JISKA (Jurnal Informatika Sunan Kalijaga)
Vol. 11, No. 3, SEPTEMBER, 2026, Pp. 339 – 349
ISSN: 2527 – 5836 (print) | 2528 – 0074 (online)

Deteksi Anomali Jaringan Transaksi Token ERC20 di Blockchain Ethereum Menggunakan Graph Neural Network

Nazhif Alaudin Fahmi [(1)*], Andrey Ferriyan [(2)]
Departemen Informatika, Universitas Pembangunan Veteran Yogyakarta, Sleman, Indonesia
e-mail : nazhfahmi@gmail.com, andrey.f@upnyk.ac.id.
* Penulis korespondensi.
Artikel ini diajukan 22 Januari 2026, direvisi 11 Mei 2026, diterima 13 Mei 2026, dan dipublikasikan 25 September 2026.

Abstract

The rapid development of Decentralized Finance (DeFi) on the Ethereum blockchain has increased ERC20 token transaction volumes but also triggered a surge in anomalous activities such as Ice Phishing, NFT Scams, and Reward Scams. Conventional detection methods on tabular data often fail to capture complex relational patterns between accounts. This study proposes a Graph Neural Network (GNN) approach using the GraphSAGE architecture to detect anomalies in ERC20 transaction networks. Transaction datasets were acquired from Etherscan, processed into graphs with additional structural features (PageRank, degree), and imbalance was handled using class weighting. Based on testing using 33,848 transactions, the model yielded an average accuracy of 95.92% and a macro F1-score of 82.62%. These results prove that GraphSAGE is effective in learning feature representations and graph structures to detect various ERC20 fraud modes simultaneously.

Keywords: Anomaly Detection, Blockchain, ERC20, Graph Neural Network, GraphSAGE

Abstrak

Perkembangan pesat *Decentralized Finance* (DeFi) di *blockchain* Ethereum meningkatkan volume transaksi token ERC20, namun juga memicu lonjakan aktivitas anomali seperti *Ice Phishing*, *NFT Scam*, dan *Reward Scam*. Metode deteksi konvensional pada data tabular sering gagal menangkap pola relasi kompleks antar akun. Penelitian ini mengusulkan pendekatan *Graph Neural Network* (GNN) menggunakan arsitektur GraphSAGE untuk mendeteksi anomali pada jaringan transaksi ERC20. *Dataset* transaksi diambil dari Etherscan, diproses menjadi graf dengan penambahan fitur struktural (*PageRank*, *degree*), dan ditangani ketidakseimbangannya menggunakan *class weighting*. Berdasarkan pengujian menggunakan 33.848 transaksi, model menghasilkan rata-rata *accuracy* sebesar 95,92% dan *F1-score macro* sebesar 82,62%. Hasil ini membuktikan bahwa GraphSAGE efektif dalam mempelajari representasi fitur dan struktur graf untuk mendeteksi berbagai modus penipuan ERC20 secara simultan.

Kata Kunci: *Blockchain*, Deteksi Anomali, ERC20, *Graph Neural Network*, GraphSAGE

1. PENDAHULUAN

Perkembangan teknologi *blockchain* telah menghadirkan paradigma baru dalam sistem keuangan digital yang terdesentralisasi (J. Zhang & Luo, 2017). Ethereum, sebagai salah satu platform *blockchain* terbesar, tidak hanya berfungsi sebagai jaringan transaksi aset kripto, tetapi juga mendukung eksekusi kontrak pintar (*smart contract*) yang memungkinkan pengembangan aplikasi terdesentralisasi (*Decentralized Finance* atau DeFi) dan *Non-Fungible Token* (NFT) (Z. Wang et al., 2021). Dalam ekosistem ini, standar token ERC20 memegang peran fundamental karena menyediakan kerangka kerja baku untuk interoperabilitas antar token, memfasilitasi integrasi dengan dompet digital dan bursa kripto, serta menjadi tulang punggung bagi inovasi ekonomi digital (Dyson et al., 2020; Rahimian et al., 2019). Fleksibilitas ini memicu pertumbuhan volume transaksi yang eksponensial. Namun, di sisi lain, tingginya nilai ekonomi dan sifat pseudonim dari transaksi *blockchain* menjadikan ekosistem ERC20 target menarik bagi berbagai aktivitas kejahatan siber (Chen et al., 2024). Selain itu, sifat *immutability* (ketiadaan perubahan) pada teknologi ini membuat transaksi yang telah dikonfirmasi tidak dapat dibatalkan, sehingga

kerugian finansial akibat kesalahan transfer atau penipuan bersifat permanen dan tidak dapat dipulihkan oleh pihak otoritas sentral manapun (K. Wang et al., 2022).

Seiring dengan meluasnya adopsi token ERC20, modus penipuan dan anomali transaksi berkembang menjadi semakin canggih dan spesifik. Fenomena baru seperti *Ice Phishing*, *NFT Scam*, dan *Reward Scam* telah muncul sebagai ancaman serius yang dapat merugikan pengguna hingga miliaran dolar. Berbeda dengan serangan *phishing* tradisional yang mencuri kunci pribadi (*private key*), *Ice Phishing* mengeksploitasi mekanisme persetujuan token (*token approval*) dengan menipu pengguna untuk memberikan hak akses transfer aset kepada pelaku. Sementara itu, *NFT Scam* dan *Reward Scam* memanfaatkan tren investasi dan iming-iming hadiah palsu untuk memancing korban berinteraksi dengan kontrak berbahaya. Kompleksitas *modus operandi* ini sering kali sulit dideteksi oleh sistem keamanan konvensional yang hanya menganalisis data transaksi secara individual tanpa melihat pola relasi antarakun (Xiong et al., 2024).

