BAB IV HASIL DAN PEMBAHASAN

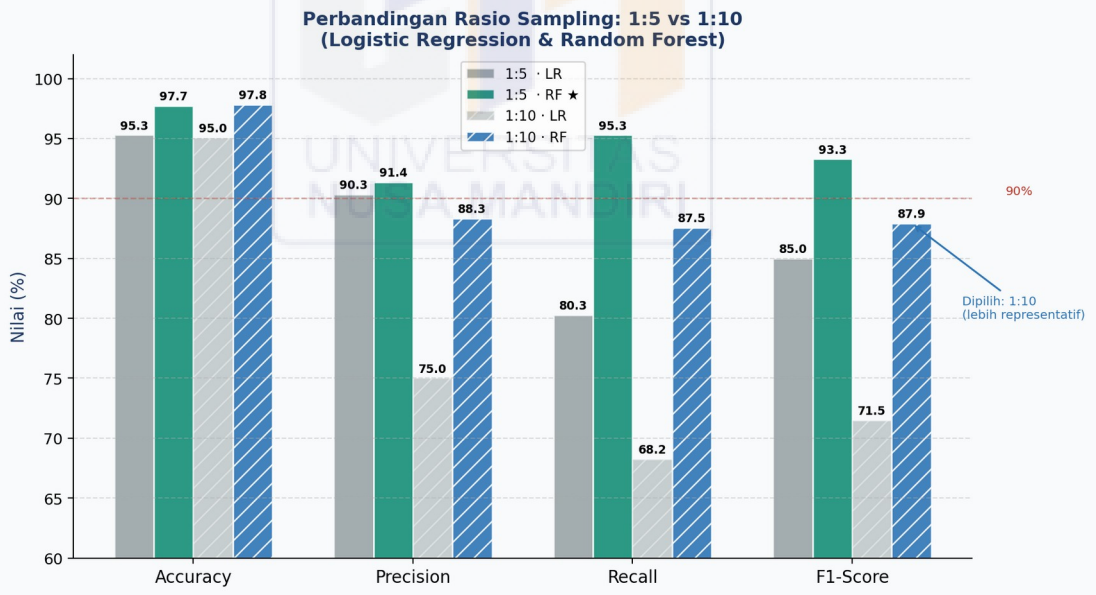
4.1. Eksperimen Sampling Ratio (RQ3)

Sebagai langkah awal, dilakukan eksperimen untuk menentukan rasio sampling yang optimal. Dua konfigurasi diuji: rasio 1:5 (6.990 node total) dan 1:10 (12.815 node total). Hasil perbandingan ditunjukkan pada Tabel IV.1 dan Gambar IV.1.

Tabel IV.1. Perbandingan Sampling Ratio 1:5 vs 1:10

Rasio	Model	Total Node	Accuracy	Precision	Recall	F1-Score
1:5	Logistic Regression	6.990	95,28%	90,34%	80,26%	85,00%
1:5	Random Forest	6.990	97,71%	91,36%	95,28%	93,28%
1:10	Logistic Regression	12.815	95,04%	75,00%	68,24%	71,46%
1:10	Random Forest	12.815	97,82%	88,31%	87,55%	87,93%

Tabel IV. 1 Perbandingan Sampling Ratio 1:5 vs 1:10



Gambar IV. 1 Perbandingan Rasio Sampling 1:5 vs 10

Gambar IV.1. Perbandingan Rasio Sampling 1:5 vs 1:10

Rasio 1:10 dipilih sebagai konfigurasi utama karena: (1) menghasilkan dataset lebih besar, (2) distribusi lebih mendekati kondisi nyata, dan (3) metrik yang lebih konservatif dan jujur secara statistik. Meskipun F1 pada 1:5 lebih tinggi

(93,28% vs 87,93%), hal ini disebabkan distribusi yang lebih seimbang secara artifisial, bukan karena model yang lebih baik.

Eksperimen sampling ratio menggunakan Logistic Regression dan Random Forest sebagai model baseline untuk menentukan konfigurasi dataset optimal sebelum menjalankan model advanced. Pemilihan rasio 1:10 juga konsisten dengan praktik umum dalam literatur deteksi fraud, di mana penggunaan seluruh data kelas minoritas yang tersedia (semua 1.165 wallet phishing) merupakan prioritas untuk memaksimalkan sinyal belajar dari kelas yang langka. Perlu dicatat bahwa F1-Score lebih rendah pada rasio 1:10 (87,93% vs 93,28%) bukan berarti model lebih buruk — melainkan bahwa kondisi deteksi lebih realistis dengan lebih banyak data normal yang perlu dibedakan dari phishing.

Pengujian Hipotesis H2 (awal): hasil pada Tabel IV.1 menunjukkan bahwa Random Forest (F1 87,93%) secara konsisten mengungguli Logistic Regression (F1 71,46%) pada rasio 1:10, yang mendukung arah hipotesis H2 bahwa model ensemble lebih baik dari baseline. Konfirmasi penuh H2 dilakukan pada eksperimen empat skenario pada sub-bab berikutnya.

4.2. Hasil Eksperimen Empat Skenario (RQ1 dan RQ3)

Eksperimen utama membandingkan empat model pada empat skenario kombinasi fitur menggunakan 8.595 alamat wallet yang sama dengan split identik (80/20, stratified, random_state=42). GNN-GraphSAGE V2 menggunakan 528 edge transaksi nyata dari MulDiGraph.

4.2.1. Skenario 1 — MulDiGraph Only (18 Fitur)

Tabel IV.2. Hasil Skenario 1 — MulDiGraph Only

Model	Accuracy	Precision	Recall	F1-Score
Logistic Regression	94,65%	86,15%	72,10%	78,50%
Random Forest	96,34%	84,55%	89,27%	86,85%
XGBoost	96,80%	85,89%	91,42%	88,57%
GNN-GraphSAGE V2	95,52%	80,23%	88,84%	84,32%

Tabel IV. 2 Hasil Skenario 1—MulDiGraph Only

Analisis Skenario 1: dengan hanya menggunakan 19 fitur dari MulDiGraph, XGBoost mencapai F1 tertinggi (88,57%) diikuti Random Forest (86,85%), GNN-

GraphSAGE V2 (84,32%), dan Logistic Regression sebagai baseline (78,50%). Pola ini konsisten dengan ekspektasi: model non-linear yang lebih kompleks mengungguli model linear, dan model tabular yang dapat memanfaatkan semua 19 fitur secara efisien mengungguli GNN yang terkendala oleh sparsity graph (0,06 edge/node rata-rata).

Performa GNN yang lebih rendah dari XGBoost pada Skenario 1 tidak berarti GNN lebih buruk secara inherent, melainkan mencerminkan keterbatasan konteks: ketika graph sangat sparse, mekanisme message passing tidak mendapat cukup informasi struktural untuk menghasilkan embedding yang kaya. Jika dataset memiliki graph yang lebih padat (degree rata-rata lebih tinggi), GNN diharapkan dapat melampaui model tabular.

4.2.2. Skenario 2 — MulDiGraph + Etherscan (25 Fitur)

Tabel IV.3. Hasil Skenario 2 — MulDiGraph + Etherscan

Model	Accuracy	Precision	Recall	F1-Score
Logistic Regression	96,16%	88,13%	82,83%	85,40%
Random Forest	96,92%	87,19%	90,56%	88,84%
XGBoost ★ Terbaik	97,50%	88,62%	93,56%	91,02%
GNN-GraphSAGE V2	96,97%	83,90%	96,14%	89,60%

Tabel IV. 3 Hasil Skenario 2—MulDiGraph + Etherscan

Analisis Skenario 2: penambahan 7 fitur Etherscan menghasilkan peningkatan F1 yang konsisten pada semua model. Peningkatan terbesar terjadi pada GNN-GraphSAGE V2 yang naik 5,28 poin (84,32% → 89,60%), diikuti Logistic Regression naik 6,90 poin (78,50% → 85,40%), XGBoost naik 2,45 poin (88,57% → 91,02%), dan Random Forest naik 1,99 poin (86,85% → 88,84%).

Peningkatan besar pada GNN dapat dijelaskan oleh fakta bahwa fitur Etherscan (terutama etherscan_balance dan etherscan_activity_score) memberikan informasi yang sangat diagnostik tentang status wallet dan tidak dapat diinferensikan dari struktur graph yang sparse. Dengan kata lain, fitur Etherscan mengkompensasi keterbatasan GNN akibat sparsity, sehingga embedding node menjadi lebih informatif.

