Vol 8 No 2 (2025):
JURNAL PENDIDIKAN TEKNOLOGI INFORMASI
(JUKANTI) EDISI NOPEMBER 2025

DETEKSI ANOMALI WEBSERVER BERBASIS HYBRID ISOLATION FOREST DAN TRANSFORMER DENGAN WEIGHTED FUSION

Ardian Yusuf Wicaksono[a], Rizky Fenaldo Maulana[b], Irvan Surya Nugraha[c], dan Yuandytha Fitria Ade Putri Sujiana[d]

[a,b,c,d]*Program Studi Informatika, Universitas Telkom, Kampus Surabaya, Surabaya 60231, Jawa Timur*

[a]ardianyw@telkomuniversity.ac.id, [b]rizkyfenaldo@telkomuniversity.ac.id,
[c]irvansn@student.telkomuniversity.ac.id, [d]yuandythaadeputri@student.telkomuniversity.ac.id

ABSTRAK

Penelitian ini mengusulkan sistem deteksi anomali pada webserver dengan mengombinasikan metode Isolation Forest dan Transformer melalui pendekatan *Weighted Fusion*. Data berupa metrik *time-series* dari layanan Nginx meliputi penggunaan CPU, memori, dan aktivitas koneksi diproses melalui normalisasi dan pembentukan *window* sebelum pelatihan. Isolation Forest dimanfaatkan untuk mendeteksi anomali berbasis nilai, sedangkan Transformer menangkap pola temporal yang kompleks guna mengidentifikasi anomali kontekstual. Evaluasi menggunakan *5-fold cross-validation* menunjukkan bahwa pendekatan hybrid mencapai kinerja rata-rata F1-score sekitar 77.92% ± 0.63% dan *Average Precision* (AP) sekitar 84.77% ± 0.69%, lebih baik dibanding penggunaan model tunggal. Stabilitas kinerja memungkinkan sistem mempertahankan keseimbangan antara *precision* dan *recall* pada data yang tidak seimbang. Secara praktis, metode ini berpotensi meningkatkan efektivitas pemantauan operasional dan mendukung mitigasi dini terhadap insiden keamanan siber seperti web defacement. Saat ini, sistem bekerja menggunakan pendekatan *offline learning*, sehingga model perlu dilatih ulang ketika terdapat perubahan pola data. Pengembangan lanjutan dapat diarahkan pada penerapan *online learning* agar deteksi lebih adaptif terhadap dinamika trafik web secara real-time, serta integrasi sumber data tambahan untuk meningkatkan ketahanan sistem. Dengan demikian, penelitian ini berkontribusi dalam merancang dan mengevaluasi kerangka hybrid berbasis *Weighted Fusion* yang efektif untuk deteksi anomali pada webserver.

Kata kunci *: Deteksi Anomali, Isolation Forest, Transformer, Time-Series, Weighted Fusion*

ABSTRACT

This study proposes an anomaly detection system for web servers by combining Isolation Forest and Transformer models through a Weighted Fusion approach. Time-series metrics collected from an Nginx-based service including CPU usage, memory utilization, and connection activity were normalized and formatted into windowed sequences prior to model training. Isolation Forest was employed to detect point anomalies, while the Transformer model captured complex temporal patterns to identify contextual anomalies. Evaluation using 5-fold cross-validation shows that the hybrid model achieves an average F1-score of approximately 77.92% ± 0.63% *and an Average Precision (AP) of around* 84.77% ± 0.69%, *outperforming each standalone model. This balanced performance demonstrates improved stability between precision and recall under imbalanced data conditions. Practically, the proposed method can enhance operational monitoring effectiveness and support early mitigation of cybersecurity incidents, such as web defacement. Currently, the system operates under an offline learning scheme, requiring model retraining when data patterns shift. Future work may explore online learning to enable adaptive real-time detection, as well as integration of additional data sources to improve robustness. Overall, this research contributes an effective hybrid framework with Weighted Fusion for anomaly detection on web servers.*

Keywords: *Anomaly Detection, Isolation Forest, Time-Series, Transformer, Weighted Fusion*

Wicaksono, A. Y., Maulana, R. F., Nugraha, I. S., & Sujiana, Y. F. A. P. (2025). DETEKSI ANOMALI WEBSERVER BERBASIS HYBRID ISOLATION FOREST DAN TRANSFORMER DENGAN WEIGHTED FUSION. Jurnal Pendidikan Teknologi Informasi (JUKANTI), 8(2), 302–317.
https://doi.org/10.37792/jukanti.v8i2.1904

1. PENDAHULUAN

Kestabilan dan kinerja infrastruktur TI memegang peran penting dalam menjaga keandalan layanan digital. Gangguan kecil pada sistem dapat berdampak signifikan terhadap ketersediaan, keamanan data, dan kepuasan pengguna. Namun, deteksi anomali secara tepat waktu masih menjadi tantangan, terutama karena volume data yang besar serta dinamika perilaku sistem yang sulit ditangani dengan metode tradisional seperti monitoring manual maupun static thresholding.

Contoh anomali penggunaan pada webserver adalah serangan eksternal yang memicu banyak kasus web defacement, yaitu perubahan tampilan atau konten situs untuk tujuan vandalisme, propaganda, maupun monetisasi misalnya penyisipan konten judi online (judol). Di Indonesia, sindikat peretas diketahui menargetkan situs pemerintah dan kampus untuk kemudian disewakan atau dialihkan menjadi halaman judi online [1]. Berbagai laporan penegakan hukum juga menunjukkan ratusan situs telah disusupi kelompok ini [2], [3], [4], yang menyebabkan kerusakan reputasi, potensi kebocoran data, serta biaya pemulihan yang tidak sedikit. Oleh karena itu, diperlukan sistem deteksi anomali yang mampu mengidentifikasi indikasi serangan dini.

Salah satu metode *unsupervised anomaly detection* yang kerap digunakan adalah Isolation Forest. Algoritma ini mengisolasi setiap titik data melalui konstruksi pohon acak dan mengukur tingkat keanehan berdasarkan kedalaman rata-rata pemisahan; semakin cepat terisolasi, semakin besar kemungkinan titik tersebut merupakan anomali. Keunggulan metode ini antara lain efisiensi komputasi dan tidak memerlukan data berlabel [5], [6]. Penelitian sebelumnya telah mengaplikasikan pendekatan ini pada berbagai domain seperti log server [7], data pengukuran [8], pengembangan struktur optimasi seperti OptIForest [9], serta adaptasi untuk data streaming [10]. Lebih lanjut, kerangka umum untuk fusi *one-class classifier* juga telah ditawarkan sebagai pendekatan hybrid yang relevan dalam deteksi anomali [11].

Namun, Isolation Forest memiliki keterbatasan dalam mengenali pola temporal yang kompleks pada data *time-series*, sehingga kurang efektif dalam mendeteksi anomali berbasis konteks. Untuk mengatasi keterbatasan tersebut, arsitektur Transformer muncul sebagai solusi yang mampu memproses urutan data melalui mekanisme *self-attention*, memungkinkan model menangkap hubungan antar-waktu (*temporal dependency*). Transformer telah banyak digunakan dalam mendeteksi anomali pada *system log*, IoT multivariat, manufaktur cerdas, dan lingkungan cloud [12], [13], [14], [15], [16], [17]. Selain itu, pendekatan hybrid juga menunjukkan bahwa kombinasi beberapa detektor dapat meningkatkan *robustness* dan interpretabilitas hasil deteksi [18].

