari perbedaan pola antara trafik normal dan trafik serangan, sehingga pendekatan ini relevan untuk pengembangan IDS pada lingkungan IoT/IIoT [6].

Beberapa algoritma machine learning telah banyak digunakan untuk tugas deteksi intrusi, termasuk algoritma berbasis pohon keputusan dan gradient boosting [4]. XGBoost dan LightGBM merupakan dua algoritma gradient boosting yang banyak digunakan pada data tabular karena memiliki kemampuan pemodelan yang kuat dan efisien [7]. XGBoost dikenal dengan mekanisme regularisasi dan optimasi komputasi yang baik, sedangkan LightGBM dirancang untuk mempercepat proses pelatihan dan mengurangi penggunaan memori pada data berukuran besar [8], [9]. Perbedaan karakteristik kedua algoritma tersebut menjadikannya potensial untuk dikombinasikan dalam pendekatan ensemble [10].

Salah satu pendekatan ensemble yang dapat digunakan dalam deteksi intrusi adalah *stacking ensemble*. Stacking bekerja dengan menggabungkan prediksi dari beberapa model dasar (*base learner*), kemudian menggunakan model lain sebagai *meta-learner* untuk menghasilkan prediksi akhir. Pendekatan ini memungkinkan model memanfaatkan kelebihan dari beberapa algoritma sekaligus, sehingga berpotensi meningkatkan kualitas prediksi dibandingkan penggunaan model Tunggal [11] Hal ini ditunjukkan oleh Mushtaq dkk. yang mengusulkan model stacked ensemble untuk IDS dan memperoleh akurasi sebesar 88,10% pada KDDTest+, lebih tinggi dibandingkan beberapa model tunggal terbaiknya, yaitu Random Forest sebesar 82,02% [12]. Oleh karena itu, penelitian ini menggunakan stacking ensemble dengan XGBoost dan LightGBM sebagai base learner untuk mendeteksi intrusi pada dataset ToN-IoT.

Berdasarkan uraian tersebut, penelitian ini berfokus pada deteksi intrusi menggunakan stacking ensemble XGBoost–

ISSN (Online) 2747-0563
Volume 6 Tahun 2026

LightGBM pada dataset ToN-IoT subset Network. Tugas klasifikasi yang dilakukan adalah klasifikasi biner, yaitu membedakan trafik normal dan trafik serangan. Model stacking yang diusulkan dibandingkan dengan model tunggal XGBoost dan LightGBM untuk mengevaluasi efektivitas penggabungan kedua model tersebut dalam tugas deteksi intrusi pada dataset ToN-IoT. Untuk memperoleh konfigurasi model yang lebih optimal, digunakan Bayesian Optimization sebagai metode penalaan hiperparameter [13].

Evaluasi dilakukan menggunakan PR-AUC, macro-F1, recall, dan confusion matrix untuk menilai performa deteksi serta pola kesalahan klasifikasi pada data uji *[14]*. Metrik berbasis *confusion matrix* digunakan untuk menilai kemampuan model dalam membedakan kelas normal dan serangan melalui komponen TP, TN, FP, dan FN, sedangkan PR-AUC digunakan karena mampu menggambarkan trade-off antara *precision* dan *recall [15]*.

II. KAJIAN PUSTAKA

A. *Intrusion Detection System*

Intrusion Detection System (IDS) merupakan mekanisme keamanan yang digunakan untuk memantau aktivitas pada host maupun jaringan guna mengidentifikasi indikasi serangan, akses tidak sah, atau perilaku anomali lainnya. Pada lingkungan IoT/IIoT, IDS menjadi penting karena peningkatan konektivitas juga meningkatkan risiko serangan siber, sedangkan banyak perangkat memiliki sumber daya komputasi dan fitur keamanan yang terbatas [3].

Secara umum, IDS dapat dibedakan menjadi host-based IDS (HIDS) dan network-based IDS (NIDS). HIDS berfokus pada aktivitas di tingkat host, seperti log sistem dan proses, sedangkan NIDS menganalisis lalu lintas jaringan untuk mendeteksi pola serangan [3]. Pada penelitian ini, fokus pembahasan berada pada konteks NIDS karena data yang digunakan berupa trafik jaringan dan tugas utama model adalah mengklasifikasikan aliran data menjadi normal atau serangan.