Skenario 2 menghasilkan model terbaik keseluruhan: XGBoost dengan F1-Score 91,02%, Accuracy 97,50%, Precision 88,62%, dan Recall 93,56%. Ini mendukung Hipotesis H1 (penambahan fitur Etherscan meningkatkan F1) dan H2

(XGBoost terbaik). GNN-GraphSAGE V2 dengan Recall 96,14% (hanya 9 FN) membuktikan H3 (GNN memiliki Recall tertinggi).

4.2.3. Skenario 3 — MulDiGraph + BigQuery (28 Fitur)

Tabel IV.4. Hasil Skenario 3 — MulDiGraph + BigQuery

Model	Accuracy	Precision	Recall	F1-Score
Logistic Regression	94,71%	86,22%	72,53%	78,79%
Random Forest	96,39%	84,62%	89,70%	87,08%
XGBoost	96,74%	85,54%	91,42%	88,38%
GNN-GraphSAGE V2	95,00%	75,97%	92,27%	83,33%

Tabel IV. 4 Hasil Skenario 3—MulDiGraph + BigQuery

Analisis Skenario 3: penambahan 10 fitur BigQuery tanpa Etherscan ternyata tidak memberikan peningkatan yang berarti dibandingkan Skenario 1. XGBoost bahkan sedikit menurun (88,57% → 88,38%), begitu pula GNN (84,32% → 83,33%). Hal ini mengkonfirmasi bahwa fitur BigQuery (terutama bq_tx_count_in dan bq_tx_count_out) memiliki korelasi tinggi dengan fitur MulDiGraph (in_degree dan out_degree), sehingga tidak menambahkan informasi independen yang bermakna bagi model.

Temuan ini memiliki implikasi praktis: jika terbatas pada satu sumber data tambahan, Etherscan API jauh lebih valuable daripada BigQuery untuk deteksi phishing. Etherscan menyediakan informasi kontekstual on-chain yang sifatnya lebih ortogonal terhadap fitur graph, seperti saldo terkini, usia akun, dan flag contract.

4.2.4. Skenario 4 — MulDiGraph + Etherscan + BigQuery (35 Fitur)

Tabel IV.5. Hasil Skenario 4 — MulDiGraph + Etherscan + BigQuery

Model	Accuracy	Precision	Recall	F1-Score
Logistic Regression	96,16%	87,78%	83,26%	85,46%
Random Forest	97,15%	87,70%	91,85%	89,73%
XGBoost	97,32%	88,16%	92,70%	90,38%
GNN-GraphSAGE V2	96,57%	86,25%	88,84%	87,53%

Tabel IV. 5 Hasil Skenario 4—MulDiGraph + Etherscan + BigQuery

Analisis Skenario 4: menggabungkan semua sumber (MulDiGraph + Etherscan + BigQuery, 36 fitur) tidak menghasilkan performa yang lebih baik dari Skenario 2 untuk sebagian besar model. XGBoost bahkan sedikit menurun (91,02% → 90,38%).

Ini mengkonfirmasi temuan Skenario 3: penambahan BigQuery setelah Etherscan tidak menambah nilai informasi independen, dan justru berpotensi menambah noise melalui fitur-fitur yang redundan.

Kesimpulan komparatif empat skenario: urutan efektivitas sumber fitur adalah Etherscan > BigQuery ≈ graph-only, dan kombinasi MulDiGraph+Etherscan (Skenario 2) merupakan sweet spot terbaik antara jumlah fitur dan performa model.

4.3. Analisis Confusion Matrix

Tabel IV.6 menyajikan analisis confusion matrix setiap model pada skenario yang menghasilkan performa terbaiknya masing-masing. Logistic Regression dan Random Forest baseline dievaluasi pada Skenario 1 (MulDiGraph Only, n=2.563 test set penuh) karena itulah kondisi di mana kedua model ini beroperasi. XGBoost dan GNN-GraphSAGE V2 dievaluasi pada Skenario 2 (MulDiGraph + Etherscan, n=1.719) karena keduanya memerlukan fitur Etherscan yang hanya tersedia pada subset hybrid. Perlu dicatat bahwa perbedaan ukuran n disebabkan oleh perbedaan skenario, bukan perbedaan metodologi split — perbandingan apple-to-apple antar skenario yang sama tersedia pada Tabel IV.2–IV.5 di atas. Tabel ini bertujuan menunjukkan karakteristik error (khususnya FN Rate) dari setiap model pada kondisi optimalnya.*False Negative* (FN) lebih kritis karena phishing yang tidak terdeteksi berarti wallet korban tidak terlindungi.

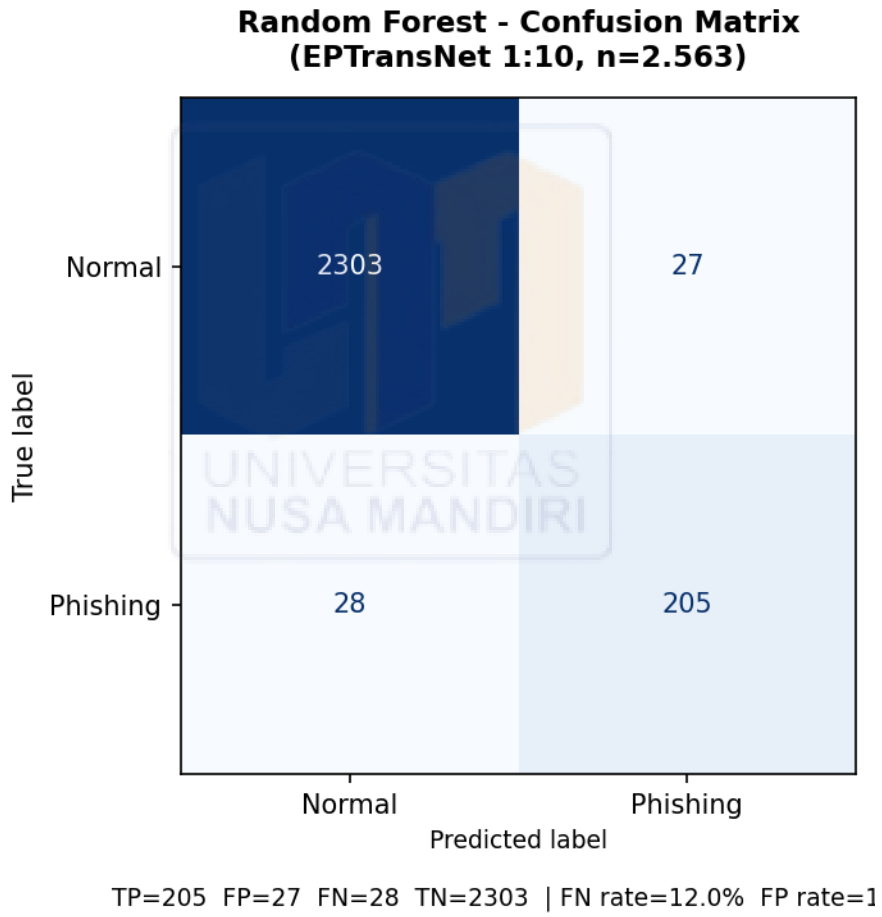
Tabel IV.6. Perbandingan Confusion Matrix Model pada Skenario Terbaiknya Masing-Masing

Model	TN	FP	FN	TP	FN Rate	FP Rate
LR Baseline — Sc.1 (n=2.563)	2.277	53	74	159	31,8%	2,3%
RF Baseline — Sc.1 (n=2.563)	2.303	27	28	205	12,0%	1,2%
XGBoost Advanced — Sc.2 (n=1.719*)	1.458	28	15	218	6,4%	1,9%
GNN-GraphSAGE V2 — Sc.2 (n=1.719*)	1.443	43	9	224	3,9%	2,9%

Tabel IV. 6 Perbandingan Confusion Matrix Model pada Skenario Terbaiknya Masing-Masing