Sejalan dengan itu, penelitian terkini menyoroti strategi hybrid yang menggabungkan keunggulan berbagai model untuk meningkatkan kinerja deteksi. Contohnya mencakup integrasi CNN dengan Random Forest untuk deteksi anomali jaringan cloud [19], kombinasi Transformer Autoencoder dengan Isolation Forest dan XGBoost untuk deteksi intrusi pada jaringan sensor nirkabel [20], serta *hybrid framework* dalam sistem otonom [21]. Strategi Weighted Fusion yang menggabungkan skor anomali dari

beberapa model dengan memberikan bobot pada masing-masing skor. Bobot menentukan seberapa besar pengaruh tiap model. Skor berbobot tersebut dijumlahkan menjadi satu nilai akhir yang digunakan untuk menentukan apakah suatu data termasuk anomali. Cara ini berpotensi meningkatkan akurasi, menurunkan false alarm, dan memperkuat generalisasi pada data yang tidak seimbang [11], [22], [23].

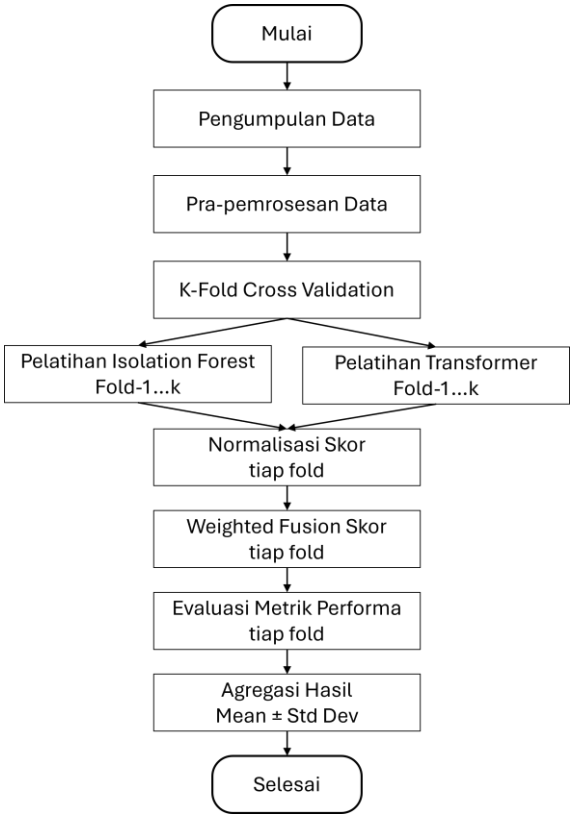
Dengan menggabungkan kemampuan Isolation Forest untuk mendeteksi *point anomaly* dan Transformer untuk menangkap *contextual anomaly*, pendekatan hybrid berpotensi meningkatkan sensitivitas dan stabilitas deteksi. Namun, penerapannya pada pemantauan webserver real-time masih jarang dikaji, sehingga menyisakan *research gap* dalam pengembangan metode hybrid yang efektif untuk data berbasis nilai maupun pola sekuensial.

Khususnya di Indonesia, belum banyak penelitian yang secara eksplisit menerapkan kombinasi Isolation Forest–Transformer dengan *weighted fusion* untuk meningkatkan stabilitas skor deteksi pada domain webserver, termasuk kasus web defacement dan penyalahgunaan situs menjadi halaman judi online. Oleh karena itu, penelitian ini menawarkan kontribusi orisinal berupa perancangan dan evaluasi framework hybrid yang lebih adaptif pada kondisi data tidak seimbang. Berdasarkan hal tersebut, tujuan penelitian ini adalah:

- Mengembangkan sistem deteksi anomali berbasis hybrid Isolation Forest dan Transformer,
- Menerapkan mekanisme *weighted fusion* untuk penggabungan skor anomali, dan
- Mengevaluasi performanya menggunakan F1-score, Average Precision (AP), dan PR-AUC (Precision–Recall Curve).

2. METODE PENELITIAN

Penelitian ini bertujuan mengembangkan sistem deteksi anomali pada webserver berbasis kombinasi model Isolation Forest dan Transformer. Gambaran umum alur penelitian dapat dilihat pada Gambar 1. Sistem ini dirancang untuk mengenali dua jenis anomali utama pada lingkungan produksi, yaitu *point anomaly* (anomali berbasis nilai tunggal) dan *contextual anomaly* (anomali berbasis pola sekuensial).



Gambar 1. Gambaran umum alur penelitian

Data diambil dari server produksi yang menjalankan layanan berbasis Nginx. Metrik sistem dikumpulkan secara periodik, kemudian diproses melalui tahapan normalisasi dan pembentukan *sequence* untuk memenuhi kebutuhan model sekuensial.

Pada tahap pelatihan yang menggunakan k-fold cross validation, Isolation Forest dioptimalkan untuk menemukan titik data yang memiliki perilaku menyimpang dari distribusi umum, sedangkan Transformer dilatih untuk memprediksi nilai metrik berikutnya dan kemudian menghitung error prediksi sebagai dasar deteksi anomali temporal.

Skor anomali dari kedua model kemudian digabungkan menggunakan pendekatan Weighted Fusion dengan pencarian bobot dan threshold terbaik secara otomatis menggunakan grid search, sehingga sistem memperoleh keputusan berbasis gabungan skor yang lebih stabil dan akurat. Penyetelan bobot dan threshold dilakukan menggunakan *grid search* pada setiap fold, berdasarkan F1-Score pada *validation set internal* pada setiap fold mengingat dataset tidak seimbang.

Sistem dievaluasi menggunakan metrik akurasi, precision, recall, F1-score, AP, dan PR-AUC (Precision–Recall Curve). Hasil menunjukkan bahwa pendekatan hybrid memberikan kinerja deteksi yang lebih baik dibandingkan penggunaan model tunggal, dengan peningkatan ketepatan dalam mengidentifikasi anomali dan penurunan tingkat false-positive.

a. Akuisisi dan pra-pemrosesan data

Data dikumpulkan dari sebuah *virtual machine* yang menjalankan layanan web Nginx. Metrik yang diekstraksi meliputi:

- CPU usage (%),
- Lima proses dengan penggunaan CPU tertinggi,
- Memory usage (MB),
- Lima proses dengan penggunaan memori tertinggi,
- Metrik trafik web (nginx requests per second, active connections), dsb.

Data direkam secara periodik dan menghasilkan total 223.878 entri selama pemantauan 40 hari. Log aktivitas dan rekaman beban sistem dianalisis untuk proses *labeling manual* dengan membandingkan dengan log sistem dan juga perbedaan penggunaan CPU maupun memory yang jauh diatas rata-rata sistem ketika berjalan, menghasilkan 13.724 entri anomali, yang kemudian digunakan untuk keperluan evaluasi model. Selanjutnya dilakukan normalisasi menggunakan StandardScaler, agar seluruh fitur memiliki skala setara (mean = 0, std = 1) sehingga mendukung stabilitas pelatihan model berbasis statistika dan neural network. Data kemudian diubah menjadi bentuk sequence agar dapat diproses oleh Transformer. Lalu untuk pelatihan, dataset dibagi menggunakan 5-fold cross-validation, untuk menjaga distribusi kelas dan untuk menguji reliabilitas metode.

b. Metode Deteksi Anomali

Penelitian ini mengusulkan pendekatan yang digunakan bersifat hybrid, menggabungkan dua komponen utama yaitu Isolation Forest kuat dalam deteksi anomali berbasis nilai titik (point anomaly), dan Transformer yang kuat dalam deteksi anomali berbasis ketidakwajaran urutan waktu (contextual/sequence anomaly). Kedua model dieksekusi secara paralel, lalu skor anomalinya digabungkan melalui *weighted fusion*.

c. Algoritma Isolation Forest

Isolation Forest adalah metode *unsupervised learning* yang bekerja berdasarkan prinsip isolasi data. Algoritma ini membangun sejumlah pohon keputusan (*isolation trees*), di mana data dibagi secara acak. Data yang merupakan anomali cenderung lebih cepat terisolasi karena berada jauh dari pusat distribusi umum.