Beberapa penelitian menunjukkan performa tinggi dalam deteksi anomali berbasis graf, namun masih memiliki keterbatasan pada cakupan deteksi. Han et al. (2024) mengembangkan MT²AD, model berbasis GNN untuk mendeteksi anomali pada jaringan transaksi Ethereum dengan mempertimbangkan informasi temporal, struktur transaksi, dan pola transaksi lintas token. P. Li et al. (2022) menggunakan PDGNN dan mencapai akurasi 96,35% serta F1-score 90,58% dalam deteksi *phishing* berbasis graf, tetapi pendekatan tersebut masih terbatas pada satu jenis serangan. Pendekatan berbasis GNN untuk deteksi phishing Ethereum juga telah dikembangkan melalui incremental graph learning, ego-graph augmentation, dan pemanfaatan informasi struktural jaringan (Huang et al., 2024; S. Li et al., 2021; Xiong et al., 2024). Chen et al. (2024) mengembangkan model DA-HGNN dengan performa sangat tinggi, yaitu F1-score 0,994 dan *recall* 0,995. Pendekatan ini memiliki kompleksitas komputasi yang besar dan belum dioptimalkan untuk skenario deteksi *real-time*. Sementara itu, Jeong & Yoon (2025) menerapkan GCN berbasis *Account Relevance* dan *BFS Tracking* untuk mendeteksi transaksi mencurigakan seperti Tornado Cash, tetapi belum diterapkan pada transaksi token ERC20 maupun pada skenario deteksi anomali pada beberapa kelas.

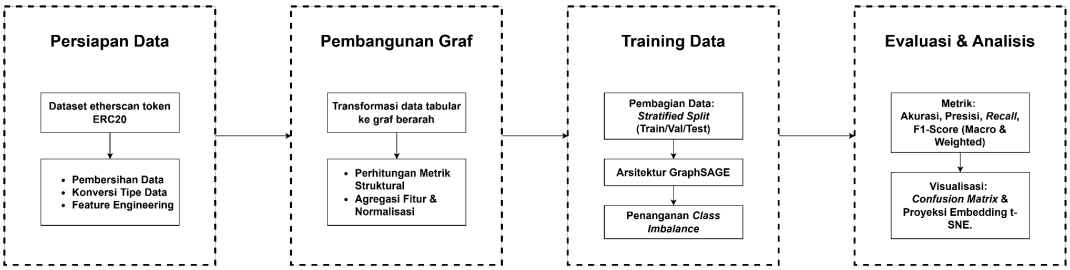
Untuk mengatasi kesenjangan tersebut, pendekatan berbasis *Graph Neural Network* menawarkan solusi yang menjanjikan. Transaksi *blockchain* secara alami membentuk struktur graf di mana alamat dompet berperan sebagai *node* dan transaksi sebagai *edge*. Representasi topologis tersebut memungkinkan karakteristik alamat dipelajari berdasarkan pola konektivitas dan hubungan transaksinya (Azad et al., 2025; Du et al., 2024; Seo et al., 2024; Y. Zhang & Liu, 2022). Model GNN mampu mempelajari representasi fitur tidak hanya dari atribut *node* itu sendiri, tetapi juga dari informasi tetangganya melalui mekanisme agregasi pesan (*message passing*).

Di antara berbagai arsitektur GNN, *GraphSAGE* (*Graph Sample and Aggregate*) memiliki keunggulan dalam pembelajaran induktif (*inductive learning*) yang memungkinkannya memprediksi label pada *node* baru yang belum pernah muncul selama proses pelatihan. Kemampuan ini mengatasi kelemahan model transduktif tradisional yang memerlukan pelatihan ulang secara menyeluruh setiap kali terdapat penambahan *node* baru, sehingga GraphSAGE jauh lebih efisien dari segi komputasi untuk skala jaringan Ethereum yang masif dan dinamis. Karakteristik ini sangat krusial untuk diterapkan pada jaringan *blockchain* yang dinamis dan terus berkembang.

Penelitian ini bertujuan untuk mengembangkan sistem deteksi anomali pada jaringan transaksi token ERC20 dengan memanfaatkan arsitektur *GraphSAGE*. Penelitian ini secara spesifik menargetkan deteksi terhadap tiga jenis anomali utama, yaitu *Ice Phishing*, *NFT Scam*, dan *Reward Scam*, yang belum banyak dibahas secara simultan dalam studi terdahulu. Kontribusi utama dari penelitian ini terletak pada pemanfaatan fitur struktural graf (seperti *PageRank*, *in-degree*, dan *out-degree*) yang dikombinasikan dengan *class weighting* untuk menangani ketidakseimbangan data yang ekstrem. Dengan pendekatan ini, diharapkan sistem mampu mengenali pola perilaku penipuan yang kompleks secara lebih akurat dan memberikan perlindungan yang lebih baik bagi ekosistem DeFi.

2. METODE PENELITIAN

Penelitian ini menggunakan metode kuantitatif dengan pendekatan pengembangan sistem prototipe yang dilakukan secara iteratif. Alur tahapan penelitian dirancang secara sistematis mulai dari studi literatur hingga evaluasi model. Alur tersebut diilustrasikan pada Gambar 1.



Gambar 1 Tahapan Penelitian

2.1 Persiapan Data

Tahap ini dimulai dengan akuisisi data transaksi token ERC20 dari Etherscan. *Dataset* mentah terdiri dari 15 atribut yang mencakup informasi teknis transaksi, identitas dompet, dan nilai ekonomi. Perincian atribut *dataset* disajikan pada Tabel 1.

Tabel 1 Deskripsi Atribut *Dataset* Transaksi ERC20

No.	Fitur	Deskripsi
1	Transaction Hash	Identitas unik heksadesimal setiap transaksi
2	Status	Status keberhasilan transaksi (Success/Failed)
3	Type	Jenis standar token (ERC-20)
4	Method	Fungsi *smart contract* yang dijalankan (misal: *Transfer*)
5	Block	Nomor blok tempat transaksi dicatat
6	Age	Waktu transaksi terjadi (*timestamp*)
7	From	Alamat dompet pengirim
8	From_NameTag	Label/tag publik pada alamat pengirim
9	To	Alamat dompet penerima
10	To_NameTag	Label/tag publik pada alamat penerima
11	Amount	Jumlah token yang ditransfer
12	Value (USD)	Nilai konversi transaksi dalam Dolar AS
13	Asset	Simbol token (misal: USDT, WETH)
14	Txn Fee	Biaya gas (*gas fee*) transaksi
15	Label	0: Normal, 1: *Ice Phishing*, 2: *NFT Scam*, 3: *Reward Scam*

Selanjutnya, dilakukan konversi tipe data untuk mengubah format mentah *blockchain* menjadi representasi numerik yang dapat diproses oleh model. Atribut *Value (USD)* dan *Amount* yang mengandung simbol mata uang dikonversi menjadi tipe data *float*, sedangkan atribut *Age* diubah menjadi format numerik *Unix timestamp* untuk memfasilitasi perhitungan interval waktu. Atribut kategori seperti *Method* dikodekan menggunakan teknik *Label Encoding*. Untuk meningkatkan kemampuan model dalam mendeteksi pola penipuan, dilakukan rekayasa fitur (*feature engineering*) dengan mengekstraksi tiga indikator utama. Pertama, fitur indikator *phishing* (is_from_phishing_tag) dibuat dengan memindai kata kunci berbahaya pada label alamat. Kedua, fitur deteksi metode *scam* (is_method_claim_airdrop) dibentuk untuk menandai fungsi kontrak yang mencurigakan seperti "Claim" atau "Airdrop". Ketiga, fitur anomali nilai (is_zero_value_usd.