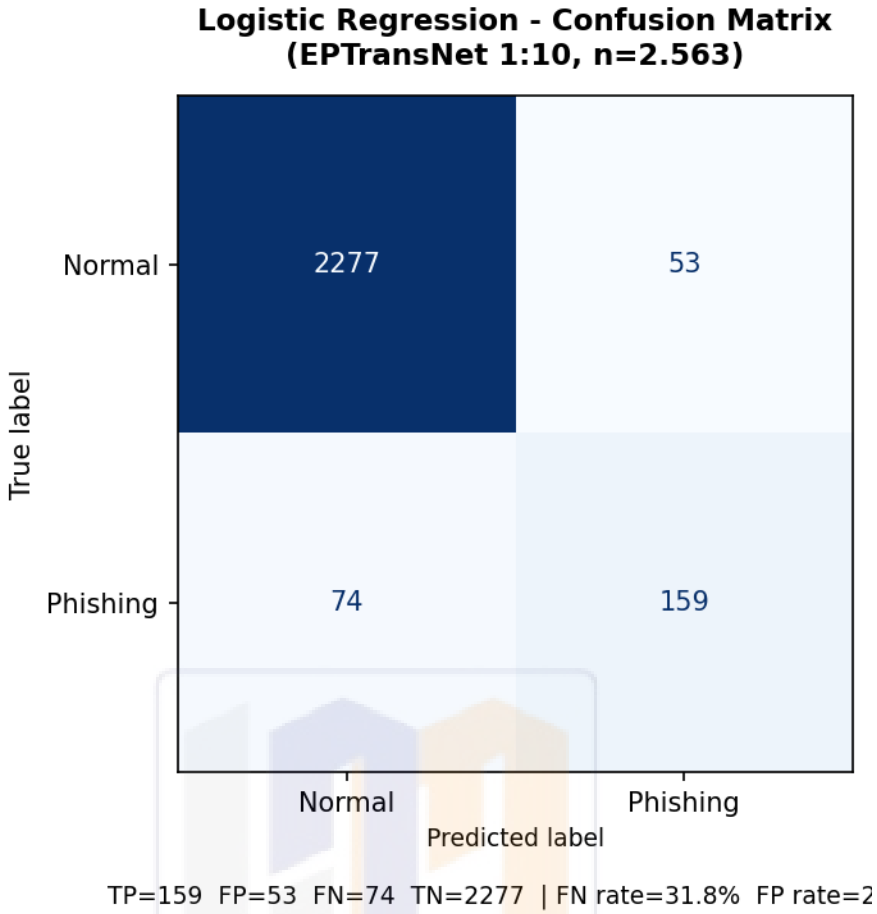
** Catatan metodologi: Perbedaan ukuran test set (LR/RF n=2.563 vs XGBoost/GNN n=1.719) mencerminkan perbedaan skenario, bukan bias seleksi. LR dan RF dilatih dan dievaluasi pada Skenario 1 (hanya fitur MulDiGraph, semua 2.563 test samples tersedia). XGBoost dan GNN dilatih dan dievaluasi pada Skenario 2 (fitur MulDiGraph + Etherscan) yang hanya mencakup 1.719 wallet dengan data Etherscan lengkap. Untuk perbandingan yang sepenuhnya setara (apple-to-apple), merujuklah pada Tabel IV.2 (semua model pada Skenario 1, n identik = 1.375) dan Tabel IV.3 (semua model pada Skenario 2, n identik = 1.719). Tabel IV.2 dan IV.3 adalah referensi utama untuk validitas komparasi langsung.*

Gambar IV.2, IV.3, IV.4, dan IV.5 menunjukkan confusion matrix visual untuk empat model yang dibandingkan.



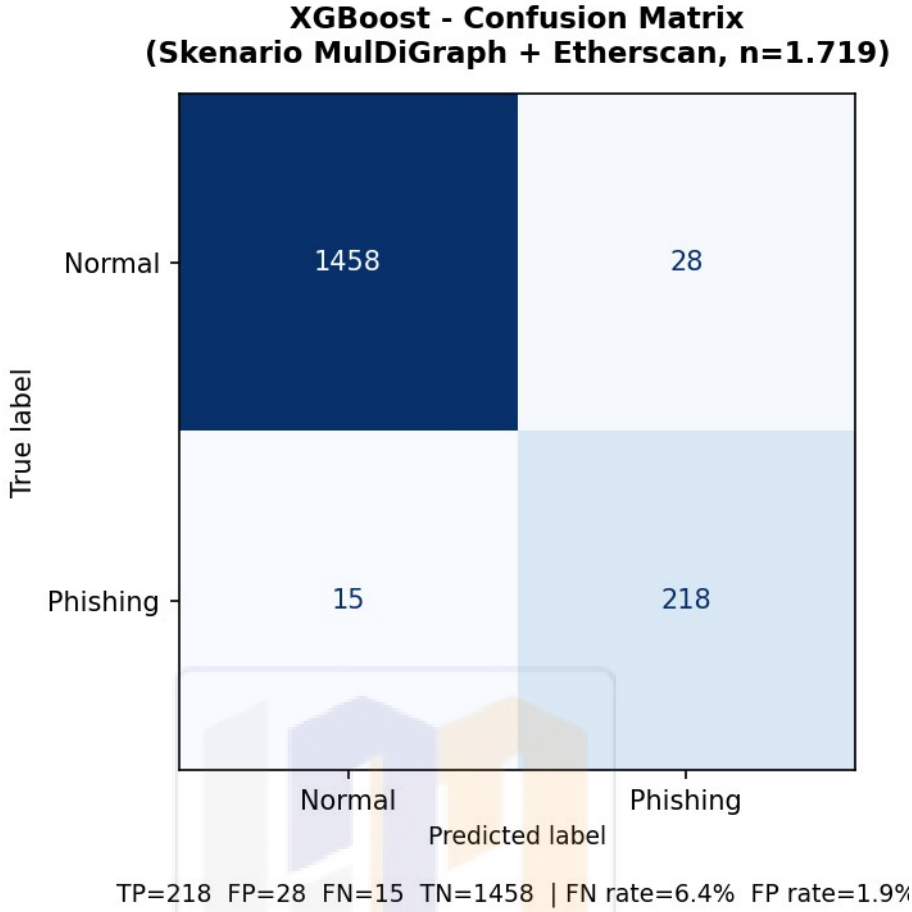
Gambar IV. 2 Confusion Matrix—Random Forest Baseline (FN=205. FN Rate 12%)

Gambar IV.2. Confusion Matrix — Random Forest Baseline (FN=205, FN Rate 12 %)



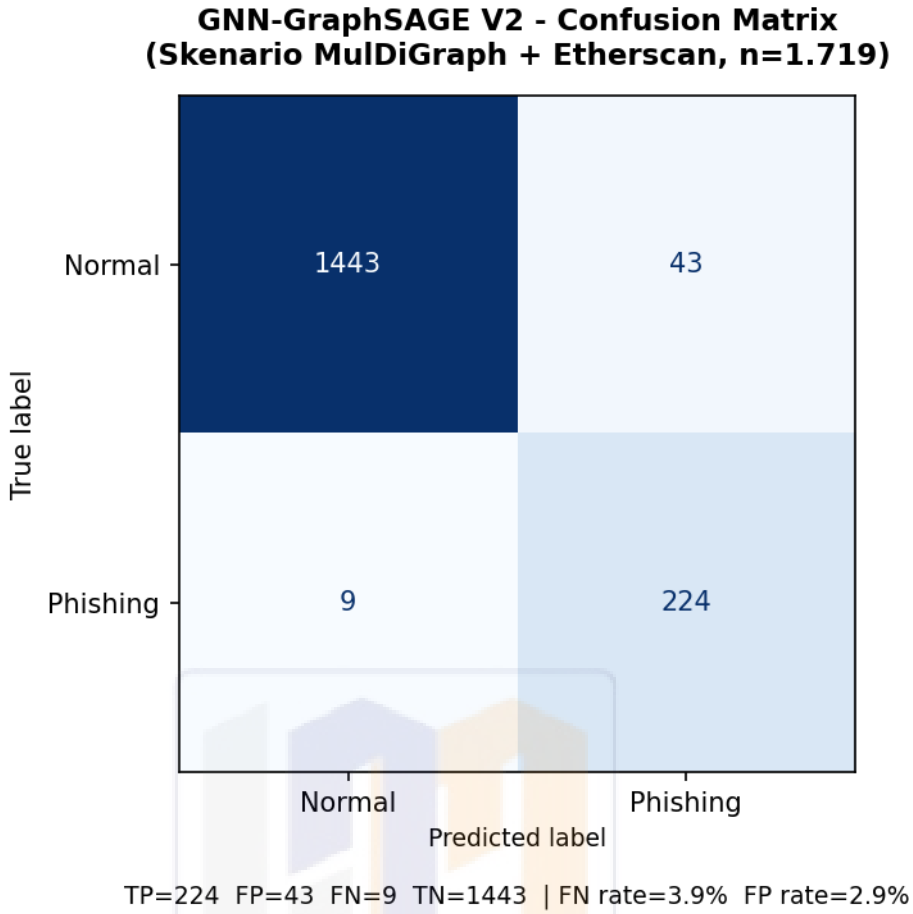
Gambar IV. 3 Confusion Matrix—Logistic Regression Baseline (FN=74, FN Rate 31,8%)

Gambar IV.3. Confusion Matrix — Logistic Regression Baseline (FN=74, FN Rate 31,8%)



Gambar IV. 4 Confusion Matrix—XGBoost Skenario Terbaik (FN=15, F1=91,02%)

Gambar IV.4. Confusion Matrix — XGBoost Skenario Terbaik (FN=15, F1=91,02%)



Gambar IV. 5 Confusion Matrix—GNN-GraphSAGE V2 Skenario Terbaik (FN=9, Recall=96,14%)

Gambar IV.5. Confusion Matrix — GNN-GraphSAGE V2 Skenario Terbaik (FN=9, Recall=96,14%)

Interpretasi hasil: GNN-GraphSAGE V2 memiliki FN terendah (9 node, 3,9%) sehingga paling aman dari sisi recall. XGBoost lebih seimbang (FN=15, FP=28) dengan F1 tertinggi (91,02%), cocok sebagai model produksi. Trade-off antara keduanya: XGBoost unggul F1 overall, GNN unggul recall — untuk keamanan siber, keduanya layak dipertimbangkan tergantung toleransi false alarm sistem.