Skor anomali $s(x, n)$ dari sebuah titik x dihitung menggunakan Persamaan (1)

$$s(x,n) = 2 - \frac{E(h(x))}{C(n)} \quad (1)$$

$E(h(x))$ = rata-rata kedalaman dari pohon tempat titik x ditemukan,

$c(n)$ = faktor normalisasi yang dihitung dengan Persamaan (2),

$$(n) = 2H(n-1) - \frac{2n-1}{n} \quad (2)$$

$H(n)$ = bilangan harmonik aproksimasi ln (n) +0.577. Semakin kecil nilai $E(h(x))$ semakin besar kemungkinan titik tersebut adalah anomali. Model ini tidak membutuhkan data berlabel dan sangat efisien pada dataset berdimensi tinggi

Model Isolation Forest memberikan skor anomali per titik data tanpa memerlukan label pada fase pelatihan, label hanya digunakan untuk evaluasi saja. Model Isolation Forest pada penelitian ini dilatih dengan parameter sesuai dengan Tabel **1**.

Tabel 1. Parameter Isolation Forest

Parameter	Nilai	Deskripsi
n_estimators	100	Jumlah pohon isolasi yang dibangun
contamination	0.05	Perkiraan rasio anomali dalam data
random_state	42	Reproducibility

Skor keluaran dari Isolation Forest dihitung dengan decision function, yang menghasilkan nilai positif untuk data normal dan negatif untuk data anomali. Threshold default ditetapkan pada: $threshold = 0.0$, sehingga klasifikasi anomali diberikan sesuai dengan Persamaan (3). Hasil skor anomali kemudian dinormalisasi sebelum digabungkan dengan skor Transformer dalam proses Weighted Fusion

$$\text{label}(x) = \begin{cases} 1 & \text{jika } score(x) < 0 \\ 0 & \text{jika } score(x) \geq 0 \end{cases} \quad (3)$$

d. Arsitektur Transformer

Transformer merupakan model deep learning berbasis *self-attention* yang dirancang untuk mengenali pola dalam urutan data secara paralel dan efisien. Tidak seperti arsitektur lain yang bergantung pada pemrosesan berurutan, Transformer memproses seluruh sequence secara simultan, memungkinkan pemahaman konteks yang lebih luas dalam satu tahap pemrosesan. Model Transformer dirancang untuk melakukan klasifikasi anomali berbasis urutan (sequence-based anomaly classification). Model menerima masukan berupa sequence berdimensi (seq_len, $n\_features$), kemudian mempelajari representasi temporal serta spasial melalui blok *multi-head self-attention* dan *feed-forward network*.

Pada arsitektur Transformer, beberapa parameter utama berpengaruh signifikan terhadap kinerja model dalam mempelajari pola temporal pada data *time-series*. Pertama, dimensi embedding (d_model) menentukan kapasitas representasi fitur; dimensi yang lebih besar memungkinkan penangkapan pola kompleks secara lebih baik, tetapi berisiko menambah beban komputasi dan *overfitting* jika tidak diimbangi dengan jumlah data yang memadai. Kedua, jumlah multi-head attention (n_heads) memengaruhi kemampuan model dalam menangkap hubungan antar-waktu pada berbagai perspektif; semakin banyak *head*, semakin kaya konteks yang dapat dipelajari, meskipun dengan biaya komputasi lebih tinggi. Parameter lain seperti ukuran *feed-forward network*, serta mekanisme regularisasi (misalnya *dropout*) juga berperan dalam mengontrol kapasitas model agar tetap seimbang antara akurasi dan generalisasi. Dengan demikian, pemilihan konfigurasi parameter yang tepat

menjadi kunci keberhasilan pelatihan Transformer dalam mendeteksi anomali berbasis pola sekuensial.

Rincian Arsitektur Transformer yang digunakan dalam penelitian ini di tampilan seperti berikut ini:

1. Input Layer
   Menerima sequence berdimensi:(seq_len × $n\_features$)
2. Dense Projection (Embedding)
   Setiap vektor fitur diproyeksikan ke dimensi $d\_model = 64$.
3. Positional Encoding via Embedding
   Penambahan representasi posisi $(0–seq_len - 1)$untuk mempertahankan informasi urutan.
4. Multi-Head Attention
   - Jumlah head: 2
   - Key dimension: 64
   - Mampu menangkap hubungan antar-waktu (*temporal dependency*).
5. Residual + Layer Normalization
   Output attention digabungkan dengan input yang dinormalisasi.
6. Feed-Forward Network
   - Dense(64) + ReLU
   - Dropout(0.1)
   - Residual + LayerNorm
7. GlobalAveragePooling1D
   Menyatukan informasi temporal menjadi satu vektor.
8. Dense Classifier
   - Dense(16, ReLU)
   - Dense(1, sigmoid) dengan probabilitas anomali
9. Loss Function

Model dioptimasi menggunakan Binary Cross-Entropy dengan Persamaan (4)

$$L = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)] \quad (4)$$

dengan:

y_i = label aktual

$\hat{y}_i$ = probabilitas prediksi

Selama proses pelatihan model, Model Transformer ini menggunakan Adam Optimizer dengan metrik performa yang digunakan untuk stopping criteria adalah Akurasi (stop maksimal 20 epoch apabila akurasi tidak mengalami kenaikan) dengan lama pelatihan 100 epoch dan batch size 32. Model Transformer dalam penelitian ini menghasilkan probabilitas anomali $\hat{y} \in [0,1]$. Nilai ini kemudian dinormalisasi dan dikombinasikan dalam tahap Weighted Fusion bersama skor Isolation Forest.

e. Model Hybrid menggunakan Weighted Fusion

Kedua model Isolation Forest dan Transformer dijalankan secara paralel dan hasil deteksinya digabungkan melalui pendekatan Weighted Fusion. Setiap model menghasilkan skor anomali yang kemudian dinormalisasi terlebih dahulu agar berada dalam rentang [0,1].

Keputusan akhir dalam proses *deteksi anomali* dilakukan dengan menggabungkan hasil dari kedua model, menggunakan pendekatan Weighted Fusion yang secara umum dihitung dengan Persamaan (5).

$$S(x) = \sum_{m=1}^{M} w_m \tilde{s}_m(x),\ \sum_{m=1}^{M} w_m = 1,\ w_m \geq 0 \quad (5)$$

$S(x)$ = skor anomali gabungan(*hybrid*)

w_m = bobot dari detektor ke-m

$\tilde{s}_m(x)$ = skor ternormalisasi detektor ke-m

M = jumlah detektor

Untuk penelitian ini untuk 2 model (Isolation Forest + Transformer) menggunakan Persamaan (6)

$$S(x) = w_{IF}\,\tilde{s}_{IF}(x) + w_{TR}\,\tilde{s}_{TR}(x),\ w_{IF} + w_{TR} = 1 \quad (6)$$

$\tilde{s}_{IF}(x)$ = skor Isolation Forest ternormalisasi

$\tilde{s}_{TR}(x)$ = skor Transformer ternormalisasi

w_{IF}, w_{TR} = bobot masing-masing model

Setelah mendapatkan skor gabungan $S(x)$, ditetapkan ambang batas dengan Persamaan (7).

$$label(x) = \begin{cases} anomali, & S(x) \geq \tau \\ normal, & S(x) < \tau \end{cases} \quad \tau = \text{threshold} \quad (7)$$

Proses pencarian nilai optimal untuk w_{IF}, w_{TR}, τ dilakukan menggunakan grid search pada *validation set internal* pada setiap fold dengan kriteria maksimalisasi F1-score, karena dataset tidak seimbang. Grid search ini menggunakan kombinasi dari tiap bobot masing-masing w_{IF}, w_{TR} berkisar antara [0.25, 0.5, 0.75] serta τ berkisar antara [0.3, 0.5, 0.7].