2.2 Pembangunan Graf

Setelah data tabular siap, tahap selanjutnya adalah transformasi menjadi struktur graf berarah *G = (V, E)*. Dalam struktur ini, setiap alamat unik pada kolom *From* dan *To* direpresentasikan

sebagai *Node* (*V*), sedangkan setiap transaksi dianggap sebagai *Edge E* yang menghubungkan *node* pengirim ke penerima. Arah *edge* sangat penting karena merepresentasikan aliran aset atau pemanggilan fungsi kontrak.

Guna memperkaya informasi topologi jaringan, dilakukan perhitungan metrik struktural untuk setiap *node* yang meliputi *in-degree*, *out-degree*, dan *PageRank*. *PageRank* digunakan untuk mengukur tingkat sentralitas atau pengaruh *node* dalam jaringan global, di mana *node* yang menerima banyak koneksi dari *node* penting lainnya akan memiliki skor lebih tinggi (Lubis, 2023). Skor *PageRank* untuk *node* v ($PR(v)$) dihitung secara iteratif menggunakan Pers. (1). Di mana a adalah faktor redaman (*damping factor*) yang umumnya disetel ke 0.85, N adalah total jumlah *node*, $N(v)$ adalah himpunan *node* yang mengarah ke (v), dan D_{out} adalah jumlah tautan keluar dari *node* u.

$$PR(v) = (1 - a) + \alpha \sum_{u \in N(v)} \frac{PR(u)}{D_{out}} \quad (1)$$

Proses selanjutnya adalah agregasi fitur untuk membentuk vektor fitur *node* yang komprehensif. Statistik deskriptif (rata-rata, jumlah total, dan varians) dihitung dari atribut *Amount*, *Value (USD)*, dan *Gas Fee* untuk seluruh transaksi yang melibatkan *node* terkait. Langkah terakhir adalah normalisasi fitur menggunakan metode StandardScaler (Z-score normalization). Mengingat rentang nilai fitur yang sangat ekstrem, misalnya nilai *PageRank* yang sangat kecil dibandingkan dengan nilai transaksi dalam USD yang sangat besar, maka normalisasi diterapkan untuk menyamakan skala distribusi data. Perbedaan skala antarf fitur dapat memengaruhi proses pembelajaran atau analisis berbasis jarak apabila tidak ditangani dengan tepat (Wongoutong, 2024). Persamaan normalisasi untuk fitur x ditunjukkan pada Pers. (2), di mana z adalah nilai hasil normalisasi, μ adalah rata-rata (*mean*) dari fitur tersebut, dan σ adalah standar deviasi. Hal ini memastikan seluruh fitur memiliki rata-rata 0 dan variansi 1 untuk menjaga stabilitas gradien selama proses pelatihan model.

$$\mathrm{z} = \frac{\mathrm{x} - \mu}{\sigma} \quad (2)$$

2.3 Training Data

Sebelum proses pelatihan dimulai, *dataset* dibagi menggunakan teknik Stratified Split. Teknik ini digunakan untuk memastikan bahwa proporsi kelas anomali (minoritas) tersebar merata di setiap *subset* data. Alokasi pembagian data untuk pelatihan, validasi, dan pengujian disajikan pada Tabel 2.

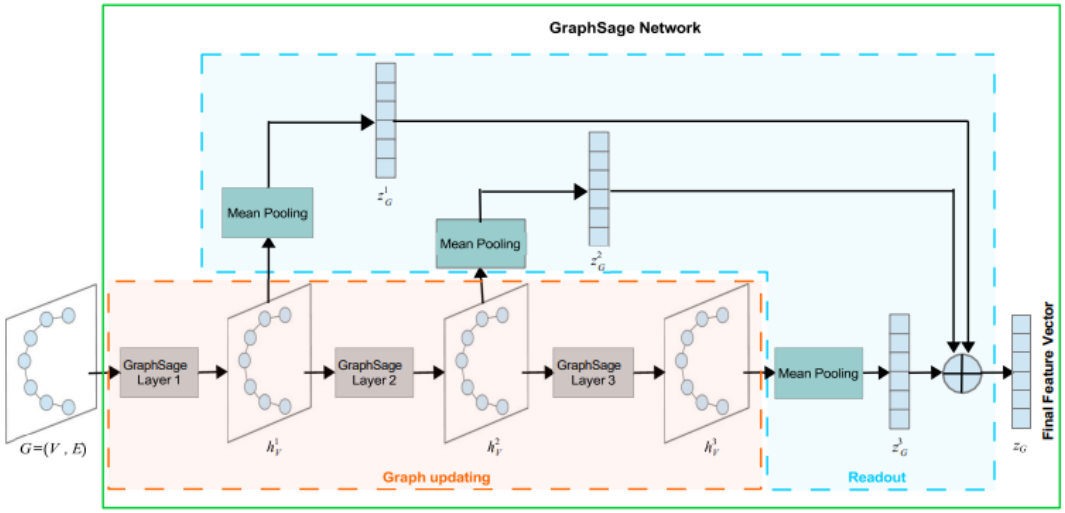
Tabel 2 Pembagian *Dataset*

Himpunan	Persentase	Keterangan
Training	59,5%	Digunakan untuk pelatihan model
Validation	10,5%	Digunakan untuk *early stopping* dan *tuning*
Test	30,0%	Digunakan untuk evaluasi akhir (tidak terlihat)

Pelatihan dilakukan menggunakan arsitektur GraphSAGE (*Graph Sample and Aggregate*). Arsitektur ini dirancang untuk melakukan pembelajaran induktif dengan mengambil sampel dan mengagregasi fitur dari lingkungan lokal *node*. Pendekatan GraphSAGE telah digunakan pada permasalahan klasifikasi berbasis graf, termasuk pada kondisi data yang tidak seimbang, dengan memanfaatkan informasi dari lingkungan node untuk membentuk representasi fitur (Seon et al., 2023). Skema arsitektur yang digunakan dalam penelitian ini diilustrasikan pada Gambar 2.