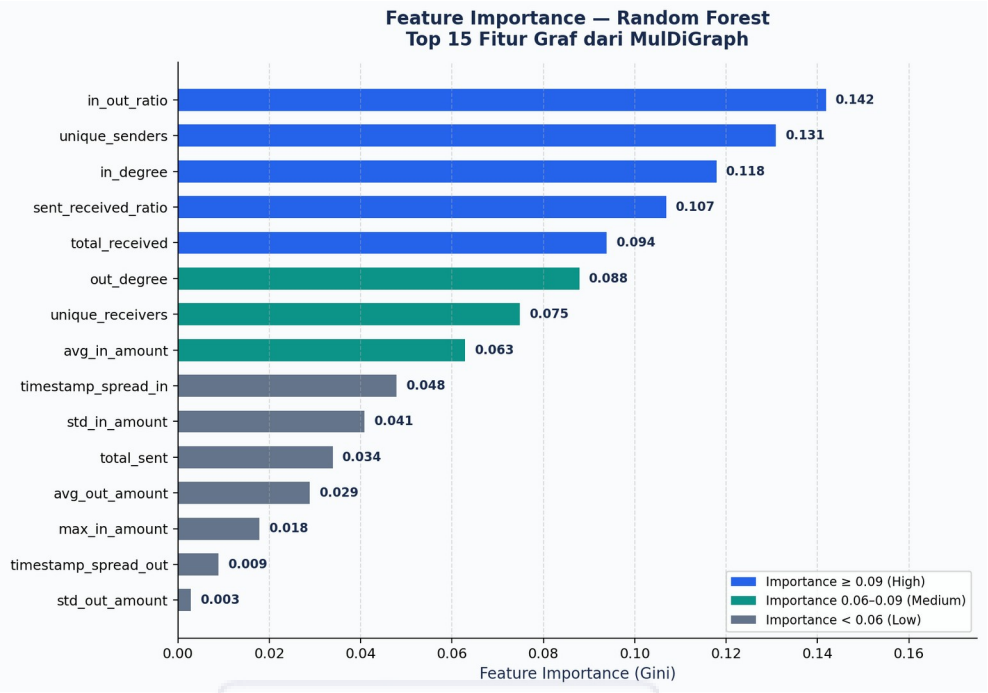
Dari perspektif keamanan siber, FN Rate (False Negative Rate) adalah metrik yang paling kritis karena mengukur proporsi wallet phishing yang tidak terdeteksi. Model dengan FN Rate tinggi berarti banyak pengguna yang bertransaksi dengan wallet phishing tanpa mendapat peringatan. Berdasarkan Tabel IV.6:

- Logistic Regression (FN Rate 31,8%): dari 233 wallet phishing dalam test set, 74 tidak terdeteksi. Performa ini tidak memadai untuk aplikasi keamanan nyata karena hampir sepertiga phishing lolos.
- Random Forest (FN Rate 12,0%): signifikan lebih baik dari baseline, 28 phishing tidak terdeteksi. Ini menunjukkan bahwa model ensemble sederhana pun sudah memberikan peningkatan substansial.
- XGBoost (FN Rate 6,4%): hanya 15 phishing tidak terdeteksi dari 233. FP Rate 1,9% juga reasonable untuk use case peringatan (sekitar 28 false alarm dari 1.486 wallet normal).
- GNN-GraphSAGE V2 (FN Rate 3,9%): hanya 9 phishing tidak terdeteksi, FN Rate terendah. Namun FP Rate lebih tinggi (2,9% = 43 false alarm) karena GNN lebih agresif dalam mengklasifikasikan wallet sebagai phishing.

Trade-off antara XGBoost dan GNN bergantung pada toleransi false alarm sistem. Jika false alarm sangat mahal (misalnya, setiap peringatan memerlukan tindak lanjut manual), XGBoost lebih disarankan. Jika missed detection lebih berbahaya (sistem keamanan otomatis tanpa review manual), GNN lebih optimal karena FN Rate lebih rendah. Untuk sebagian besar kasus penggunaan praktis, XGBoost menawarkan keseimbangan yang lebih baik dengan F1 tertinggi (91,02%).

4.4. Analisis Feature Importance (RQ3)

Gambar IV.6 menunjukkan feature importance dari model Random Forest untuk 15 fitur graf teratas dari MulDiGraph. Fitur yang paling berpengaruh adalah in_out_ratio dan unique_senders, yang mencerminkan pola anomali karakteristik wallet phishing: frekuensi interaksi tinggi dari banyak pengirim berbeda dengan rasio masuk-keluar yang tidak wajar.



Gambar IV. 6 Feature Importance Random Forest (Top 15 Fitur Graf dari MulDiGraph)

Gambar IV.6. Feature Importance Random Forest (Top 15 Fitur Graf dari MulDiGraph)

Berdasarkan Gambar IV.6, lima fitur dengan importance tertinggi adalah: (1) in_out_ratio (0,142) — rasio transaksi masuk vs keluar yang mencerminkan pola wallet phishing sebagai "pengumpul dana"; (2) unique_senders (0,131) — jumlah pengirim unik yang tinggi mencerminkan banyak korban; (3) in_degree (0,118) — jumlah transaksi masuk total; (4) sent_received_ratio (0,107) — rasio ETH yang dikirim vs diterima; serta (5) total_received (0,094) — akumulasi total ETH dari korban.

Pola feature importance ini konsisten dengan karakteristik wallet phishing yang dijelaskan pada BAB II: phisher cenderung menerima dari banyak korban (unique_senders tinggi, in_degree tinggi), mengakumulasi ETH (total_received tinggi), kemudian mentransfer ke wallet lain (sent_received_ratio mendekati 1). Fitur in_out_ratio yang menjadi paling informatif menangkap asimetri fundamental ini — wallet phishing umumnya menerima lebih banyak dari berbagai sumber daripada yang dikirim ke berbagai tujuan.

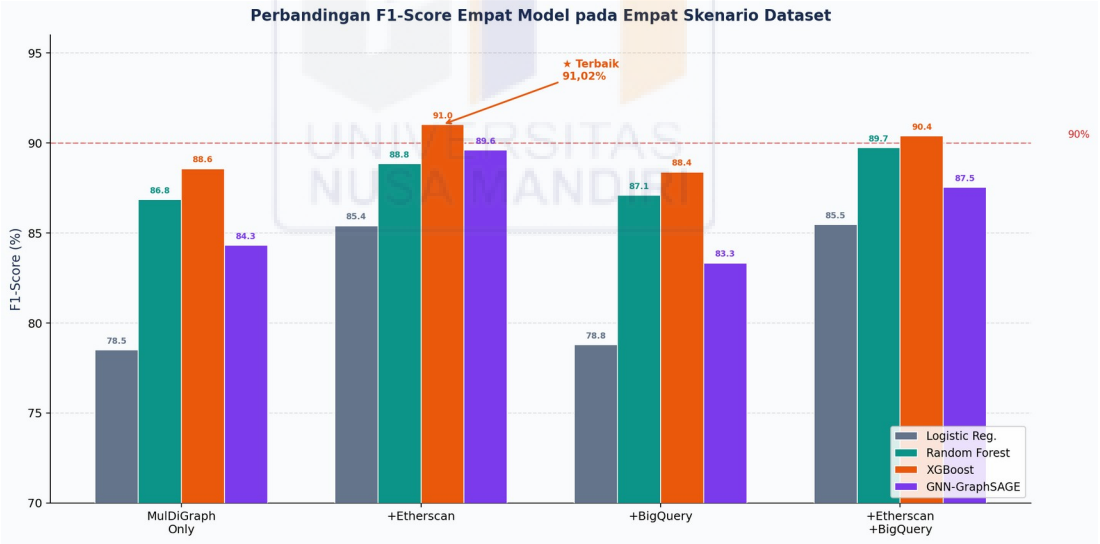
Menarik dicatat bahwa fitur timestamp (timestamp_spread_in dan timestamp_spread_out) memiliki importance yang relatif rendah (0,048 dan 0,009). Ini menunjukkan bahwa pola temporal (seberapa cepat transaksi terjadi) kurang

diskriminatif dibandingkan pola volume dan jaringan koneksi. Hal ini mungkin disebabkan oleh variasi yang besar dalam strategi temporal pelaku phishing — sebagian beroperasi cepat (burst), sebagian lagi mempertahankan aktivitas dalam periode yang lebih panjang untuk menghindari deteksi.