Evaluasi sistem dilakukan dengan menghitung beberapa metrik performa utama, yaitu:

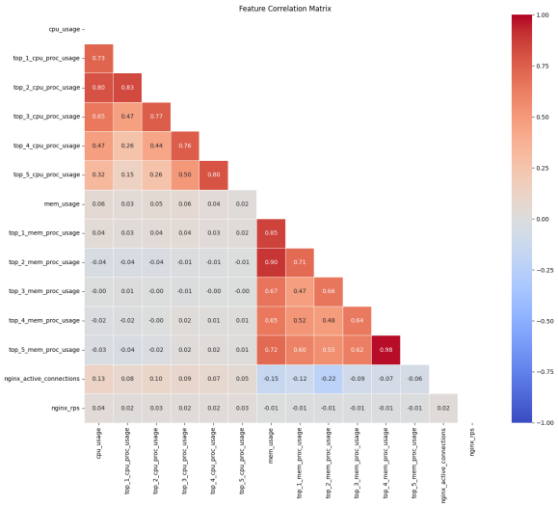
- Accuracy = (TP + TN) / (TP + TN + FP + FN)
- Precision = TP / (TP + FP)
- Recall = TP / (TP + FN)
- F1-score = 2 × (Precision × Recall) / (Precision + Recall)
- Average Precision (AP) dan Precision–Recall Curve untuk mengukur kestabilan klasifikasi.

3. HASIL DAN PEMBAHASAN

Bagian ini menyajikan hasil dari penerapan metode kombinasi *Isolation Forest* dan *Transformer* dalam mendeteksi anomali pada sistem IT berbasis metrik performa. Penelitian ini menggunakan data *time-series* berupa penggunaan *CPU*, memori, serta metrik aktivitas layanan seperti *nginx_rps* dan *nginx_active_connections*. Setiap tahapan eksperimen ditampilkan secara sistematis mulai dari eksplorasi data, pelatihan model, evaluasi hasil, hingga analisis performa sistem *deteksi anomali*. Visualisasi hasil serta analisis kuantitatif digunakan untuk memperkuat temuan yang diperoleh selama proses pengujian.

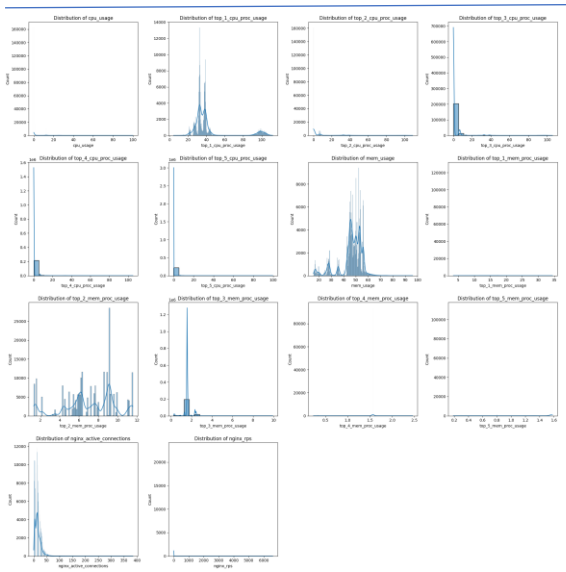
a. Eksplorasi data

Bagian awal dari analisis ini dimulai dengan eksplorasi data, yang bertujuan untuk memahami hubungan antar fitur serta distribusi masing-masing metrik sistem. Langkah ini penting untuk mengidentifikasi potensi keterkaitan antar variabel dan mendeteksi adanya ketidakseimbangan data (*skewness*) yang dapat memengaruhi performa model *deteksi anomali*. Salah satu pendekatan yang digunakan adalah dengan menganalisis korelasi antar metrik performa sistem. Korelasi ini dihitung menggunakan metode *Pearson* dan divisualisasikan dalam bentuk matriks panas (*heatmap*). Hasilnya ditunjukkan pada Gambar 2.



Gambar 2. Korelasi antar Metrik Performa Sistem (Heatmap)

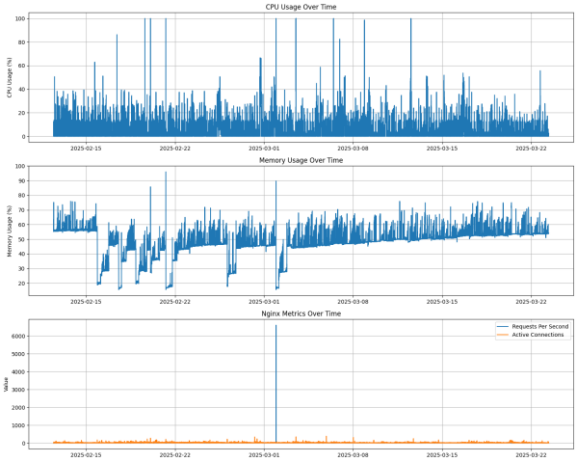
Selain hubungan antar fitur, dilakukan juga analisis distribusi masing-masing fitur secara individual untuk melihat sebaran nilai yang dimiliki oleh data. Hal ini bertujuan untuk mengidentifikasi apakah data memiliki distribusi normal, simetris, atau justru sangat condong (*skewed*), yang bisa memengaruhi deteksi anomali. Visualisasi distribusi ini ditampilkan pada Gambar 3.



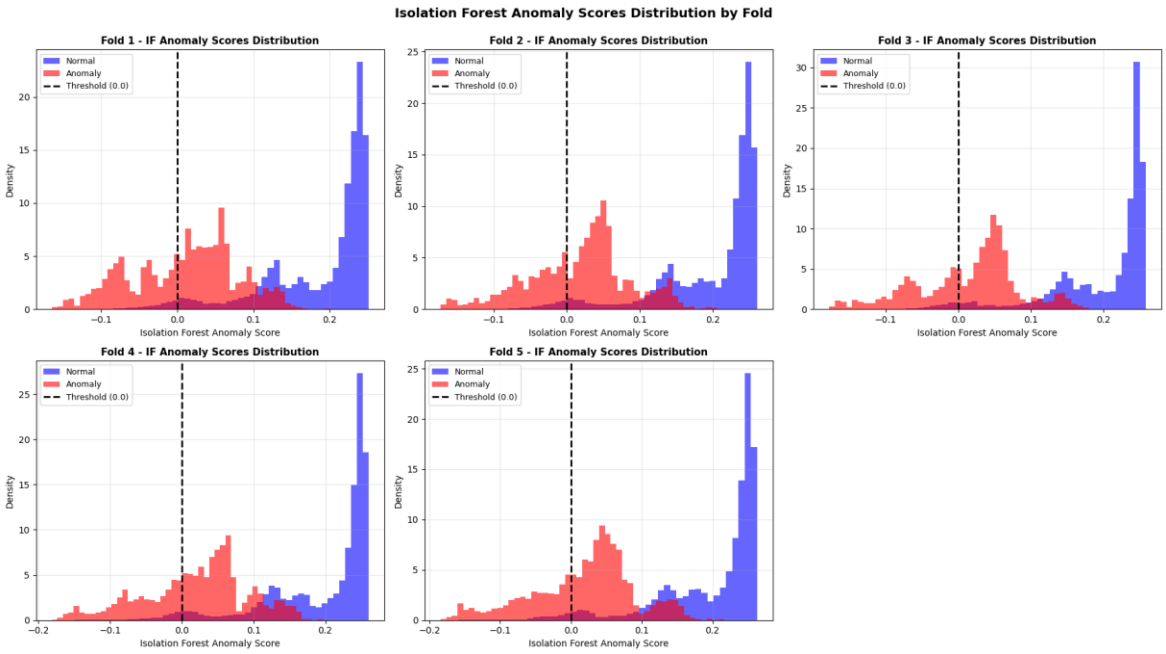
Gambar 3. Distribusi Histogram Tiap Fitur Sistem