B. *Dataset ToN-IoT*

ToN-IoT merupakan dataset publik yang diperkenalkan untuk mendukung pengembangan IDS berbasis data pada lingkungan IoT dan IIoT. Dataset ini dirancang dari testbed yang merepresentasikan jaringan IoT/IIoT secara realistis dan mencakup beberapa sumber data heterogen, yaitu telemetry perangkat/sensor, log sistem operasi, dan trafik jaringan. Selain itu, dataset ini menyediakan label untuk klasifikasi biner normal-serangan serta type untuk klasifikasi multi-kelas berdasarkan jenis serangan [6].

C. *Ensemble Learning dan Stacking Ensemble*

Ensemble learning adalah pendekatan dalam machine learning yang menggabungkan prediksi dari beberapa model untuk menghasilkan keputusan akhir yang lebih baik daripada satu model tunggal. Gagasan utamanya adalah bahwa setiap model memiliki kelebihan dan kelemahan masing-masing, sehingga kombinasi beberapa model dapat meningkatkan akurasi dan stabilitas prediksi [10]. Secara umum, strategi utama dalam ensemble learning meliputi bagging, boosting, dan stacking [16], [17].

Stacking merupakan salah satu metode ensemble learning yang bekerja dalam dua tingkat. Tingkat pertama terdiri atas beberapa base learner yang dilatih pada data yang sama, lalu masing-masing menghasilkan prediksi. Prediksi-prediksi tersebut kemudian digunakan sebagai masukan bagi model tingkat kedua, yaitu meta-learner, untuk menghasilkan keputusan akhir [11]. Dengan mekanisme ini, stacking tidak hanya menggabungkan hasil dari beberapa model, tetapi juga mempelajari hubungan antar-prediksi model agar keputusan akhir menjadi lebih baik [18].

D. *Algoritma Klasifikasi yang Digunakan*

1) *XGBoost*

XGBoost (*Extreme Gradient Boosting*) adalah algoritma *ensemble* berbasis *gradient boosting decision tree* yang membangun pohon keputusan secara bertahap, di mana setiap pohon baru ditambahkan untuk memperbaiki kesalahan prediksi dari pohon sebelumnya. XGBoost mengoptimalkan fungsi objektif yang terdiri dari komponen *loss* dan regularisasi, sehingga model tidak hanya berusaha meminimalkan kesalahan prediksi, tetapi juga mengendalikan kompleksitas pohon untuk mengurangi risiko *overfitting*. Selain itu, XGBoost mendukung penanganan nilai hilang secara otomatis, *tree pruning*, serta optimasi komputasi seperti pemrosesan paralel pada proses pencarian split. Karakteristik tersebut membuat XGBoost banyak digunakan pada data tabular karena mampu menghasilkan performa prediksi yang tinggi dengan proses pelatihan yang tetap efisien [8].

2) *LightGBM*

LightGBM juga merupakan algoritma *gradient boosting decision tree* yang dirancang dengan penekanan pada efisiensi komputasi dan penggunaan memori. Berbeda dari pendekatan pertumbuhan pohon *level-wise* yang membangun pohon secara seimbang pada setiap level, LightGBM menggunakan strategi *leaf-wise growth*, yaitu memilih daun dengan penurunan *loss* terbesar untuk dikembangkan lebih lanjut. Strategi ini memungkinkan model mencapai performa yang baik dengan jumlah iterasi yang lebih efisien. Selain itu, LightGBM menerapkan teknik seperti *Gradient-based One-Side Sampling* (GOSS) dan *Exclusive Feature Bundling* (EFB) untuk mempercepat pelatihan dan mengurangi dimensi fitur, sehingga banyak digunakan pada data tabular berukuran besar atau berdimensi tinggi [9].