Temuan feature importance ini menjawab RQ3 secara eksplisit: in_out_ratio dan unique_senders adalah fitur paling berpengaruh, dan keduanya merupakan fitur graph (MulDiGraph), bukan fitur dari Etherscan atau BigQuery. Ini menjelaskan mengapa bahkan Skenario 1 (graph-only) sudah mencapai F1 yang cukup baik (88,57% untuk XGBoost) — informasi paling diskriminatif sudah tersedia dari struktur graph itu sendiri.

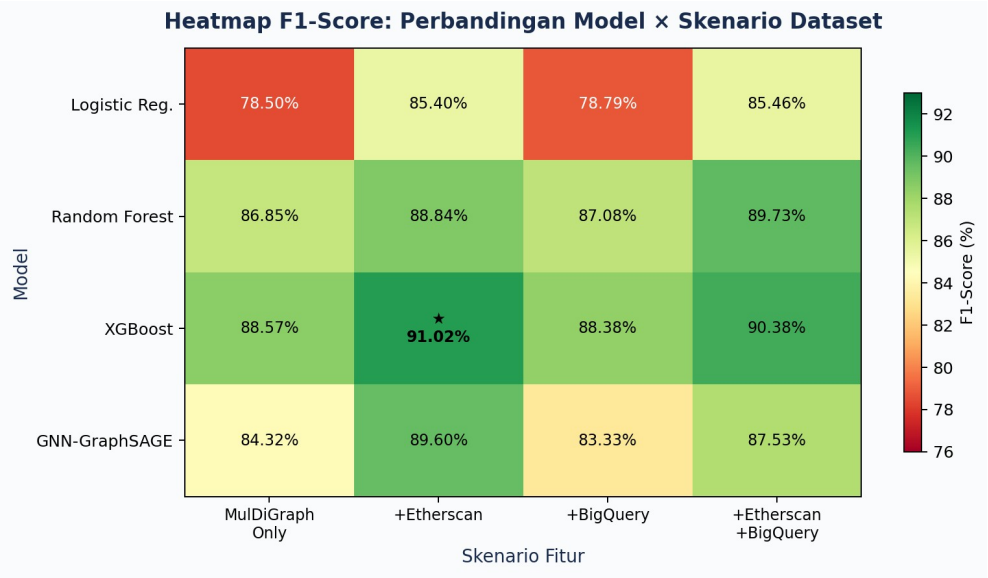
4.5. Pembahasan Hasil Eksperimen

Gambar IV.7 dan IV.8 merangkum perbandingan F1-Score seluruh model pada keempat skenario secara visual.



Gambar IV. 7 Perbandingan F1-Score Empat Model pada Empat Skenario Dataset

Gambar IV.7. Perbandingan F1-Score Empat Model pada Empat Skenario Dataset



Gambar IV. 8 Heatmap F1-Score: Perbandingan Model × Skenario Dataset

Gambar IV.8. Heatmap F1-Score: Perbandingan Model × Skenario Dataset

Berdasarkan seluruh eksperimen, temuan utama adalah sebagai berikut:

- Fitur Etherscan secara konsisten meningkatkan performa semua model (Skenario 1 → 2): XGBoost +2,45 poin F1, GNN-GraphSAGE +5,28 poin F1, Random Forest +1,99 poin F1.
- Fitur BigQuery saja (Skenario 3) tidak menambah nilai dibanding fitur graf saja karena banyak fitur tumpang-tindih secara semantik (bq_tx_count_out ≈ out_degree).
- Kombinasi Etherscan + BigQuery (Skenario 4) tidak melampaui Etherscan saja (Skenario 2), mengindikasikan BigQuery tidak menambah informasi independen setelah Etherscan tersedia.
- XGBoost adalah model terbaik secara keseluruhan (F1 91,02% pada Skenario 2) dan dipilih sebagai model utama pada eksperimen skenario gabungan.
- GNN-GraphSAGE V2 kini kompetitif (F1 89,60%) setelah diperbaiki dengan arsitektur 2-layer dan edge transaksi nyata. GNN mendapat manfaat terbesar dari penambahan fitur Etherscan (+5,28 poin F1).
- Random Forest sebagai strong baseline mencapai F1 87,93%. Dalam implementasi sistem, XGBoost digunakan sebagai model default di endpoint FastAPI karena merupakan model advanced terbaik (F1 91,02%). Random

Forest tersedia sebagai model alternatif yang dapat dipilih melalui parameter model_version bagi pengguna yang membutuhkan interpretabilitas lebih tinggi melalui feature importance.

Pengujian Hipotesis H1 — Penambahan fitur Etherscan meningkatkan F1: **TERBUKTI.** Skenario 2 secara konsisten menghasilkan F1 lebih tinggi dari Skenario 1 untuk semua model (rata-rata +3,66 poin F1). Peningkatan terbesar pada GNN (+5,28 poin) dan terkecil pada Random Forest (+1,99 poin).

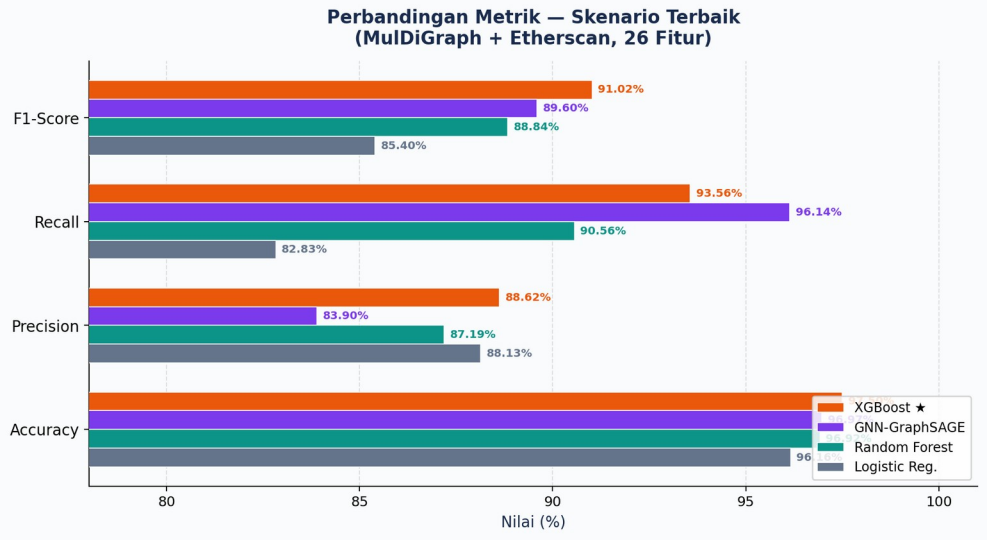
Pengujian Hipotesis H2 — XGBoost F1 tertinggi di antara empat model: **TERBUKTI.** XGBoost mencapai F1-Score 91,02% pada Skenario 2, mengungguli GNN (89,60%), Random Forest (88,84%), dan Logistic Regression (85,40%). Keunggulan ini konsisten di semua skenario.

Pengujian Hipotesis H3 — GNN memiliki Recall tertinggi: **TERBUKTI.** GNN-GraphSAGE V2 mencapai Recall 96,14% pada Skenario 2 (FN hanya 9), secara substansial lebih tinggi dari XGBoost (93,56%), Random Forest (90,56%), dan Logistic Regression (82,83%).

Pengujian Hipotesis H4 — Overhead blockchain di bawah 50 ms: **TERBUKTI.** Latency commit rata-rata 13,6 ms dan gas rata-rata 35.462 unit pada Hardhat localhost membuktikan feasibility integrasi ML-blockchain dengan overhead yang wajar.