b. Tren Time-Series Metrik Sistem

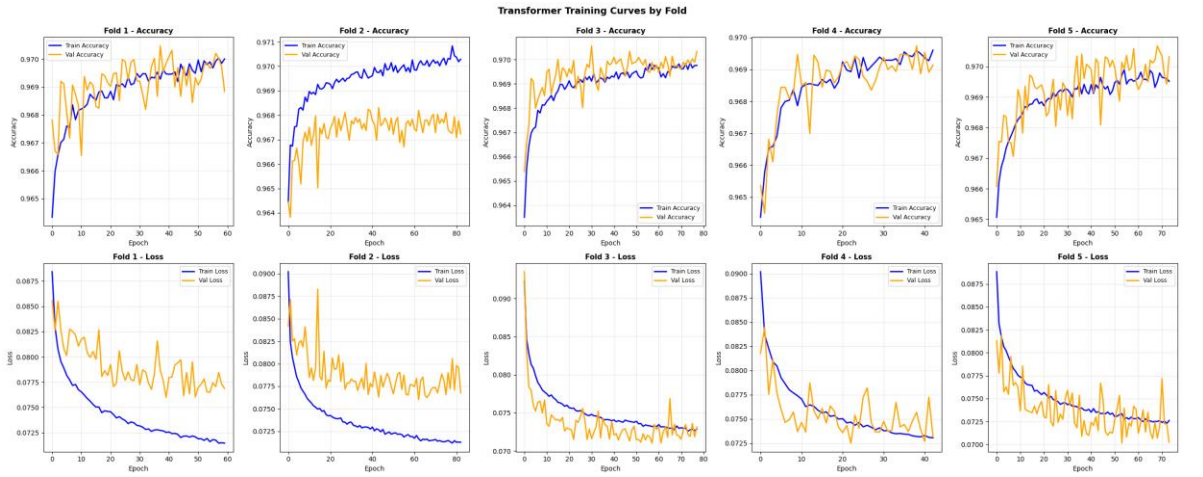
Analisis terhadap pola waktu dilakukan untuk mendeteksi fluktuasi beban sistem secara temporal. Gambar 4 menampilkan visualisasi *time-series* dari tiga metrik utama, yaitu *CPU usage*, *memory usage*, dan metrik dari layanan *Nginx*.



Gambar 4. Tren Time-Series Penggunaan CPU, Memori, dan Trafik Nginx



Gambar 5. Distribusi Skor Anomali dari Model Isolation Forest pada tiap fold



Gambar 6. Grafik Akurasi dan Loss Model Transformer Selama Pelatihan tiap fold

Dari visualisasi tersebut, diperoleh beberapa temuan penting. *CPU usage* mengalami lonjakan-lonjakan tajam yang mencapai hingga 100%, yang dapat dikategorikan sebagai anomali karena menyimpang dari pola normal sistem. *Memory usage* menunjukkan pola kenaikan bertahap (*increasing trend*) dengan beberapa penurunan drastis secara tiba-tiba yang berpotensi sebagai *outlier*. Metrik *Nginx*, seperti jumlah permintaan per detik (*requests per second*), menunjukkan *spike* ekstrem terutama pada awal Maret, yang dapat mengindikasikan lonjakan trafik atau potensi serangan, seperti *DDoS*. Pola-pola ini menyediakan konteks temporal yang krusial bagi model *Transformer* untuk mengidentifikasi ketidakwajaran dalam urutan waktu.

Dengan melihat data serta log sistem yang menunjukkan serangan, dataset ini lalu dilakukan pelabelan manual dengan total 13.724 entri data anomali dari 223.878 total data. Data *time-series* metrik sistem (CPU, memori, metrik Nginx) diproses dengan StandardScaler yang dilatih pada data train pada setiap *fold* dan dibentuk menjadi sequence berukuran seq_len untuk kebutuhan model sekuensial. Walaupun pembagian data training menggunakan skema 5-*fold cross-validation*, data tetap dijaga urutan waktu (tanpa *shuffling*) untuk mencegah data leakage pada *time-series*.

c. Evaluasi Skor Anomali dari Isolation Forest

Isolation Forest bekerja dengan mengukur seberapa cepat suatu titik data dapat diisolasi dalam pohon keputusan acak. Gambar 5 memperlihatkan distribusi skor anomali yang dihasilkan oleh model pada tiap fold.

Beberapa hal yang dapat diamati dari hasil distribusi skor anomali adalah bahwa mayoritas skor normal berada pada rentang positif (normal) antara 0.10 hingga 0.23. Titik-titik dengan skor di bawah nol (skor < 0 atau negatif), yang berada di sebelah kiri garis *threshold*, diidentifikasi sebagai *outlier* atau anomali. Namun pada label, terlihat di rentang 0 hingga 0.10 terdapat anomali juga. *Threshold* ditetapkan pada 0.0, karena pada rentang 0 sampai 0.10 masih ada data normal untuk menghindari false positif dan juga sesuai dengan prinsip bahwa semakin cepat suatu data dapat diisolasi, maka semakin besar kemungkinan data tersebut merupakan anomali.

d. Evaluasi Pelatihan Model Transformer

Model *Transformer* digunakan untuk memprediksi nilai metrik performa sistem selanjutnya berdasarkan data historis. Akurasi dan *loss* selama proses pelatihan pada setiap fold ditampilkan dalam Gambar 6. Selama pelatihan, metrik utama yang dicatat adalah accuracy dan val_loss untuk *early stopping*, namun untuk penilaian akhir pada data uji digunakan F1-score dan Average Precision (PR-AUC) mengingat ketidakseimbangan data. Hasil pelatihan tiap fold menunjukkan kurva validasi yang stabil (tanpa overfitting berarti) dan performa prediksi yang konsisten pada *hold-out*. Temuan dari grafik pelatihan menunjukkan bahwa akurasi pada data pelatihan dan validasi relatif stabil, berada di atas 96.4% pada setiap fold, yang mencerminkan kemampuan generalisasi model yang baik. *Loss* validasi juga tampak rendah dan konsisten sepanjang *epoch*, mengindikasikan bahwa model tidak mengalami *overfitting* yang signifikan. Meskipun terdapat beberapa fluktuasi kecil pada *epoch* tertentu, hal ini kemungkinan besar disebabkan oleh *noise* dalam data atau *shifting pattern* pada sistem produksi. Secara keseluruhan, model ini efektif digunakan untuk mendeteksi anomali berbasis *error* prediksi, di mana *error* yang melebihi *threshold* akan diklasifikasikan sebagai anomali.

e. Evaluasi pencarian parameter Hybrid pada Weighted Fusion

Skor anomali dari IF dan Transformer terlebih dahulu dinormalisasi ke [0,1]. Dengan menggunakan grid search parameter bobot dan threshold pada Weighted Fusion, dalam penelitian ini dicari performa terbaik berdasarkan F1-score. Karena ketimpangan data yang sangat jauh, pengukuran akurasi jadi tidak efektif. Hasil grid search ini ditampilkan pada Tabel 2 Hasil grid search parameter Weighted Fusion tiap fold.