Seperti terlihat pada Gambar 2, GraphSAGE memperbarui representasi *node* dengan mengagregasi fitur dari tetangga lokalnya. Proses pembaruan vektor fitur pada lapisan ke- k mengikuti Pers. (3), di mana h_v^k adalah vektor hasil agregasi fitur tetangga, W_k adalah matriks

bobot yang dipelajari, dan σ adalah fungsi aktivasi ReLU. Selain itu, mengingat *dataset* memiliki ketidakseimbangan kelas yang ekstrem (kelas anomali < 11%), penelitian ini menerapkan strategi *class weighting* pada fungsi *loss* (*weighted cross-entropy*). Ketidakseimbangan kelas dapat menyebabkan model pembelajaran lebih berpihak pada kelas mayoritas dan menurunkan kemampuan identifikasi kelas minoritas (He & Garcia, 2009). Salah satu pendekatan untuk mengatasinya adalah pemberian bobot kelas yang lebih besar pada kelas minoritas (Bahtiar et al., 2025). Bobot untuk setiap kelas w_c dihitung secara proporsional terbalik terhadap frekuensinya agar model memberikan penalti yang lebih besar pada kesalahan prediksi kelas minoritas, seperti pada Pers. (4), di mana n_{total} adalah jumlah total sampel, $n_{classes}$ adalah jumlah kelas, dan n_c adalah jumlah sampel pada kelas c.



Gambar 2 Arsitektur GraphSAGE (Hemis et al., 2025)

$$h_v^k = \sigma\left(W_k \sum_{u \varepsilon N(v)} \frac{h_u^{k-1}}{N(v)} + B_k h_v^{k-1} \right) \tag{3}$$

$$w_c = \frac{n_{total}}{n_{classes} \times n_c} \tag{4}$$

2.4 Evaluasi & Analisis

Evaluasi kinerja model dilakukan secara komprehensif menggunakan pendekatan kuantitatif dan visual. Karena penelitian ini menangani masalah klasifikasi kelas dengan ketidakseimbangan data yang ekstrem, penggunaan akurasi saja tidak cukup untuk merepresentasikan keandalan model. Oleh karena itu, evaluasi didasarkan pada *confusion matrix* yang memetakan hasil prediksi ke dalam empat kategori, yaitu *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN). Berdasarkan komponen tersebut, dihitung metrik utama *precision*, *recall*, dan F1-score pada Pers. (5) sampai (7).

$$Precision = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{5}$$

$$Recall = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{6}$$

$$F1 = \frac{2 \times \mathrm{P} \times \mathrm{R}}{\mathrm{P} + \mathrm{R}} \tag{7}$$

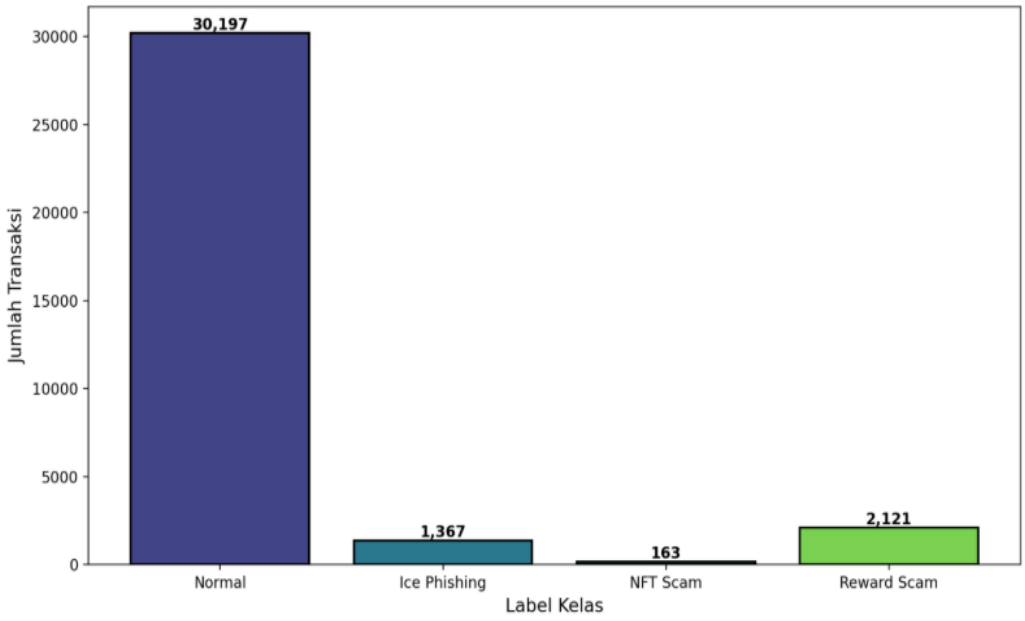
Selain metrik numerik, analisis kualitatif dilakukan menggunakan teknik reduksi dimensi t-Distributed Stochastic Neighbor Embedding (t-SNE). Metode t-SNE digunakan untuk memproyeksikan vektor fitur *node* (yang berdimensi tinggi, 128 dimensi dari *hidden layer* GraphSAGE) ke dalam ruang 2 dimensi. Visualisasi ini bertujuan untuk verifikasi apakah model berhasil mempelajari representasi fitur yang diskriminatif, yang ditandai dengan terbentuknya klaster-klaster terpisah antara transaksi normal dan berbagai jenis modus penipuan.

3. HASIL DAN PEMBAHASAN

Bagian ini memaparkan hasil eksperimen secara komprehensif. Analisis dimulai dari distribusi data, dinamika proses pelatihan model, dan evaluasi kinerja klasifikasi berdasarkan metrik numerik. Analisis juga dilakukan secara visual menggunakan t-SNE dan *confusion matrix* untuk memahami karakteristik kesalahan prediksi.