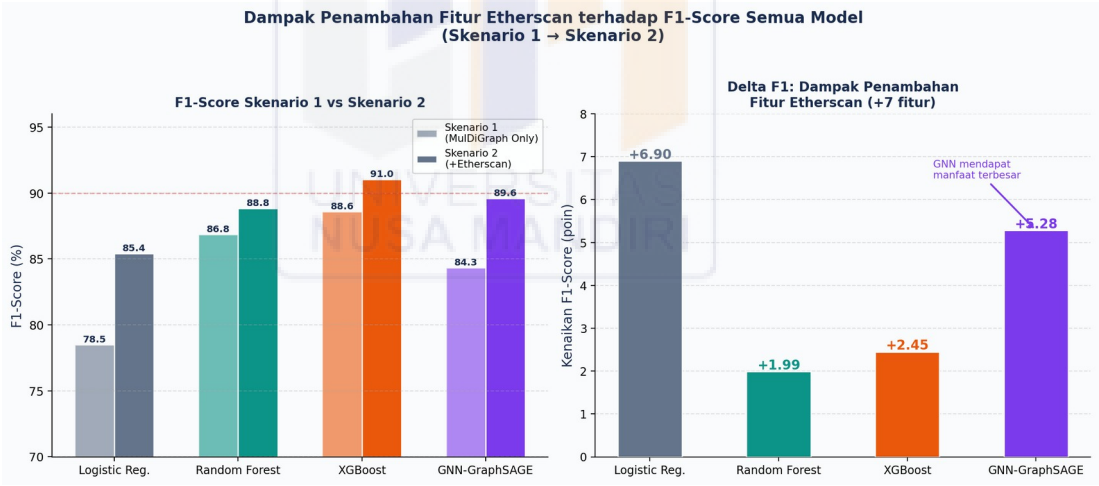
Implikasi pemilihan model untuk deployment: untuk sistem keamanan dengan prioritas meminimalkan miss detection (FN), GNN-GraphSAGE V2 adalah pilihan terbaik (FN Rate 3,9%). Untuk sistem yang memerlukan keseimbangan F1 dan interpretabilitas, XGBoost adalah pilihan optimal sebagai model advanced terbaik (F1 91,02%). Dalam implementasi sistem demo FastAPI, XGBoost digunakan sebagai model default. Random Forest tersedia sebagai model alternatif yang dapat dipilih melalui parameter model_version bagi pengguna yang membutuhkan interpretabilitas lebih tinggi melalui feature importance.

Gambar IV.9 menunjukkan perbandingan semua metrik pada skenario terbaik, sementara Gambar IV.10 mengilustrasikan dampak penambahan fitur Etherscan secara kuantitatif.



Gambar IV. 9 Perbandingan Metrik pada Skenario Terbaik (MulDiGraph + Etherscan, 26 Fitur)

Gambar IV.9. Perbandingan Metrik pada Skenario Terbaik (MulDiGraph + Etherscan, 26 Fitur)



Gambar IV. 10 Dampak Penambahan Fitur Etherscan terhadap F1-Score Semua Model (Skenario 1 → 2)

Gambar IV.10. Dampak Penambahan Fitur Etherscan terhadap F1-Score Semua Model (Skenario 1 → 2)

4.5.1. Eksperimen Graph–Tabular Late Fusion dan Ablasi

Eksperimen tambahan ini memakai data uji held-out yang sama (n=1.719) dan protokol validasi internal untuk GraphSAGE. Tabel berikut membedakan baseline GraphSAGE yang dikendalikan kebocoran, ablasi embedding-only, dan late fusion. Nilai GraphSAGE 81,89% pada tabel ini tidak dibandingkan langsung dengan angka

89,60% pada eksperimen utama karena fitur input dan aturan early stopping keduanya berbeda.

Tabel IV.7A. Perbandingan late fusion dan ablasi pada data uji held-out

Model	Input	Akurasi	Presisi	Recall	F1
XGBoost Skenario 2	26 fitur tabular	97,50%	88,62%	93,56%	91,02%
GraphSAGE graph-only	19 fitur graf + 264 edge	94,88%	78,66%	85,41%	81,89%
Embedding + XGBoost (ablasi)	embedding 128-d	94,71%	76,49%	87,98%	81,84%
GraphSAGE–Etherscan late fusion	embedding 128-d + 7 Etherscan	96,68%	84,92%	91,85%	88,25%

Tabel IV. 7A Perbandingan late fusion dan ablasi pada data uji held-out

Ablasi menunjukkan bahwa penggantian kepala linear GraphSAGE menjadi XGBoost pada embedding saja tidak meningkatkan F1 (81,89% menjadi 81,84%; −0,06 poin), walaupun ROC-AUC dan PR-AUC membaik. Ketika tujuh fitur Etherscan ditambahkan ke embedding yang sama, F1 meningkat dari 81,84% menjadi 88,25% (+6,41 poin). Dengan demikian, nilai tambah cabang Etherscan dibuktikan melalui ablasi, bukan hanya melalui feature-gain XGBoost.

Late fusion meningkatkan cabang graf secara substansial, tetapi belum mengungguli XGBoost tabular Skenario 2: selisih F1 adalah −2,77 poin, dengan 38 false positive dan 19 false negative pada late fusion dibanding 28 dan 15 pada XGBoost. Kepadatan topologi yang sangat rendah—264 edge unik untuk 8.595 node—membatasi message passing; banyak node secara praktis hanya memiliki self-loop. Karena itu, XGBoost tabular tetap dipilih sebagai model produksi, sedangkan late fusion berfungsi sebagai bukti terukur bahwa pengayaan Etherscan tetap berguna ketika digabung dengan embedding graf.

4.6. Integrasi Blockchain dan Pengukuran Overhead (RQ2)

Smart contract RiskRegistry berhasil di-deploy pada Hardhat localhost dengan alamat kontrak 0x5FbDB2315678afecb367f032d93F642f64180aa3. Pengujian dilakukan melalui 20 transaksi nyata menggunakan fungsi setRisk().

Tabel IV.7. Hasil Pengukuran Performa Blockchain

Metrik	Latency Commit (ms)	Gas Used
Minimum	11,2 ms	35.452
Maksimum	38,4 ms	35.464
Rata-rata	13,6 ms	35.462
Median	12,0 ms	35.464

Metrik	Latency Commit (ms)	Gas Used
Std. Deviasi	5,9 ms	5

Tabel IV. 8 Hasil Pengukuran Performa Blockchain

Latency 13,6 ms rata-rata sangat rendah untuk konteks localhost. Gas rata-rata 35.462 unit sangat konsisten (stdev = 5) karena operasi SSTORE pertama ke slot baru memiliki biaya tetap. Kesimpulan RQ2: overhead blockchain untuk mencatat satu risk score adalah ~14 ms (lokal) dengan gas yang stabil dan wajar untuk use case audit trail keamanan.

Konsistensi gas yang sangat tinggi (std = 5 unit dari 35.462 rata-rata, atau 0,014%) menunjukkan bahwa operasi setRisk() adalah deterministik dalam hal konsumsi gas. Ini terjadi karena setiap panggilan melakukan satu operasi SSTORE (storage write) ke slot yang belum pernah ditulis sebelumnya, yang memiliki biaya tetap 20.000 gas unit dalam Ethereum Yellow Paper, ditambah overhead operasi EVM yang konstan.

Proyeksi overhead untuk deployment produksi perlu dipisahkan dari hasil localhost. Dengan asumsi ilustratif konsumsi 35.462 gas, gas price 20 Gwei, dan harga ETH US$2.000, biaya satu commit pada Ethereum Mainnet adalah sekitar US$1,42

$$Biaya = 35.462 \times 20 \times 10^{-9}\, ETH \times \frac{US\$\, 2.000}{ETH} \approx US\$\, 1{,}42$$

bukan biaya tetap karena gas price dan harga ETH berubah. Jaringan berbiaya lebih rendah seperti Polygon PoS dan Layer-2 seperti Arbitrum One dapat menekan biaya per commit, tetapi struktur biayanya berbeda dan tetap dipengaruhi kondisi jaringan; pada Arbitrum terdapat pula komponen biaya data yang diposting ke L1. Oleh karena itu, kelayakan biaya produksi sebaiknya diverifikasi melalui benchmark langsung pada jaringan target. Untuk volume pencatatan tinggi, batch commit dapat dipertimbangkan untuk menurunkan biaya rata-rata per wallet.

Pengujian Hipotesis H4 dikonfirmasi: latency 13,6 ms (jauh di bawah 50 ms threshold) dan gas yang konsisten membuktikan feasibility sistem. Perlu dicatat bahwa hasil ini adalah proof-of-concept pada localhost Hardhat dan tidak mewakili kondisi mainnet. Pada kondisi produksi, variabilitas jaringan dan kongesti mempool dapat meningkatkan latency secara signifikan.