3) *Regresi Logistik*

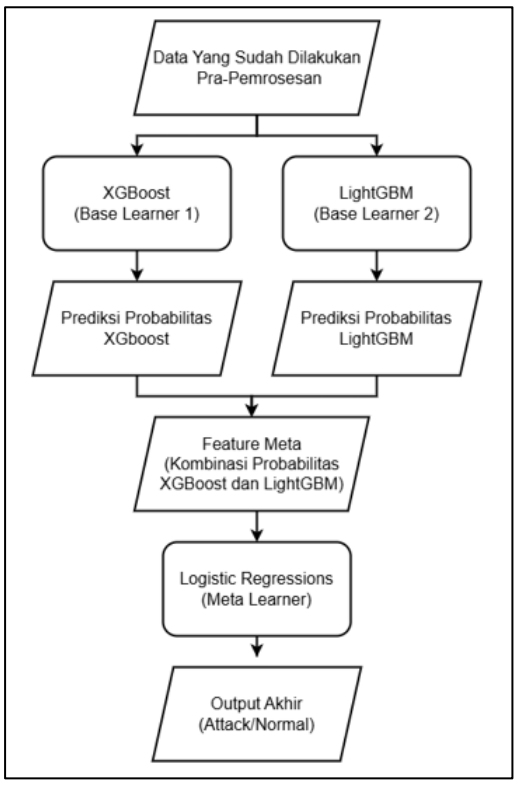
Regresi logistik merupakan metode klasifikasi yang digunakan untuk memodelkan probabilitas suatu observasi termasuk ke dalam kelas tertentu. Secara umum, metode ini bekerja dengan membentuk kombinasi linier dari fitur-fitur masukan, kemudian mentransformasikan nilai tersebut menggunakan fungsi logistik (*sigmoid*) sehingga menghasilkan probabilitas pada rentang 0 sampai 1. Nilai probabilitas tersebut selanjutnya digunakan untuk menentukan kelas akhir berdasarkan ambang keputusan tertentu [19].

ISSN (Online) 2747-0563
Volume 6 Tahun 2026

Dalam penelitian ini, Regresi Logistik digunakan sebagai meta learner karena model ini dapat menggabungkan probabilitas keluaran dari XGBoost dan LightGBM menjadi keputusan akhir yang sederhana dan efisien.

III. METODOLOGI PENELITIAN

A. Arsitektur Model Usulan

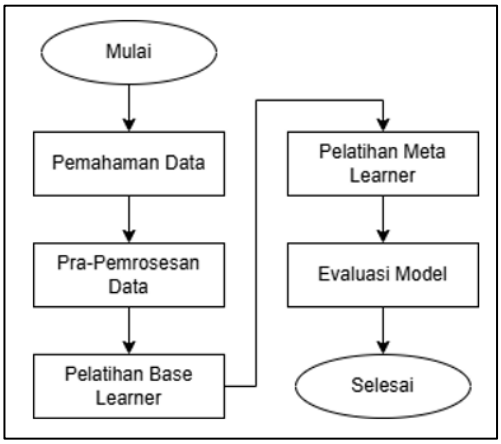


Gbr. 1. Diagram Arsitektur Model Stacking

Penelitian ini mengusulkan model stacking ensemble yang terdiri atas dua base learner, yaitu XGBoost dan LightGBM, serta Logistic Regression sebagai meta learner. Data hasil pra-pemrosesan digunakan sebagai input bagi kedua base learner untuk menghasilkan prediksi probabilitas. Probabilitas tersebut kemudian dikombinasikan menjadi feature meta yang digunakan oleh Logistic Regression untuk menghasilkan prediksi akhir berupa normal atau attack.

B. Tahapan Penelitian

Tahapan awal metodologi penelitian ini dimulai dengan pemahaman data, yaitu mengidentifikasi karakteristik dataset ToN-IoT subset Network, struktur fitur, serta label yang digunakan untuk klasifikasi biner normal dan serangan. Setelah itu, dilakukan pra-pemrosesan data yang mencakup pembersihan data, penanganan nilai tidak tersedia, pemilihan fitur yang digunakan, serta transformasi fitur numerik dan kategorikal agar sesuai dengan kebutuhan model machine learning. Data hasil pra-pemrosesan kemudian digunakan pada tahap pelatihan base learner, yaitu XGBoost dan LightGBM, dengan konfigurasi hiperparameter yang diperoleh melalui proses optimasi. Selanjutnya, prediksi dari base learner dimanfaatkan untuk membentuk fitur meta yang digunakan dalam pelatihan meta learner, yaitu Regresi Logistik, sehingga menghasilkan model stacking ensemble.