Tabel 2. Hasil grid search parameter Weighted Fusion tiap fold

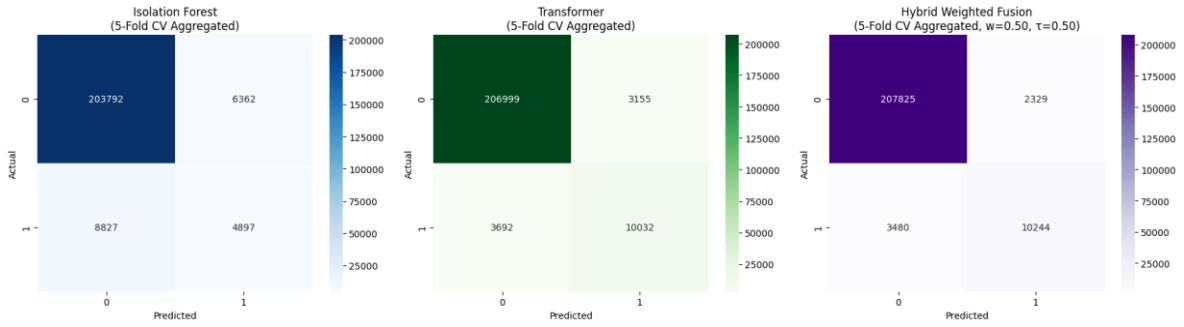
w_{IF}	w_{TR}	τ	F1-fold1 (%)	F1-fold2 (%)	F1-fold3 (%)	F1-fold4 (%)	F1-fold5 (%)
0.25	0.75	0.30	75.94	76.14	76.49	75.85	77.06
0.25	0.75	0.50	76.29	76.18	76.70	75.73	76.96
0.25	0.75	0.70	64.45	60.52	63.92	60.13	61.08
0.50	0.50	0.30	66.86	67.10	67.39	67.10	68.73
0.50	**0.50**	**0.50**	**77.31**	**77.41**	**78.61**	**77.69**	**78.57**
0.50	0.50	0.70	61.54	58.48	63.85	53.99	57.35
0.75	0.25	0.30	53.47	55.99	55.73	54.44	55.97
0.75	0.25	0.50	76.15	74.04	74.35	74.42	73.96
0.75	0.25	0.70	43.94	37.89	38.02	31.90	28.39

F1-score tertinggi didapatkan ketika kombinasi bobot w_{IF}=0.5 w_{TR}=0.5 dan threshold τ=0.5. Kombinasi ini konsisten tertinggi pada semua fold. Dipilih F1-score tertinggi karena untuk menyesuaikan dataset yang tidak seimbang antara kelas normal dan anomali dan selaras dengan tujuan mengoptimalkan keseimbangan precision-recall.

f. Evaluasi Sistem Deteksi Anomali

Setelah kedua model dijalankan secara paralel, lalu di evaluasi secara masing-masing, penelitian ini menggabungkan hasil deteksi menggunakan Weighted Fusion terbaik dengan parameter w_{IF}=0.5 w_{TR}=0.5 τ=0.5. Gambar 7. menampilkan *confusion matrix* dari hasil pengujian terhadap data validasi aggregated atau gabungan dari setiap fold terhadap model Isolation Forest, Transformer dan Hybrid antara Isolation Forest dan Transformer menggunakan Weighted Fusion. Perbandingan kinerja rata-rata Isolation Forest, Transformer, dan Hybrid Weighted Fusion pada setiap fold ditunjukkan pada Tabel 3 dengan indikator Accuracy, Recall, Precision, F1-score, AP. Serta selisih matrix performa model hybrid dan model tunggal ditampilkan pada Tabel 4. Sesuai tabel, Hybrid Weighted Fusion menunjukkan peningkatan menyeluruh (seperti F1-score naik menjadi 77.92% ± 0.63%, Average Precision 84.77% ± 0.69%), sementara accuracy tidak dijadikan metrik utama walaupun angkanya sangat bagus karena bias di data yang tidak seimbang.

Model Isolation Forest cenderung under-detect anomali kontekstual (recall rendah) karena tidak memodelkan urutan. Transformer lebih sensitif terhadap perubahan pola, namun berisiko menaikkan false positives. Model Fusion menurunkan false positives dan menjaga recall melalui kombinasi skor yang saling melengkapi, sehingga lebih stabil untuk monitoring real-time. Model mampu mendeteksi sebagian besar anomali dengan tingkat false alarm yang masih tergolong rendah, sehingga cukup andal untuk digunakan dalam sistem webserver.



Gambar 7. Confusion matrix Aggregated semua fold dari setiap model

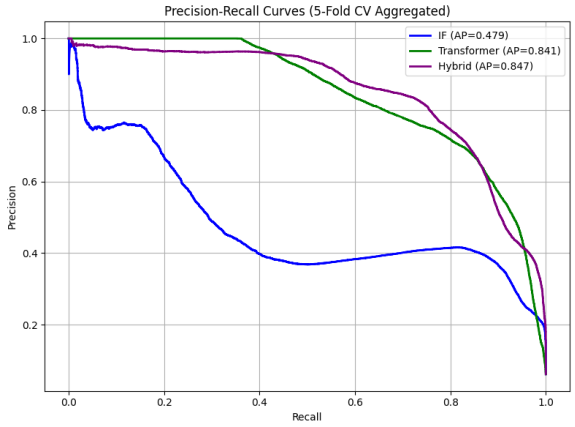
Tabel 3. Hasil metrik performa rata-rata tiap model

Model	Akurasi (%) (mean ± std)	Recall (%) (mean ± std)	Precision (%) (mean ± std)	F1-score (%) (mean ± std)	AP (%) (mean ± std)
Isolation Forest	93.22 ± 0.24	35.67 ± 2.49	43.48 ± 2.75	39.18 ± 2.55	48.08 ± 2.81
Transformer	96.94 ± 0.08	73.09 ± 1.40	76.10 ± 1.09	74.55 ± 0.48	84.18 ± 0.32
Hybrid Weighted Fusion	**97.41 ± 0.12**	**74.65 ± 0.51**	**81.49 ± 0.84**	**77.92 ± 0.63**	**84.77 ± 0.69**

Tabel 4. Selisih metrix performa hybrid vs model tunggal

Perbandingan	Δ Akurasi (%)	Δ Recall (%)	Δ Precision (%)	Δ F1-score (%)	Δ AP (%)
Hybrid – Isolation Forest	+4.19	+38.98	+38.01	+38.74	+36.69
Hybrid – Transformer	+0.47	+1.56	+5.39	+3.37	+0.59

Gambar 8 menampilkan kurva Precision–Recall (PR-curve) untuk tiga pendekatan: Isolation Forest, Transformer, dan metode gabungan (Weighted Fusion) gabungan dari semua fold. Kurva ini menunjukkan hubungan antara tingkat Precision dan Recall pada berbagai nilai ambang (threshold) yang digunakan dalam klasifikasi anomali.