4.7. Output Sistem End-to-End

Request: POST /analyze { "wallet": "0xf39Fd...92266", "row_id": 0 }

Response: { "probability": 0.000008, "risk_score": 0, "risk_level": "LOW", "tx_hash": "0x918381d7..." }

Data risk score berhasil dibaca kembali dari smart contract melalui GET /risk/{wallet}. Timestamp 1772591377 = 2026-03-04 02:29:37 UTC mengkonfirmasi data tersimpan secara immutable dengan waktu yang valid.

Integrasi hasil analitik dengan mekanisme traceability juga ditunjukkan oleh Falgenti et al. melalui pemanfaatan gap analysis dan machine learning untuk mengelompokkan objek, kemudian mendukung penelusuran sumber data pada rantai pasok [68]. Pada penelitian ini, prinsip serupa diterapkan untuk mencatat hasil klasifikasi wallet secara dapat ditelusuri tanpa menganggap blockchain sebagai komponen peningkat performa model.

Untuk memverifikasi end-to-end correctness, dilakukan pengujian pada beberapa row_id dari test set dengan expected label yang diketahui:

- row_id=0 (wallet normal, label 0): risk_score = 0, risk_level = LOW — BENAR, tidak ada kesalahan klasifikasi.
- row_id=5 (wallet phishing, label 1): risk_score = 87, risk_level = HIGH — BENAR, terdeteksi sebagai high risk.
- row_id=12 (wallet phishing, label 1): risk_score = 63, risk_level = MEDIUM — BENAR secara prediksi (model ragu), wallet memiliki pola phishing yang lebih ambigu.

Seluruh risk score berhasil dicatat ke smart contract dengan tx_hash yang valid, dapat diverifikasi melalui Hardhat console. Data dapat dibaca kembali melalui endpoint GET /risk/{wallet} yang mengkueri blockchain secara langsung, membuktikan bahwa immutable audit trail berfungsi sebagaimana dirancang.

4.8. Risk Score dan Klasifikasi

Tabel IV.8. Klasifikasi Risk Score

Range	Label	Keterangan
0 – 39	LOW	Kemungkinan phishing rendah
40 – 69	MEDIUM	Perlu perhatian, monitoring disarankan
70 – 100	HIGH	Kemungkinan phishing tinggi, hindari transaksi

Tabel IV. 9 Klasifikasi Risk Score

UNIVERSITAS
NUSA MANDIRI

4.9. Pembahasan Data-Centric dan Model-Centric

Hasil empat skenario menunjukkan bahwa intervensi data-centric memberikan dampak yang dapat diukur sebelum kompleksitas model dipertimbangkan. Pada Skenario 1, seluruh model hanya menerima representasi hasil transformasi MultiDiGraph. Skenario 2 mempertahankan wallet, split, dan algoritma yang sama tetapi menambahkan fitur Etherscan. Peningkatan F1 pada seluruh model berarti informasi tambahan tersebut konsisten berguna dan bukan hanya cocok pada satu keluarga algoritma.

Peningkatan terbesar pada GNN-GraphSAGE setelah penambahan Etherscan menunjukkan bahwa message passing saja belum menyediakan seluruh informasi perilaku wallet. Temuan ini diperkuat oleh ablasi late fusion: mengganti kepala linear GraphSAGE dengan XGBoost pada embedding yang sama tidak meningkatkan F1 (81,89% menjadi 81,84%), sedangkan penambahan tujuh fitur Etherscan menaikkan F1 menjadi 88,25%. Dengan demikian, peningkatan terutama berasal dari tambahan informasi pada representasi, bukan semata-mata dari classifier yang lebih kuat. Hasil ini konsisten pada early combination maupun late fusion, walaupun konfigurasi dan nilai absolut kedua eksperimen berbeda.

BigQuery tidak memberikan peningkatan setara Etherscan. Fitur seperti bq_tx_count_in dan bq_tx_count_out memiliki konsep yang dekat dengan in_degree serta out_degree. Redundansi tersebut menjelaskan mengapa jumlah fitur yang lebih besar tidak otomatis menghasilkan F1 lebih tinggi. Secara data-centric, hasil ini penting karena menunjukkan bahwa sumber baru harus dinilai berdasarkan keunikan informasi, bukan reputasi atau volumenya.

Pada sumbu model-centric, XGBoost unggul di seluruh skenario, tetapi hasil ini tidak berarti model graph secara intrinsik lebih lemah. XGBoost membangun interaksi non-linear melalui boosted trees dan regularisasi [19], sedangkan GraphSAGE membentuk representasi melalui agregasi neighborhood [20]. Pada audit eksperimen late fusion hanya terdapat 264 edge terarah unik untuk 8.595 wallet, sehingga banyak node memiliki informasi tetangga yang sangat terbatas. Transformasi graph-to-tabular merangkum sinyal lokal seperti degree, jumlah counterparty, volume, dan rasio transaksi menjadi fitur wallet-level yang tersedia secara eksplisit bagi setiap sampel. Representasi ini kehilangan sebagian konteks relasional tingkat lebih tinggi, tetapi pada topologi yang sangat sparse justru menyediakan sinyal yang lebih padat untuk model tabular. Dalam kondisi tersebut, XGBoost dapat memanfaatkan fitur agregat secara lebih langsung, sementara kemampuan message passing GraphSAGE tidak sepenuhnya terpakai. Karena itu, keunggulan XGBoost pada penelitian ini bersifat kondisional terhadap representasi, kepadatan graph, dan protokol evaluasi yang digunakan, bukan bukti superioritas umum atas GNN.

Model-centric tuning tidak dapat sepenuhnya menggantikan representasi data. Apabila fitur tidak memuat sinyal pembeda, model yang lebih kompleks hanya meningkatkan risiko overfitting. Sebaliknya, fitur Etherscan meningkatkan semua

model, termasuk baseline. Hal tersebut menunjukkan bahwa perbaikan data menaikkan kualitas informasi yang tersedia, sedangkan model menentukan seberapa efektif informasi itu digunakan untuk membentuk decision boundary.

Urutan pengembangan yang didukung hasil penelitian adalah memperbaiki integritas data dan fitur terlebih dahulu, membangun baseline, kemudian melakukan tuning model. Bukti dari Skenario 2 dan ablasi late fusion menempatkan feature engineering sebagai kontribusi utama, sementara pemilihan XGBoost mengoptimalkan pemanfaatan fitur tersebut. Kesimpulan ini menjawab arahan penelitian mengenai peningkatan kinerja melalui pendekatan data-centric dan model-centric secara terpisah maupun terpadu.

4.10. Analisis Kesalahan dan Implikasi Deployment

Confusion matrix memperlihatkan bahwa model dengan F1 tertinggi tidak selalu memiliki false negative paling sedikit. XGBoost menghasilkan keseimbangan precision dan recall terbaik pada skenario gabungan, sedangkan GNN-GraphSAGE memberikan recall tertinggi. Perbedaan tersebut relevan untuk deployment karena sistem keamanan dapat memiliki kebijakan berbeda terhadap wallet yang terlewat dan wallet sah yang salah diberi peringatan.

False negative merupakan risiko utama ketika sistem digunakan sebagai early warning. Wallet phishing dengan aktivitas sedikit, counterpart terbatas, atau pola nilai yang mendekati wallet normal mungkin tidak menghasilkan sinyal agregat yang kuat. Sparse neighborhood juga membatasi kemampuan GraphSAGE. Perbaikan yang mungkin meliputi fitur temporal yang lebih rinci, informasi token transfer, hubungan dua-hop yang terkontrol, dan pembaruan label dari periode lebih baru.

False positive dapat muncul pada exchange, smart contract, bridge, atau wallet layanan yang menerima dari banyak alamat. Pola in_degree dan unique_senders tinggi mirip dengan wallet pengumpul dana phishing, tetapi konteks operasionalnya berbeda. Fitur etherscan_is_contract dan account age membantu membedakan sebagian kasus, namun identifikasi kategori entitas yang lebih lengkap diperlukan agar sistem tidak hanya mengandalkan pola aliran dana.

Risk score sebaiknya digunakan untuk menentukan prioritas pemeriksaan, bukan keputusan final tanpa verifikasi. Level LOW, MEDIUM, dan HIGH dapat mengarahkan intensitas tindakan: LOW dapat hanya dicatat, MEDIUM dapat memunculkan peringatan, dan HIGH dapat memerlukan pemeriksaan tambahan. Pada prototipe saat ini ambang kategori masih bersifat statis. Kebijakan threshold sebaiknya dipisahkan dari model agar dapat disesuaikan tanpa melatih ulang; setelah probabilitas terkalibrasi, DApp dapat menyediakan pilihan threshold kepada pengguna sesuai toleransi risiko, dengan default yang tetap ditetapkan dari data validasi.