Gbr. 2. Diagram Alur Penelitian

Tahap akhir penelitian adalah evaluasi model dengan menggunakan metrik kinerja seperti accuracy, precision, recall, F1-score, macro-F1, PR-AUC, dan confusion matrix untuk menilai kemampuan model dalam membedakan trafik normal dan serangan

C. Pemahaman Data

Penelitian ini menggunakan dataset ToN-IoT subset Network yang berisi data lalu lintas jaringan dalam bentuk tabular. Dataset ini memuat fitur yang merepresentasikan aktivitas koneksi, statistik paket dan byte, serta atribut protokol seperti DNS, SSL, HTTP, dan indikator anomali. Dataset ToN-IoT dipilih karena dirancang untuk mendukung penelitian deteksi intrusi pada lingkungan IoT dan IIoT.

Tugas klasifikasi pada penelitian ini difokuskan pada klasifikasi biner menggunakan kolom label, yaitu normal dan serangan. Kelas normal merepresentasikan trafik jaringan yang tidak mengandung serangan, sedangkan kelas *attack* merepresentasikan trafik yang terindikasi sebagai serangan. Beberapa fitur seperti timestamp, alamat IP, port, serta atribut teks tertentu tidak digunakan sebagai masukan model karena bersifat identifier atau terlalu spesifik terhadap lingkungan data.

D. Pra-Pemrosesan Data

Tahap pra-pemrosesan dilakukan untuk menyiapkan data sebelum proses pelatihan model. Langkah awal yang dilakukan meliputi standardisasi nama kolom, pembersihan nilai bertipe teks, normalisasi nilai kosong, serta penanganan simbol - sebagai nilai tidak tersedia pada fitur tertentu. Selain itu, dilakukan penghapusan fitur berdasarkan aturan, terutama pada fitur yang bersifat *identifier* atau terlalu spesifik terhadap lingkungan data, seperti *timestamp*, alamat IP, port, dan atribut teks tertentu. Penghapusan fitur tersebut bertujuan mengurangi risiko model mempelajari pola yang terlalu spesifik pada data latih sehingga dapat memicu *overfitting*. Setelah proses pembersihan awal, data yang digunakan berjumlah 190.474 *record*.

ISSN (Online) 2747-0563
Volume 6 Tahun 2026

Fitur yang digunakan kemudian dikelompokkan menjadi fitur numerik, boolean, dan kategorikal. Pada fitur numerik, nilai dikonversi ke bentuk numerik dan nilai yang hilang ditangani menggunakan imputasi median. Pada fitur boolean, nilai true/false atau 1/0 dinormalisasi menjadi nilai biner, kemudian nilai kosong diisi menggunakan nilai yang paling sering muncul. Sementara itu, pada fitur kategorikal, nilai kosong diisi dengan kategori khusus dan selanjutnya dilakukan *one-hot encoding*. Setelah seluruh tahapan pra-pemrosesan selesai, data dibagi menjadi data latih dan data uji menggunakan *stratified split* dengan proporsi data uji sebesar 20%.

E. Pelatihan Base Learner

Tahap pelatihan base learner menggunakan dua algoritma gradient boosting, yaitu XGBoost dan LightGBM. Kedua model tersebut digunakan sebagai model tunggal pembanding sekaligus sebagai base learner pada skema stacking ensemble. Sebelum pelatihan final, hiperparameter masing-masing model dioptimasi menggunakan Bayesian Optimization dengan Optuna dan Stratified K-Fold Cross-Validation.

Pada XGBoost, parameter yang dioptimasi meliputi n_estimators, max_depth, learning_rate, subsample, colsample_bytree, gamma, dan min_child_weight. Pada LightGBM, parameter yang dioptimasi meliputi n_estimators, num_leaves, max_depth, learning_rate, feature_fraction, bagging_fraction, min_data_in_leaf, dan lambda_l2. Nilai rata-rata PR-AUC dari validasi silang digunakan sebagai fungsi objektif, kemudian konfigurasi terbaik digunakan untuk melatih ulang model pada seluruh data latih.