Gambar 8. Kurva Precision–Recall (PR-curve)

Dari visualisasi terlihat bahwa:

1. Isolation Forest menghasilkan kurva PR yang relatif rendah dan sempit, menandakan bahwa model kesulitan menjaga keseimbangan antara Precision dan Recall. Hal ini sejalan dengan karakter IF yang hanya mendeteksi anomali berbasis distribusi titik (*point anomaly*), sehingga kurang sensitif terhadap pola temporal yang lebih kompleks. Akibatnya, recall rendah dan Precision berfluktuasi pada sebagian besar threshold.
2. Transformer menunjukkan peningkatan performa yang signifikan dibanding Isolation Forest. Kurva yang lebih tinggi dan luas mengindikasikan bahwa Transformer mampu menangkap konteks data sekuensial sehingga dapat mengenali anomali yang berkaitan dengan pola waktu (*contextual anomaly*). Namun, pada beberapa bagian kurva terlihat penurunan Precision ketika Recall meningkat, mencerminkan peningkatan *false positives*.
3. Hybrid Weighted Fusion memperlihatkan kurva PR terbaik di antara ketiga pendekatan. Kurvanya dominan berada di atas dua model lainnya dan memberikan nilai rata-rata Precision yang lebih tinggi pada berbagai nilai Recall. Hal ini menegaskan bahwa kombinasi skor Transformer dan Isolation Forest memberikan deteksi anomali yang lebih stabil serta mampu mengurangi *false positives* tanpa mengorbankan kemampuan mendeteksi anomali.

Nilai Average Precision (AP) untuk pendekatan Hybrid Weighted Fusion berada di kisaran 84.77% ± 0.69%, lebih

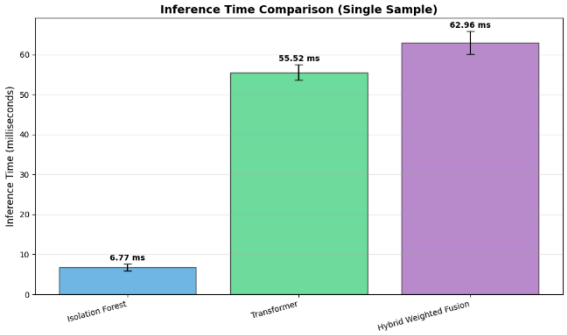
tinggi daripada Transformer maupun Isolation Forest, yang menegaskan keunggulannya dalam menangani data tidak seimbang (*imbalanced dataset*). Dengan demikian, Hybrid Weighted Fusion menjadi pendekatan yang paling efektif untuk skenario monitoring sistem karena menawarkan kompromi optimal antara sensitivitas dan ketepatan.

Penelitian oleh [20] menunjukkan pendekatan serupa, yaitu menggabungkan *Autoencoder* berbasis *Transformer*, *Isolation Forest*, dan *XGBoost* untuk mendeteksi anomali dalam jaringan sensor nirkabel (*WSN*), dan berhasil mencapai akurasi sebesar 95%. Penelitian ini semakin menguatkan bahwa integrasi teknik *deteksi anomali* yang berbeda dapat memberikan hasil yang lebih optimal.

Hasil pengujian waktu *inference* per sampel pada Tabel 5 dan Gambar 9 menunjukkan bahwa Isolation Forest memiliki kinerja paling cepat dengan waktu rata-rata sekitar 6.77 ms, disertai variasi yang relatif rendah (0.86 ms) dan rentang waktu antara 5.31–8.23 ms. Model Transformer membutuhkan waktu komputasi lebih besar, yaitu rata-rata 55.52 ms dengan standar deviasi 1.95 ms, serta rentang waktu 52.19–63.02 ms yang mencerminkan kompleksitas arsitektur berbasis *self-attention*. Sementara itu, metode Hybrid Weighted Fusion menghasilkan waktu komputasi rata-rata sekitar 62.96 ms, sedikit lebih tinggi dibandingkan Transformer karena memproses keluaran dari dua model, namun tetap berada dalam rentang yang relatif stabil (58.78–72.99 ms) dengan variasi rendah.

Tabel 5. Perbandingan waktu inferensi

Model	Rata-rata (ms)	Std Dev (ms)	Min (ms)	Max (ms)
Isolation Forest	6.767	0.862	5.307	8.234
Transformer	55.522	1.949	52.188	63.016
Hybrid Weighted Fusion	62.961	2.831	58.784	72.992



Gambar 9. Perbandingan waktu inferensi

Secara keseluruhan, meskipun pendekatan hybrid memberikan biaya komputasi lebih besar dibandingkan Isolation Forest saja, waktu eksekusi pada kisaran puluhan milidetik tersebut masih tergolong cepat dan layak digunakan untuk skenario deteksi anomali pada web secara *near real-time*. Hal ini membuka peluang untuk integrasi dalam sistem keamanan webserver nyata, misalnya sebagai komponen pendeteksi awal pada *Intrusion Detection System* (IDS) atau sebagai modul analitik dalam platform *Security Information and Event Management* (SIEM), sehingga anomali dapat direspons lebih cepat dan akurat. Dari sudut pandang *trade-off*, peningkatan kinerja deteksi melalui kombinasi dua model (Hybrid) datang dengan konsekuensi peningkatan kompleksitas komputasi dibandingkan penggunaan Isolation Forest tunggal. Namun, biaya tambahan tersebut relatif kecil dibandingkan peningkatan stabilitas dan akurasi deteksi yang diperoleh, sehingga strategi hybrid tetap menjadi opsi

yang menarik untuk diterapkan dalam sistem webserver.

Meskipun hasil evaluasi menunjukkan peningkatan kinerja melalui pendekatan hybrid, model yang diusulkan masih memiliki beberapa keterbatasan. Pertama, sistem masih bergantung pada *offline learning* dan data historis, sehingga kurang adaptif terhadap perubahan pola trafik yang terjadi secara real-time. Kedua, nilai ambang (*threshold*) pada Isolation Forest belum dioptimasi secara dinamis, sehingga perlu dilakukan penyesuaian manual maupun otomatis menggunakan parameter tuning untuk menjaga sensitivitas deteksi terhadap konteks operasional yang berbeda. Selain itu, model Transformer perlu dilatih ulang ketika terdapat data baru atau perubahan distribusi pola waktu, yang berpotensi menambah biaya komputasi serta waktu pemrosesan. Oleh karena itu, diperlukan pengembangan lebih lanjut untuk mengadopsi mekanisme *online learning* atau pembaruan model secara inkremental agar sistem dapat beradaptasi dengan lebih cepat terhadap dinamika lingkungan webserver.

4. KESIMPULAN

Penelitian ini mengusulkan sistem deteksi anomali pada webserver dengan mengombinasikan metode Isolation Forest dan Transformer melalui pendekatan Weighted Fusion. Hasil evaluasi menunjukkan bahwa masing-masing model memiliki karakteristik yang saling melengkapi. Isolation Forest efektif untuk mendeteksi *point anomaly* dengan efisiensi komputasi tinggi, namun cenderung memiliki *recall* rendah. Sementara itu, Transformer mampu menangkap pola temporal sehingga memberikan *recall* lebih baik, tetapi berpotensi menghasilkan *false positive* lebih tinggi.

Integrasi keduanya melalui Weighted Fusion dengan konfigurasi terbaik bobot w_{IF}=0.5 w_{TR}=0.5 dan threshold τ=0.5 menghasilkan peningkatan performa dengan nilai rata-rata F1-score sebesar 77.92% ± 0.63% dan Average Precision (AP) sebesar 84.77% ± 0.69%, lebih tinggi dibandingkan penggunaan model tunggal. Kurva Precision–Recall memperlihatkan bahwa pendekatan hybrid mampu mempertahankan stabilitas deteksi pada berbagai nilai ambang, sehingga lebih andal pada kondisi data yang tidak seimbang. Waktu inferensi yang berada pada kisaran 63 ms menunjukkan potensi penerapan sistem dalam skenario *near real-time*.

Dalam konteks praktis, pendekatan ini berpotensi diintegrasikan ke sistem keamanan organisasi seperti *Intrusion Detection System* (IDS) atau *Security Information and Event Management* (SIEM) sebagai komponen pendeteksi awal ancaman, termasuk indikasi serangan web defacement maupun penyalahgunaan situs. Penerapannya dapat memperkuat proses monitoring operasional dengan mendeteksi anomali secara lebih akurat dan cepat.