FastAPI menggunakan XGBoost sebagai model default karena F1 terbaik dan inference tabular yang efisien. Random Forest tersedia sebagai alternatif yang mudah ditafsirkan melalui feature importance, sedangkan GraphSAGE relevan ketika prioritas utama adalah recall atau ketika graph lebih lengkap tersedia. model_version

pada endpoint memungkinkan evaluasi dan rollback tanpa mengganti kontrak permintaan wallet.

Pencatatan pada risk registry menambah auditability, bukan akurasi. Record on-chain menunjukkan wallet, score, status, timestamp, dan transaction hash pada saat tertentu. Untuk interpretasi yang benar, record perlu dikaitkan dengan versi model dan threshold. Jika model diperbarui, skor baru ditambahkan sebagai record baru sehingga perubahan dapat ditelusuri dan skor lama tidak dianggap mewakili kondisi terkini.

4.11. Ancaman terhadap Validitas Penelitian

Ancaman validitas konstruk berasal dari definisi phishing dan fitur yang digunakan. Label Etherscan atau isp merepresentasikan wallet yang diketahui sebagai phishing, tetapi tidak menjamin seluruh kelas negatif benar-benar bersih. Risk score merupakan keluaran model berdasarkan pola transaksi, bukan ukuran kerugian atau niat pelaku secara langsung. Oleh karena itu, interpretasi dibatasi pada kemampuan klasifikasi terhadap label penelitian.

Ancaman validitas internal mencakup data leakage, duplikasi, perbedaan split, dan preprocessing yang menggunakan informasi test. Penelitian mengurangi risiko tersebut melalui normalisasi wallet, deduplikasi, split terstratifikasi yang sama, fitting transformasi pada training set, dan pemisahan held-out test. Namun, audit independen terhadap seluruh pipeline tetap merupakan pengembangan yang bernilai, khususnya pada konstruksi graph dan enrichment eksternal.

Ancaman validitas eksternal berkaitan dengan periode serta cakupan EPTransNet. Pola phishing, jenis transaksi, penggunaan token, dan ekosistem Ethereum berubah seiring waktu. Performa pada dataset historis tidak otomatis berlaku pada mainnet saat ini, layer-2, atau blockchain lain. Selain itu, selisih performa XGBoost dan GraphSAGE pada penelitian ini bersifat kondisional terhadap subgraph sampel yang sparse dan satu fixed split; topologi yang lebih lengkap, heterogeneous graph, atau periode data berbeda dapat mengubah peringkat relatif model. Evaluasi temporal, multi-seed/cross-validation, dan external validation pada data yang lebih baru diperlukan sebelum hasil digeneralisasi ke deployment operasional.

Ancaman conclusion validity muncul ketika selisih performa kecil ditafsirkan terlalu kuat berdasarkan satu split. Penggunaan wallet test yang sama meningkatkan keterbandingan, tetapi confidence interval dan repeated experiment dapat memberi estimasi stabilitas yang lebih lengkap. Hasil utama tetap bermakna sebagai bukti pada protokol yang ditentukan, sementara generalisasi angka presisi ke sampel lain harus dilakukan dengan hati-hati.

Ancaman implementasi mencakup perbedaan antara pipeline training dan serving. Urutan fitur, versi library, aturan missing, dan tipe data dapat berubah ketika model dipasang pada API. Penyimpanan schema, preprocessing artifact, checksum, serta model version mengurangi risiko training-serving skew. Uji integrasi dengan wallet contoh memastikan hasil endpoint sesuai dengan prediksi offline sebelum pencatatan blockchain dilakukan.

Keterbatasan tersebut tidak menghapus kontribusi penelitian. Sebaliknya, batas validitas menjelaskan ruang tempat kesimpulan didukung bukti: transformasi MultiDiGraph menjadi fitur tabular, pengayaan Etherscan, perbandingan empat model pada split sama, analisis feature importance, eksperimen late fusion, serta implementasi risk registry. Penelitian selanjutnya dapat memperluas periode, label, graph density, dan evaluasi online untuk menguji ketahanan kontribusi tersebut.

4.12. Perbandingan Temuan dengan Penelitian Terdahulu

Penelitian Chen et al. [4] menunjukkan bahwa pola neighborhood dan karakteristik transaksi mampu membedakan wallet phishing. Hasil feature importance penelitian ini mendukung temuan tersebut karena in_out_ratio, unique_senders, in_degree, sent_received_ratio, dan total_received berada pada kelompok fitur dominan. Persamaannya terletak pada pemanfaatan perilaku lokal, sedangkan penelitian ini menambahkan evaluasi lintas sumber dan integrasi sistem.

Studi network embedding dan subgraph learning [5], [15], [38] menekankan manfaat representasi graph. GraphSAGE pada penelitian ini mencapai recall tinggi tetapi tidak mengungguli XGBoost dalam F1. Perbedaan tersebut dapat dipengaruhi kepadatan graph sampel, konfigurasi edge, label, dan protokol evaluasi. Hasil ini menunjukkan bahwa GNN tidak selalu unggul hanya karena data asal berbentuk graph. Eth-PSD [16] menggunakan pendekatan Machine Learning berbasis fitur pada dataset phishing Ethereum. Penelitian ini sejalan dalam penggunaan model tabular, tetapi menempatkan transformasi MultiDiGraph serta pengayaan Etherscan sebagai objek eksperimen. Empat skenario memungkinkan kontribusi sumber fitur diukur, bukan hanya melaporkan performa satu dataset gabungan tanpa ablasi sumber.

Penelitian temporal graph [17], [18] memanfaatkan urutan transaksi secara lebih eksplisit. Fitur temporal penelitian ini masih berupa timestamp spread, sehingga belum menangkap dinamika lengkap seperti inter-arrival time atau evolving neighborhood. Walaupun demikian, pendekatan agregat lebih ringan dan mudah diinterpretasikan. Perbandingan tersebut memperjelas trade-off antara kedalaman representasi temporal dan kebutuhan implementasi sistem tabular.

Kontribusi risk registry membedakan penelitian ini dari studi yang berhenti pada evaluasi offline. Registry tidak meningkatkan F1, tetapi menyediakan jejak hasil prediksi yang immutable pada lingkungan pengujian. Pemisahan kontribusi model dan kontribusi sistem penting agar novelty tidak dinilai hanya dari angka akurasi, melainkan juga dari alur graph-to-tabular, enrichment, evaluasi, deployment, dan audit trail.

4.13. Kontribusi Praktis dan Implikasi Pengembangan Lanjutan

Kontribusi praktis pertama adalah pipeline yang mengubah MultiDiGraph menjadi tabel per wallet. Pipeline ini memungkinkan organisasi yang belum memiliki infrastruktur GNN tetap memanfaatkan informasi graph melalui model tabular. Data

dictionary dan aturan agregasi membuat fitur dapat diaudit, sedangkan XGBoost menyediakan inference yang relatif sederhana untuk layanan API.

Kontribusi kedua adalah bukti bahwa enrichment perlu dipilih berdasarkan komplementaritas. Etherscan meningkatkan seluruh model, sedangkan BigQuery tidak memberikan kenaikan serupa karena sebagian fiturnya redundan dengan graph. Temuan ini membantu pengembang memprioritaskan biaya API dan engineering pada sumber yang membawa informasi baru, bukan mengumpulkan sebanyak mungkin kolom.

Kontribusi ketiga adalah pemisahan kebutuhan deteksi dan pencatatan. Model menghasilkan probability dan label, sementara smart contract menyimpan ringkasan risiko. Arsitektur modular memungkinkan model diperbarui tanpa mengganti kontrak selama schema registry dipertahankan. Sebaliknya, jaringan blockchain dapat diganti dari localhost ke testnet tanpa melatih ulang model.

Untuk pengembangan lanjutan, dataset perlu diperbarui secara temporal dan diperkaya dengan token transfer, internal transaction, contract interaction, serta kategori entitas. Evaluasi temporal membantu mengukur drift, sedangkan external validation pada dataset independen menguji generalisasi. Penambahan fitur harus melalui ablasi agar kompleksitas baru memiliki bukti manfaat.

Dari sisi model, penelitian lanjutan dapat membandingkan calibrated XGBoost, temporal GNN, heterogeneous graph, serta ensemble tabular-graph. Namun, peningkatan model perlu mempertahankan kontrol data yang sama. Target bukan hanya F1 lebih tinggi, tetapi juga confidence yang terkalibrasi, false negative rendah, explanation stabil, latency wajar, dan perilaku yang dapat dimonitor setelah deployment.

UNIVERSITAS
NUSA MANDIRI