F. Pelatihan Meta Learner

Pelatihan meta learner dilakukan untuk membangun model stacking ensemble. Meta learner yang digunakan adalah Regresi Logistik karena sederhana, efisien, dan sesuai untuk menggabungkan prediksi probabilitas dari base learner. Fitur meta dibentuk menggunakan mekanisme OOF (Out-of-Fold Prediction), yaitu XGBoost dan LightGBM dilatih pada train-fold, kemudian menghasilkan prediksi probabilitas pada validation-fold. Mekanisme OOF digunakan agar meta learner dilatih menggunakan prediksi yang dihasilkan dari data yang tidak digunakan dalam pelatihan base learner pada fold terkait. Prediksi probabilitas dari XGBoost dan LightGBM digabungkan menjadi fitur meta yang digunakan untuk melatih Regresi Logistik. Hiperparameter utama Regresi Logistik, yaitu nilai C, dioptimasi menggunakan Optuna dengan metrik PR-AUC. Setelah parameter terbaik diperoleh, meta learner dilatih ulang menggunakan seluruh fitur meta dari data latih, kemudian digunakan untuk menghasilkan prediksi akhir pada data uji.

G. Evaluasi Model

Evaluasi dilakukan pada data uji yang tidak digunakan dalam proses pelatihan maupun optimasi. Model yang dievaluasi meliputi XGBoost, LightGBM, dan stacking ensemble XGBoost–LightGBM. Prediksi probabilitas dikonversi menjadi label kelas menggunakan threshold 0,5, yaitu probabilitas $\geq$ 0,5 diklasifikasikan sebagai serangan dan probabilitas < 0,5 diklasifikasikan sebagai normal.

Metrik evaluasi yang digunakan meliputi PR-AUC, macro-F1, recall, classification report, dan confusion matrix. PR-AUC digunakan karena relevan untuk mengevaluasi klasifikasi pada data yang tidak seimbang, sedangkan macro-F1 dan recall digunakan untuk menilai keseimbangan performa serta kemampuan model dalam mendeteksi serangan. Hasil evaluasi dari ketiga model dibandingkan untuk mengetahui apakah stacking ensemble mampu memberikan performa yang lebih baik dibandingkan model tunggal.

IV. HASIL DAN PEMBAHASAN

A. *Hasil Pelatihan Base Learner*

Pelatihan base learner dilakukan menggunakan dua algoritma, yaitu XGBoost dan LightGBM. Sebelum model dilatih secara final, masing-masing model terlebih dahulu melalui proses optimasi hiperparameter menggunakan Bayesian Optimization dengan Optuna. Metrik PR-AUC digunakan sebagai fungsi objektif karena sesuai untuk mengevaluasi model pada data dengan distribusi kelas yang tidak seimbang. Data yang digunakan setelah proses pembersihan berjumlah 190.474 record, kemudian dibagi menjadi data latih sebanyak 152.379 record dan data uji sebanyak 38.095 record. Distribusi kelas pada data latih dan data uji tetap dipertahankan melalui stratified split, dengan proporsi kelas normal sebesar 22,07% dan kelas *attack* sebesar 77,93%.

Proses optimasi hiperparameter dilakukan menggunakan Stratified K-Fold Cross-Validation pada data latih untuk memperoleh konfigurasi terbaik dari masing-masing base learner. Setiap model diberikan kuota maksimum 50 trial pencarian hiperparameter, proses optimasi dihentikan lebih awal apabila tidak terdapat peningkatan nilai PR-AUC terbaik setelah 10 trial berturut-turut. Hasil optimasi menunjukkan bahwa XGBoost memperoleh nilai PR-AUC validasi silang sebesar 0,999968, sedangkan LightGBM memperoleh nilai PR-AUC validasi silang sebesar 0,999978. Ringkasan hasil optimasi base learner ditunjukkan pada Tabel I.