3.1 Analisis Distribusi *Dataset*

Setelah melalui tahapan pra-pemrosesan dan pembangunan graf, *dataset* final yang terbentuk terdiri dari 21.940 *node* dan 33.848 *edge*. Analisis awal terhadap label transaksi menunjukkan adanya ketidakseimbangan kelas (*class imbalance*) yang sangat ekstrem. Distribusi jumlah sampel untuk setiap kelas disajikan secara visual pada Gambar 3.



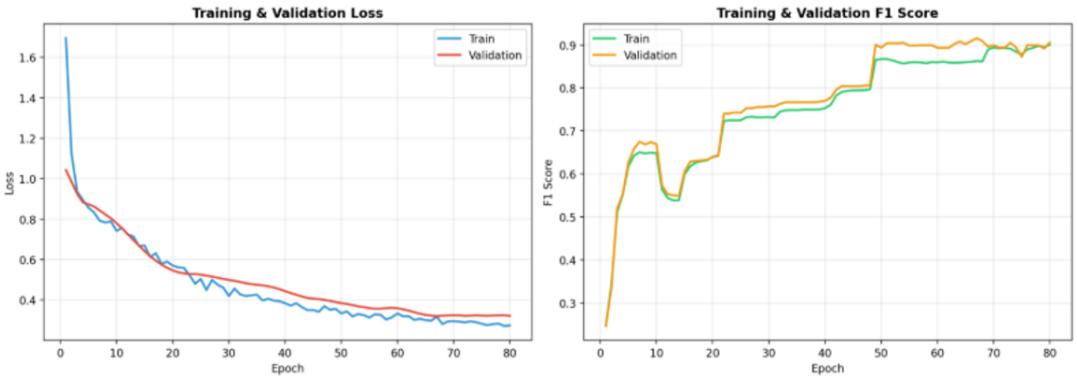
Gambar 3 Distribusi Data Split

Berdasarkan Gambar 3, terlihat bahwa kelas Normal mendominasi *dataset* dengan proporsi mencapai 89,22% (30.197 transaksi). Sementara itu, kelas anomali memiliki proporsi yang jauh lebih kecil, *Reward Scam* sebesar 6,27%, *Ice Phishing* sebesar 4,04%, dan *NFT Scam* hanya sebesar 0,48%. Ketimpangan ini menegaskan urgensi penerapan strategi *class weighting* yang telah dijelaskan pada metodologi, guna mencegah model menjadi bias dan hanya memprediksi kelas mayoritas (Normal).

3.2 Analisis Proses Pelatihan

Proses pelatihan model GraphSAGE dilakukan selama 100 *epoch* dengan memantau perubahan nilai *loss* dan F1-score pada data latih (*training*) dan data validasi (*validation*), sebagaimana diilustrasikan pada Gambar 4. Kurva *loss* menunjukkan penurunan yang tajam pada 20 *epoch* pertama dan mulai stabil mendekati *epoch* ke-80. Kurva validasi (garis merah) juga bergerak

seiring dengan kurva *training* (garis biru) dengan celah (*gap*) yang relatif sempit. Pada grafik F1-score, skor validasi terus meningkat dan kemudian stabil pada nilai yang tinggi. Pola tersebut mengindikasikan bahwa model memiliki kemampuan generalisasi yang baik dan tidak menunjukkan gejala *overfitting* yang signifikan selama proses pelatihan.



Gambar 4 Kurva Pelatihan

3.3 Evaluasi Kinerja Klasifikasi

Validasi model dilakukan menggunakan skema *train-validation-test split* dengan pendekatan *stratified sampling* untuk menjaga proporsi distribusi kelas pada setiap *subset* data. *Dataset* dibagi menjadi 70% untuk data latih dan validasi serta 30% untuk data uji, kemudian data latih dibagi kembali dengan rasio 85:15 untuk pelatihan dan validasi. Dengan demikian, komposisi akhir data adalah sekitar 59,5% untuk pelatihan, 10,5% untuk validasi, dan 30% untuk pengujian.

Proses pelatihan memanfaatkan mekanisme *early stopping* berdasarkan performa pada data validasi untuk mencegah *overfitting*. Untuk memastikan validitas hasil, pengujian dilakukan sebanyak lima kali (*multiple runs*) dengan inisialisasi bobot acak yang berbeda. Perincian performa untuk setiap kelas disajikan dalam Tabel 3.

Tabel 3 Hasil Evaluasi Tiap Label

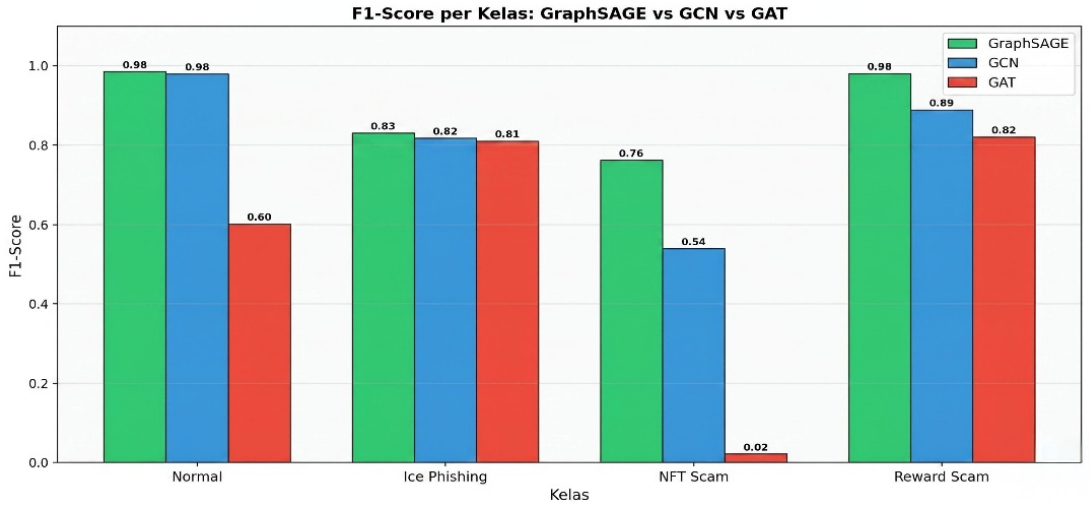
Label	TP	FP	FN	TN	Precision	Recall	F1-Score	Support
Normal	5678	59	24	821	0,9934	0,9785	0,9858	5802
Ice Phishing	269	1	3	6309	0,8152	0,8533	0,8322	272
NFT Scam	33	0	11	6538	0,6232	0,9730	0,7611	44
Reward Scam	422	2	40	6118	0,9936	0,9134	0,9805	462

Tabel 3 menunjukkan bahwa model efektif dalam mendeteksi anomali, dibuktikan dengan *recall yang* tinggi pada kelas *Ice Phishing* (98,53%) dan *NFT Scam* (97,3%). Meskipun terdapat sedikit penurunan pada *precision* untuk kelas NFT Scam (0,6232), yang berarti ada beberapa transaksi normal yang salah dideteksi sebagai scam (*False Positive*), hal ini merupakan trade-off yang dapat diterima demi meminimalkan *False Negative* (serangan yang lolos).