Meski demikian, sistem memiliki beberapa keterbatasan. Di antaranya adalah ketergantungan pada threshold statis terutama pada model Isolation Forest, serta kebutuhan pelatihan ulang saat terjadi perubahan pola trafik terutama pada Transformer, sehingga adaptivitas terhadap kondisi dinamis masih terbatas. Penggunaan skema *offline learning* juga membatasi respons sistem terhadap data baru yang masuk secara kontinu.

Arah penelitian selanjutnya dapat difokuskan pada:

- pengembangan mekanisme *adaptive/online thresholding* untuk menangani perubahan perilaku sistem,
- penerapan *online learning* atau pembaruan model secara inkremental agar lebih adaptif terhadap dinamika trafik web, dan
- pengujian pada lingkungan server berskala besar atau *multi-node* untuk menilai kinerja dan skalabilitas dalam kondisi produksi yang lebih kompleks.

Pendekatan ini diharapkan dapat meningkatkan fleksibilitas dan ketahanan sistem dalam mengidentifikasi anomali pada infrastruktur webserver nyata.

DAFTAR PUSTAKA

[1] Badan Siber dan Sandi Negara (BSSN), "Langkah-Langkah Penanggulangan Insiden Web Defacement." [Online]. Available: https://www.bssn.go.id/langkah-langkah-penanggulangan-insiden-web-defacement-judi-online-2/

[2] CNN Indonesia, "Polisi Sebut Sindikat Judol Kamboja Retas 855 Situs Pemerintah RI." [Online]. Available: https://www.cnnindonesia.com/nasional/20240712201314-12-1120710/polisi-sebut-sindikat-judol-kamboja-retas-855-situs-pemerintah-ri

[3] CSIRT Tangerang Kota, "Analisa Web Defacement: Judi Online." [Online]. Available: https://csirt.tangerangkota.go.id/berita/analisa-web-defacement

[4] Detikcom, "Sindikat Judol Raup Rp 170 M dari Sewakan Situs Pemerintah yang Diretas." [Online]. Available: https://news.detik.com/berita/d-7436294/sindikat-judol-raup-rp-170-m-dari-sewakan-situs-pemerintah-yang-diretasnya

[5] A. Farzad and T. A. Gulliver, "Unsupervised log message anomaly detection," *ICT Express*, vol. 6, no. 3, pp. 229–237, 2020, doi: 10.1016/j.icte.2020.06.003.

[6] M. S. Lakshmi, G. Rajavikram, V. Dattatreya, B. S. Jyothi, S. Patil, and M. Bhavsingh, "Evaluating the Isolation Forest Method for Anomaly Detection in Software-Defined Networking Security," *Journal of Electrical Systems*, vol. 19, no. 4, pp. 279–297, 2023, doi: 10.52783/jes.639.

[7] D. B. Santoso and Y. Wahyuni, "SESTEM LOG WEB SERVER SEBAGAI PENDETEKSI ANOMALI MENGGUNAKAN ISOLATION FOREST — Web Server Log System as an Anomaly Detector Using Isolation Forest," *JUBIKOM: Jurnal Aplikasi Bisnis dan Komputer*, vol. 4, no. 3, pp. 90–96, 2024, [Online]. Available: https://journal.unpak.ac.id/index.php/jubikom/article/view/10941

[8] V. Aschepkov, "The use of the Isolation Forest model for anomaly detection in measurement data," *Innovative Technologies and Scientific Solutions for Industries*, no. 1(27), pp. 236–245, 2024, doi: 10.30837/itssi.2024.27.236.

[9] H. Xiang and others, "OptIForest: Optimal Isolation Forest for Anomaly Detection," in *Proceedings of the IJCAI*, 2023, pp. 2379–2387. doi: 10.24963/ijcai.2023/264.

[10] L. A. Muhammed, "Anomaly Detection in Streaming Data using Isolation Forest Tree," 2024. [Online]. Available: https://www.researchgate.net/publication/383022377

[11] S. Fatemifar, M. Awais, A. Akbari, and J. Kittler, "Developing a Generic Framework for Anomaly Detection," *Pattern Recognit*, vol. 124, p. 108500, 2022, doi: 10.1016/j.patcog.2021.108500.

[12] F. Hang, W. Guo, H. Chen, L. Xie, C. Zhou, and Y. Liu, "Logformer: Cascaded Transformer for System Log Anomaly Detection," *Computer Modeling in Engineering & Sciences*, vol. 136, no. 1, pp. 517–529, 2023, doi: 10.32604/cmes.2023.025774.

[13] H. Kenji, "Real-Time Anomaly Detection Using Transformer-Based Architectures in Cloud Traffic," 2025. [Online]. Available: https://www.researchgate.net/publication/391768629

[14] M. Orabi, K. P. Tran, P. Egger, and S. Thomassey, "Anomaly detection in smart manufacturing: An Adaptive Adversarial Transformer-based model," *J Manuf Syst*, vol. 77, pp. 591–611, 2024, doi: 10.1016/j.jmsy.2024.09.021.

[15] W. Sakong, J. Kwon, K. Min, S. Wang, and W. Kim, "Anomaly Transformer Ensemble Model for Cloud Data Anomaly Detection," *IEEE Transactions on Cloud Computing*, vol. 12, no. 4, pp. 1305–1313, 2024, doi: 10.1109/TCC.2024.3466174.

[16] F. Zeng, M. Chen, C. Qian, Y. Wang, Y. Zhou, and W. Tang, "Multivariate time series anomaly detection with adversarial transformer architecture in the Internet of Things," *Future Generation Computer Systems*, vol. 144, pp. 244–255, 2023, doi: 10.1016/j.future.2023.02.015.

[17] S. Zia, N. Bibi, S. Alhazmi, N. Muhammad, and A. Alhazmi, "Enhanced Anomaly Detection in IoT Through Transformer-Based Adversarial Perturbations Model," *Electronics (Basel)*, vol. 14, no. 6, 2025, doi: 10.3390/electronics14061094.

[18] K. Xu, M. Xia, X. Mu, W. Chen, and B. Ni, "EnsembleLens: Ensemble-based Visual Exploration of Anomaly Detection Algorithms with Multidimensional Data," *IEEE Trans Vis Comput Graph*, vol. 25, no. 1, pp. 109–119, 2019, doi: 10.1109/TVCG.2018.2864886.

[19] A. D. Vibhute and V. Nakum, "Deep learning-based network anomaly detection and classification in an imbalanced cloud environment," in *Procedia Computer Science*, 2024, pp. 1636–1645. doi: 10.1016/j.procs.2024.01.161.

[20] A. Haque and H. Soliman, "A Transformer-Based Autoencoder with Isolation Forest and XGBoost for Malfunction and Intrusion Detection in Wireless Sensor Networks for Forest Fire Prediction," *Future Internet*, vol. 17, no. 4, 2025, doi: 10.3390/fi17040164.

[21] S. Nazat and others, "Ensemble Learning Framework for Anomaly Detection in VANETs," *Sensors*, vol. 25, no. 16, p. 5105, 2025, doi: 10.3390/s25165105.

[22] H. He and others, "Isolation Forest-voting Fusion-multioutput: A Stroke Risk Prediction Model," *Comput Methods Programs Biomed*, 2024, doi: 10.1016/j.cmpb.2024.108500.

[23] S. Wang, R. Jiang, Z. Wang, and Y. Zhou, "Deep Learning-based Anomaly Detection and Log Analysis for Time Series Data," *arXiv preprint*, 2024.