TABEL I
HASIL OPTIMASI BASE LEARNER

Base Learner	**Hiperparameter yang Dioptimasi**	**Trial Tereksekusi**	**Skor PR-AUC**
XGBoost	n_estimators, max_depth, learning_rate, subsample, colsample_bytree, gamma, min_child_weight	12	0,999968
LightGBM	n_estimators, num_leaves, max_depth, learning_rate, feature_fraction, bagging_fraction, min_data_in_leaf, lambda_l2	14	0,999978

ISSN (Online) 2747-0563
Volume 6 Tahun 2026

Berdasarkan Tabel I, kedua base learner memperoleh nilai PR-AUC validasi silang yang sangat tinggi. LightGBM menghasilkan nilai PR-AUC CV sedikit lebih tinggi dibandingkan XGBoost, meskipun selisih keduanya sangat kecil. Hasil ini menunjukkan bahwa XGBoost dan LightGBM memiliki kemampuan awal yang baik sebagai base learner dalam membedakan trafik normal dan serangan. Setelah konfigurasi terbaik diperoleh, XGBoost dan LightGBM dilatih ulang pada seluruh data latih untuk digunakan pada tahap pembentukan fitur meta dalam model stacking ensemble.

B. *Hasil Pelatihan Meta Learner*

Setelah *base learner* dilatih, fitur meta dibentuk menggunakan mekanisme *Out-of-Fold Prediction*. Data latih dibagi dengan *Stratified K-Fold* sebanyak lima *fold*, kemudian XGBoost dan LightGBM dilatih pada *train-fold* dan digunakan untuk menghasilkan prediksi probabilitas pada *validation-fold*. Prediksi dari kedua model tersebut digabungkan menjadi dua fitur meta, yaitu pred_xgb dan pred_lgbm.

Meta learner yang digunakan adalah Regresi Logistik. Parameter C dioptimasi menggunakan Bayesian Optimization dengan PR-AUC sebagai fungsi objektif, dan memperoleh nilai PR-AUC validasi silang sebesar 0,999972. Setelah parameter terbaik diperoleh, Regresi Logistik dilatih ulang menggunakan seluruh fitur meta dari data latih. Ringkasan hasil optimasi meta learner ditunjukkan pada Tabel II.

TABEL II
HASIL OPTIMASI META LEARNER

Meta Learner	**Hiperparameter yang Dioptimasi**	**Trial Tereksekusi**	**Skor PR-AUC**
Regresi Logistik	C	19	0,999972

C. *Evaluasi Model Stacking*

Evaluasi akhir dilakukan pada data uji menggunakan model stacking ensemble. Pada tahap ini, data uji terlebih dahulu diprediksi oleh XGBoost dan LightGBM untuk menghasilkan fitur meta berupa probabilitas prediksi. Fitur meta tersebut kemudian digunakan oleh Regresi Logistik untuk menghasilkan prediksi akhir. Hasil evaluasi model stacking ditunjukkan pada Tabel III.

TABEL III
HASIL EVALUASI MODEL STACKING ENSEMBLE

Model	**Metrik**	**Nilai**
XGBoost	Recall	0,997373
	Macro-F1	0,995352
	PR-AUC	0,999982
LightGBM	Recall	0,998316
	Macro-F1	0,996376
	PR-AUC	0,999985
Stacking	Recall	0,997103
	Macro-F1	0,995202
	PR-AUC	0,999984

Berdasarkan Tabel III, ketiga model menunjukkan performa yang sangat tinggi pada dataset ToN-IoT subset Network dengan nilai PR-AUC di atas 0,9999 dan nilai Macro-F1 di atas 0,995. Hasil ini menunjukkan bahwa XGBoost, LightGBM, dan stacking ensemble sama-sama mampu membedakan trafik normal dan serangan dengan baik pada data uji.

LightGBM memperoleh nilai Recall tertinggi sebesar 0,998316, yang menunjukkan kemampuan terbaik dalam mendeteksi serangan dibandingkan model lainnya. Selain itu, LightGBM juga menghasilkan nilai Macro-F1 tertinggi sebesar 0,996376, yang mengindikasikan keseimbangan performa yang baik pada kedua kelas. Di sisi lain, stacking ensemble memperoleh nilai PR-AUC tertinggi sebesar 0,999984, yang menunjukkan kemampuan yang sangat baik dalam mempertahankan trade-off antara precision dan recall pada berbagai nilai threshold.