Rendahnya nilai *precision* pada kelas *NFT Scam* (0,6232) menunjukkan adanya kecenderungan model menghasilkan *false positive*, yaitu transaksi normal yang salah diklasifikasikan sebagai *scam*. Hal ini disebabkan oleh beberapa faktor. Pertama, jumlah sampel *NFT Scam* yang sangat kecil (hanya 0,48% dari total data) menyebabkan model kesulitan mempelajari representasi yang benar-benar spesifik. Kedua, karakteristik transaksi *NFT Scam* sering kali menyerupai transaksi normal, terutama pada nilai transaksi kecil atau interaksi awal dengan *smart contract*. Ketiga, penggunaan *class weighting* meningkatkan sensitivitas model terhadap kelas minoritas, namun juga meningkatkan kemungkinan *over-detection* (*false alarm*). Oleh karena itu, *trade-off* antara *recall* yang tinggi dan *precision* menjadi konsekuensi yang perlu dipertimbangkan.

3.4 Perbandingan dengan Metode Lain

Untuk memperkuat validasi performa model, dilakukan eksperimen perbandingan antara arsitektur *GraphSAGE* dengan dua arsitektur *Graph Neural Network* populer lainnya, yaitu GCN (*Graph Convolutional Network*) dan GAT (*Graph Attention Network*). Penggunaan GCN dalam deteksi transaksi Ethereum merujuk pada penelitian Jeong & Yoon (2025) yang menerapkan metode tersebut untuk melacak pola transaksi mencurigakan. Sementara itu, GAT sering digunakan sebagai pembanding dalam literatur deteksi *phishing* Ethereum karena kemampuannya dalam memberikan bobot perhatian pada *node* tetangga, sebagaimana dibahas dalam studi Xiong et al. (2024), seperti terlihat pada Gambar 5.



Gambar 5 Confusion Matrix pada Data Uji

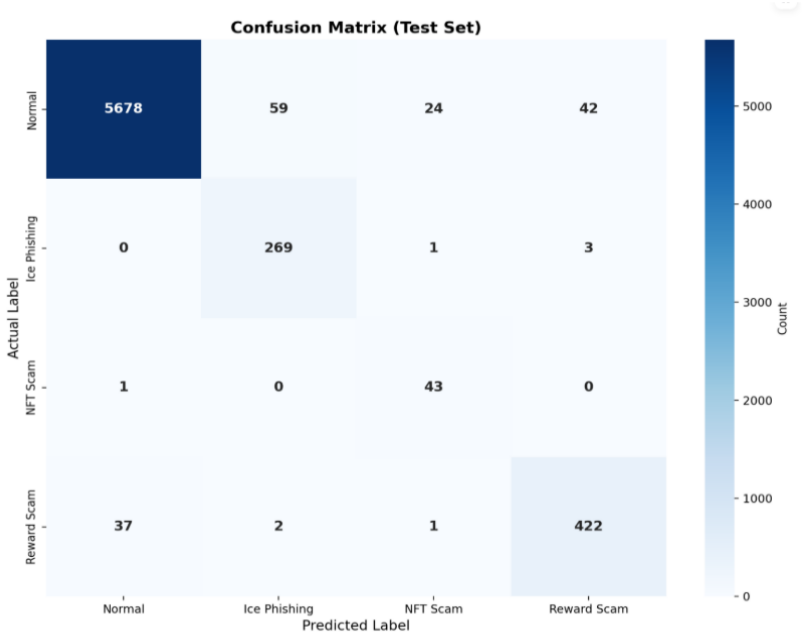
Berdasarkan visualisasi F1-Score per kelas pada Gambar 5, GraphSAGE menunjukkan performa paling stabil di seluruh kelas. Pada kelas Normal, GraphSAGE dan GCN memiliki performa setara (F1-Score 0.98), namun GAT mengalami penurunan signifikan (0.60). Pada kelas minoritas seperti NFT Scam, GraphSAGE unggul dengan F1-Score 0.76 dibandingkan GCN (0.54) dan GAT (0.02). Hal ini menunjukkan bahwa mekanisme sampling dan agregasi lokal pada GraphSAGE lebih efektif dalam menangkap pola relasi pada data yang tidak seimbang dibandingkan metode lain yang cenderung sensitif terhadap *noise* dan distribusi data ekstrem.