Meskipun demikian, perbedaan performa antar model relatif kecil. Temuan ini menunjukkan bahwa XGBoost dan LightGBM sebagai model berbasis gradient boosting telah memiliki kemampuan deteksi yang sangat kuat pada dataset ToN-IoT, sementara pendekatan stacking ensemble mampu memberikan peningkatan performa tambahan pada metrik tertentu melalui penggabungan prediksi kedua base learner.

V. KESIMPULAN

Penelitian ini berhasil merancang dan mengevaluasi model deteksi intrusi berbasis machine learning menggunakan XGBoost, LightGBM, dan stacking ensemble XGBoost–LightGBM pada dataset ToN-IoT subset Network. Hasil pengujian menunjukkan bahwa ketiga model memperoleh performa yang sangat tinggi dengan nilai PR-AUC di atas 0,9999 dan Macro-F1 di atas 0,995. LightGBM menghasilkan nilai Recall tertinggi sebesar 0,998316 dan Macro-F1 tertinggi sebesar 0,996376, sedangkan stacking ensemble memperoleh nilai PR-AUC tertinggi sebesar 0,999984. Hasil tersebut menunjukkan bahwa model berbasis gradient boosting memiliki kemampuan yang sangat baik dalam membedakan trafik normal dan serangan pada dataset ToN-IoT, sementara pendekatan stacking ensemble mampu memberikan peningkatan performa tambahan melalui penggabungan prediksi kedua base learner.

Secara praktis, rancangan model ini dapat diposisikan sebagai pendekatan berbasis machine learning yang potensial untuk mendukung sistem deteksi intrusi pada lingkungan IoT/IIoT. Namun, penelitian ini memiliki beberapa keterbatasan. Evaluasi hanya dilakukan pada dataset ToN-IoT subset Network dengan skema stratified train-test split dan berfokus pada klasifikasi biner normal dan attack. Selain itu, analisis performa masih dilakukan pada tingkat agregat sehingga belum mengevaluasi perilaku model pada masing-masing jenis serangan maupun kondisi data yang berbeda dari distribusi pelatihan.

Sebagai pengembangan lebih lanjut, penelitian dapat diperluas dengan mengevaluasi kemampuan generalisasi model terhadap jenis serangan baru, perubahan distribusi data, maupun perangkat atau lingkungan jaringan yang belum pernah diamati saat pelatihan. Selain itu, evaluasi pada skenario inferensi yang lebih mendekati kondisi operasional jaringan nyata juga dapat

ISSN (Online) 2747-0563
Volume 6 Tahun 2026

dilakukan untuk memperoleh gambaran yang lebih komprehensif mengenai performa model.

UCAPAN TERIMA KASIH

Penulis menyampaikan apresiasi kepada dosen pembimbing dan rekan-rekan yang telah memberikan dukungan selama proses perumusan metodologi, pengolahan dataset ToN-IoT, pelatihan model, dan analisis hasil evaluasi deteksi intrusi berbasis *stacking ensemble* XGBoost–LightGBM. Dukungan dan masukan dari berbagai pihak menjadi bagian penting dalam penyelesaian penelitian ini. Hasil penelitian ini diharapkan dapat memberikan kontribusi bagi pengembangan sistem deteksi intrusi berbasis machine learning, khususnya pada lingkungan IoT/IIoT, serta menjadi referensi awal bagi penelitian lanjutan yang membahas pengembangan model deteksi intrusi, pemanfaatan metode *ensemble learning*, dan evaluasi model pada dataset keamanan jaringan yang lebih beragam.

REFERENSI

[1] E. Sisinni, A. Saifullah, S. Han, U. Jennehag, and M. Gidlund, "Industrial internet of things: Challenges, opportunities, and directions," IEEE Trans. Industr. Inform., vol. 14, no. 11, pp. 4724–4734, Nov. 2018, doi: 10.1109/TII.2018.2852491.

[2] L. Diana, P. Dini, and D. Paolini, "Overview on Intrusion Detection Systems for Computers Networking Security," Computers, vol. 14, no. 3, p. 87, Mar. 2025, doi: 10.3390/computers14030087.