3.5 Analisis Kesalahan dan Visualisasi

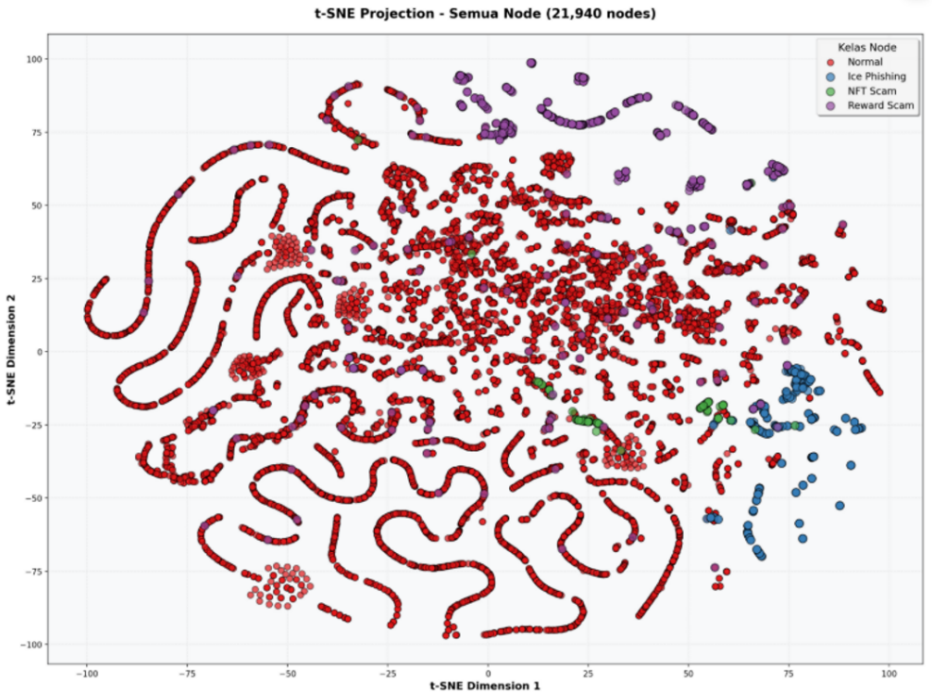
Untuk memahami di mana letak kesalahan model, dilakukan analisis menggunakan *confusion matrix* yang ditampilkan pada Gambar 6. Berdasarkan Gambar 6, mayoritas kesalahan prediksi terjadi antara kelas *Reward Scam* dan Normal. Hal ini wajar mengingat karakteristik *Reward Scam* sering kali melibatkan transfer token yang sah di awal (pancingan) sebelum penipuan terjadi, sehingga pola transaksinya memiliki kemiripan fitur yang tinggi dengan aktivitas normal. Namun, untuk serangan berbahaya seperti *Ice Phishing*, model sangat akurat dalam memisahkannya. Selanjutnya, untuk memverifikasi kemampuan GraphSAGE dalam membentuk representasi fitur (*embedding*), dilakukan proyeksi vektor fitur node dari dimensi tinggi (128 dimensi) ke 2 dimensi menggunakan algoritma t-SNE (*t-Distributed Stochastic Neighbor Embedding*), seperti terlihat pada Gambar 7.

Visualisasi t-SNE pada Gambar 7 memproyeksikan *embedding* dari 21.940 *node* ke dalam ruang dua dimensi untuk mengevaluasi daya pembeda model. Hasil proyeksi memperlihatkan bahwa kelas Normal (89,22%) mendominasi ruang fitur dengan sebaran luas yang mempertahankan struktur global graf. Sebaliknya, kelas anomali (*Ice Phishing, NFT Scam, Reward Scam*) membentuk klaster-klaster yang terpisah secara signifikan, selaras dengan capaian F1-Score

Macro sebesar 82,62%. Meskipun terdapat sedikit irisan (*overlap*) yang merefleksikan area ambiguitas sekitar 4,08% terutama akibat kemiripan fitur antara transaksi penipuan bernilai kecil dengan transaksi normal, pemisahan struktural yang tegas membuktikan bahwa GraphSAGE efektif menangkap pola diskriminatif utama di tengah ketidakseimbangan data yang ekstrem.



Gambar 6 Confusion Matrix pada Data Uji



Gambar 7 Visualisasi t-SNE

Proyeksi *embedding* mengindikasikan bahwa kedekatan posisi antar titik pada ruang fitur berkorelasi langsung dengan peningkatan ambiguitas klasifikasi, khususnya pada kelas dengan

karakteristik transaksi yang tidak memiliki pola ekstrem. Area yang saling beririsan menunjukkan bahwa sebagian sampel memiliki representasi fitur yang sangat mirip meskipun berasal dari label berbeda, sehingga model cenderung menghasilkan *false positive*. Pola ini sejalan dengan distribusi kesalahan yang terlihat pada *confusion matrix*, di mana kesalahan tidak terjadi secara acak melainkan terpusat pada kelas dengan kemiripan struktur transaksi. Hal ini menegaskan bahwa keterbatasan utama model terletak pada keterpisahan fitur di ruang laten, bukan pada kemampuan deteksi anomali secara umum, sehingga peningkatan kualitas fitur atau pendekatan representasi yang lebih kontekstual menjadi kunci untuk mengurangi ambiguitas antarkelas.

4. KESIMPULAN

Berdasarkan analisis dan pembahasan yang telah dilakukan, penelitian ini menyimpulkan bahwa pendekatan *Graph Neural Network* (GNN) dengan arsitektur GraphSAGE efektif dalam mendeteksi anomali pada jaringan transaksi token ERC20. Model yang diusulkan mampu mengidentifikasi pola serangan spesifik (*Ice Phishing*, *NFT Scam*, dan *Reward Scam*) di tengah dominasi transaksi normal, dengan menghasilkan rata-rata akurasi 95,92% dan F1-Score Macro 82,62%. Keberhasilan ini menjawab permasalahan deteksi pada *imbalanced dataset* melalui penerapan strategi *class weighting* serta pemanfaatan fitur struktural graf (seperti *PageRank* dan *Degree*) yang terbukti meningkatkan sensitivitas model terhadap anomali langka. Selain itu, visualisasi *t-SNE* mengonfirmasi bahwa model berhasil mempelajari representasi fitur yang diskriminatif, di mana *node* anomali membentuk klaster yang terpisah dari aktivitas legal, meskipun terdapat sedikit ambiguitas pada pola transaksi *NFT Scam*. Secara umum, kemampuan pembelajaran induktif dari GraphSAGE terbukti layak diterapkan pada karakteristik jaringan *blockchain* yang dinamis.

Namun demikian, penelitian ini memiliki beberapa keterbatasan. Keterbatasan tersebut meliputi distribusi data yang sangat tidak seimbang, terutama pada kelas NFT Scam, yang berpotensi memengaruhi stabilitas model, serta keterbatasan fitur yang belum sepenuhnya menangkap konteks semantik dari interaksi *smart contract*. Selain itu, evaluasi masih dilakukan pada skenario statis sehingga belum merepresentasikan kondisi *real-time* secara penuh.

Sebagai tindak lanjut dari penelitian ini, disarankan beberapa rekomendasi akademik dan praktis. Pertama, perluasan *dataset* anomali, khususnya pada kelas NFT Scam, diperlukan untuk memperkaya variasi pola serangan agar model lebih *robust*. Kedua, eksplorasi arsitektur GNN lain diperlukan untuk membandingkan efektivitas mekanisme model dalam menangani *node* yang memiliki kemiripan tinggi dengan transaksi normal. Ketiga, pengembangan sistem ke arah deteksi *real-time* dapat dilakukan melalui integrasi API *blockchain* agar klasifikasi dapat dilakukan secara langsung pada transaksi yang baru masuk.