[3] M. M. Rahman, S. Al Shakil, and M. R. Mustakim, "A survey on intrusion detection system in IoT networks," Cyber Security and Applications, vol. 3, Dec. 2024, doi: 10.1016/j.csa.2024.100082.

[4] A. Thakkar and R. Lohiya, "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, vol. 28, no. 4, pp. 3211–3243, Jun. 2021, doi: 10.1007/s11831-020-09496-0.

[5] A. A. Alsulami, Q. Abu Al-Haija, A. Tayeb, and A. Alqahtani, "An Intrusion Detection and Classification System for IoT Traffic with Improved Data Engineering," Applied Sciences (Switzerland), vol. 12, no. 23, Dec. 2022, doi: 10.3390/app122312336.

[6] A. Alsaedi, N. Moustafa, Z. Tari, A. Mahmood, and Adna N Anwar, "TON-IoT telemetry dataset: A new generation dataset of IoT and IIoT for data-driven intrusion detection systems," IEEE Access, vol. 8, pp. 165130–165150, 2020, doi: 10.1109/ACCESS.2020.3022862.

[7] C. Bentéjac, A. Csörgő, and G. Martínez-Muñoz, "A comparative analysis of gradient boosting algorithms," Artif. Intell. Rev., vol. 54, no. 3, pp. 1937–1967, Mar. 2021, doi: 10.1007/s10462-020-09896-5.

[8] T. Chen and C. Guestrin, "XGBoost," in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, NY, USA: ACM, Aug. 2016, pp. 785–794. doi: 10.1145/2939672.2939785.

[9] G. Ke et al., "LightGBM: A Highly Efficient Gradient Boosting Decision Tree," in Advances in Neural Information Processing Systems 30 (NeurIPS 2017), 2017.

[10] T. G. Dietterich, "Ensemble Methods in Machine Learning," 2000, pp. 1–15. doi: 10.1007/3-540-45014-9_1.

[11] D. H. Wolpert, "Stacked Generalization," Neural Networks, vol. 5, no. 2, pp. 241–259, 1992, doi: 10.1016/S0893-6080(05)80023-1.

[12] E. Mushtaq, A. Zameer, and A. Khan, "A two-stage stacked ensemble intrusion detection system using five base classifiers and MLP with optimal feature selection," Microprocess. Microsyst., vol. 94, p. 104660, Oct. 2022, doi: 10.1016/j.micpro.2022.104660.

[13] J. Snoek, H. Larochelle, and R. P. Adams, "Practical Bayesian Optimization of Machine Learning Algorithms," in Advances in Neural Information Processing Systems 25 (NIPS 2012), F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds., Red Hook, NY: Curran Associates, Inc., Aug. 2012, pp. 2951–2959. Accessed: Jan. 26, 2026. [Online]. Available: https://papers.nips.cc/paper_files/paper/2012/hash/05311655a15b75fab86956663e1819cd-Abstract.html

[14] Z. Dai et al., "An intrusion detection model to detect zero-day attacks in unseen data using machine learning," PLoS One, vol. 19, no. 9, Sep. 2024, doi: 10.1371/journal.pone.0308469.

[15] T. Saito and M. Rehmsmeier, "The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets," PLoS One, vol. 10, no. 3, Mar. 2015, doi: 10.1371/journal.pone.0118432.

[16] L. Breiman, "Bagging predictors," Mach. Learn., vol. 24, no. 2, pp. 123–140, Aug. 1996, doi: 10.1007/BF00058655.

[17] L. Breiman, "Random Forests," Mach. Learn., vol. 45, no. 1, pp. 5–32, Oct. 2001, doi: 10.1023/A:1010933404324.

[18] M. Ali et al., "Effective network intrusion detection using stacking-based ensemble approach," Int. J. Inf. Secur., vol. 22, no. 6, pp. 1781–1798, Dec. 2023, doi: 10.1007/s10207-023-00718-7.

[19] D. W. . Hosmer, Stanley. Lemeshow, and R. X. . Sturdivant, Applied logistic regression. Wiley, 2013.