DAFTAR PUSTAKA

Azad, P., Coskunuzer, B., Kantarcioglu, M., & Akcora, C. G. (2025). Chainlet Orbits: Topological Address Embedding for Blockchain. *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.1*, 25–36. https://doi.org/10.1145/3690624.3709322

Bahtiar, A., Hutomo, M. I. P., Widiyanto, A., & Khomsah, S. (2025). Class Weighting Approach for Handling Imbalanced Data on Forest Fire Classification Using EfficientNet-B1. *JISKA (Jurnal Informatika Sunan Kalijaga)*, *10*(1), 63–73. https://doi.org/10.14421/jiska.2025.10.1.63-73

Chen, Z., Liu, S.-Z., Huang, J., Xiu, Y.-H., Zhang, H., & Long, H.-X. (2024). Ethereum Phishing Scam Detection Based on Data Augmentation Method and Hybrid Graph Neural Network Model. *Sensors*, *24*(12), Article ID: 4022. https://doi.org/10.3390/s24124022

Du, H., Che, Z., Shen, M., Zhu, L., & Hu, J. (2024). Breaking the Anonymity of Ethereum Mixing Services Using Graph Feature Learning. *IEEE Transactions on Information Forensics and Security*, *19*, 616–631. https://doi.org/10.1109/TIFS.2023.3326984

Dyson, S. F., Buchanan, W. J., & Bell, L. (2020). Scenario-Based Creation and Digital Investigation of Ethereum ERC20 Tokens. *Forensic Science International: Digital Investigation*, *32*, Article ID: 200894. https://doi.org/10.1016/j.fsidi.2019.200894

He, H., & Garcia, E. A. (2009). Learning from Imbalanced Data. *IEEE Transactions on Knowledge and Data Engineering*, *21*(9), 1263–1284. https://doi.org/10.1109/TKDE.2008.239

Han, B., Wei, Y., Wang, Q., Collibus, F. M. De, & Tessone, C. J. (2024). MT^2AD: Multi-Layer Temporal Transaction Anomaly Detection in Ethereum Networks with GNN. *Complex & Intelligent Systems*, *10*(1), 613–626. https://doi.org/10.1007/s40747-023-01126-z

Hemis, M., Kheddar, H., Ghanem, M. C., & Boudraa, B. (2025). Hierarchical Graph Neural Network for Compressed Speech Steganalysis. *Array*, *28*, Article ID: 100510. https://doi.org/10.1016/j.array.2025.100510

Huang, H., Zhang, X., Wang, J., Gao, C., Li, X., Zhu, R., & Ma, Q. (2024). PEAE-GNN: Phishing Detection on Ethereum via Augmentation Ego-Graph Based on Graph Neural Network. *IEEE Transactions on Computational Social Systems*, *11*(3), 4326–4339. https://doi.org/10.1109/TCSS.2023.3349071

Jeong, J., & Yoon, J. W. (2025). GNN-based Detection of Malicious Transactions in Ethereum Networks: Leveraging Tornado Cash Transaction Patterns. *Proceedings of the 7th ACM International Symposium on Blockchain and Secure Critical Infrastructure*, 1–8. https://doi.org/10.1145/3709016.3737795

Li, P., Xie, Y., Xu, X., Zhou, J., & Xuan, Q. (2022). *Phishing Fraud Detection on Ethereum using Graph Neural Network*. http://arxiv.org/abs/2204.08194

Li, S., Xu, F., Wang, R., & Zhong, S. (2021). *Self-supervised Incremental Deep Graph Learning for Ethereum Phishing Scam Detection*. http://arxiv.org/abs/2106.10176

Lubis, A. (2023). Implementation of PageRank Algorithm for Visualization and Weighting of Keyword Networks in Scientific Papers. *Journal of Applied Data Sciences*, *4*(4), 382–391. https://doi.org/10.47738/jads.v4i4.138

Rahimian, R., Eskandari, S., & Clark, J. (2019). *Resolving the Multiple Withdrawal Attack on ERC20 Tokens*. http://arxiv.org/abs/1907.00903

Seo, M., Kim, J., You, M., Shin, S., & Kim, J. (2024). gShock: A GNN-Based Fingerprinting System for Permissioned Blockchain Networks Over Encrypted Channels. *IEEE Access*, *12*, 146328–146342. https://doi.org/10.1109/ACCESS.2024.3469583

Seon, J., Lee, S., Sun, Y. G., Kim, S. H., Kim, D. I., & Kim, J. Y. (2023). GraphSAGE with Contrastive Encoder for Efficient Fault Diagnosis in Industrial IoT Systems. *ICT Express*, *9*(6), 1226–1232. https://doi.org/10.1016/j.icte.2023.07.012

Wang, K., Wang, Q., & Boneh, D. (2022). *ERC-20R and ERC-721R: Reversible Transactions on Ethereum*. http://arxiv.org/abs/2208.00543

Wang, Z., Jin, H., Dai, W., Choo, K.-K. R., & Zou, D. (2021). Ethereum Smart Contract Security Research: Survey and Future Research Opportunities. *Frontiers of Computer Science*, *15*(2), Article ID: 152802. https://doi.org/10.1007/s11704-020-9284-9

Wongoutong, C. (2024). The Impact of Neglecting Feature Scaling in K-Means Clustering. *PLOS ONE*, *19*(12), Article ID: e0310839. https://doi.org/10.1371/journal.pone.0310839

Xiong, A., Tong, Y., Jiang, C., Guo, S., Shao, S., Huang, J., Wang, W., & Qi, B. (2024). Ethereum Phishing Detection Based on Graph Neural Networks. *IET Blockchain*, *4*(3), 226–234. https://doi.org/10.1049/blc2.12031

Zhang, J., & Luo, Y. (2017). Degree Centrality, Betweenness Centrality, and Closeness Centrality in Social Network. *2017 2nd International Conference on Modelling, Simulation and Applied Mathematics (MSAM2017)*, 300–303. https://doi.org/10.2991/msam-17.2017.68

Zhang, Y., & Liu, D. (2022). Toward Vulnerability Detection for Ethereum Smart Contracts Using Graph-Matching Network. *Future Internet*, *14*(11), Article ID: 326. https://doi.org/10.3390/fi14110326