HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

Trương Gia Hiển

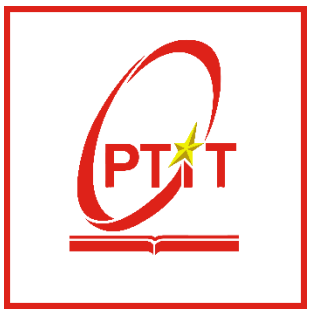
PHÁT HIỆN SỰ BẤT THƯỜNG TRONG HOẠT ĐỘNG CỦA MẠNG BLOCKCHAIN DỰA TRÊN HỌC MÁY

ĐỀ ÁN TỐT NGHIỆP THẠC SĨ KỸ THUẬT

(Theo định hướng ứng dụng)

HÀ NỘI - 2025

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG



Trương Gia Hiển

PHÁT HIỆN SỰ BẤT THƯỜNG TRONG HOẠT ĐỘNG CỦA MẠNG BLOCKCHAIN DỰA TRÊN HỌC MÁY

CHUYÊN NGÀNH: HỆ THỐNG THÔNG TIN

MÃ SỐ: 8.48.01.04

ĐỀ ÁN TỐT NGHIỆP THẠC SĨ KỸ THUẬT

(Theo định hướng ứng dụng)

NGƯỜI HƯỚNG DẪN KHOA HỌC: TS. Nguyễn Trung Kiên

HÀ NỘI - 2025

LỜI CAM ĐOAN

Tôi cam đoan đây là công trình nghiên cứu của riêng tôi.

Các số liệu, kết quả nêu trong đề án tốt nghiệp là trung thực và chưa từng được ai công bố trong bất kỳ công trình nào khác.

Tác giả đề án tốt nghiệp ký và ghi rõ họ tên

Trương Gia Hiển

MỤC LỤC

LỜI CAM ĐOAN i

MỤC LỤC ii

DANH MỤC CÁC KÝ HIỆU, CÁC CHỮ VIẾT TẮT v

DANH MỤC CÁC BẢNG vii

DANH MỤC CÁC HÌNH viii

MỞ ĐẦU ix

CHƯƠNG 1. CƠ SỞ LÝ THUYẾT 1

1.1. Giới thiệu chung 1

1.2. Tổng quan về Blockchain 1

1.2.1. Khái niệm của Blockchain 1

1.2.2. Đặc điểm của Blockchain 1

1.2.3. Cấu trúc dữ liệu của Blockchain 3

1.2.4. Giao dịch và vòng đời của giao dịch 4

1.3. Tổng quan về DeFi 5

1.3.1. Khái niệm của DeFi 5

1.3.2. Mục tiêu của DeFi 6

1.3.3. Ứng dụng phổ biến của DeFi 6

1.4. Hoạt động bất thường trong Blockchain 6

1.4.1. Khái niệm về hoạt động bất thường 6

1.4.2. Các hình thức gian lận giao dịch trong DeFi 7

1.5. Phương pháp phát hiện bất thường 8

1.5.1. Bài toán phát hiện bất thường trong dữ liệu giao dịch 8

1.5.2. Mô hình học máy 9

1.5.3. Mô hình học sâu 10

1.6. Các công trình liên quan 11

1.6.1. Tổng quan các công trình 11

1.6.2. Khoảng trống nghiên cứu 12

1.6.3. Đóng góp của luận văn 13

1.7. Kết luận chương 13

CHƯƠNG 2. PHƯƠNG PHÁP NGHIÊN CỨU VÀ THIẾT KẾ THỰC NGHIỆM 14

2.1. Giới thiệu chung 14

2.2. Dữ liệu 14

2.2.1. Bộ dữ liệu 14

2.2.2. Khai phá dữ liệu 14

2.2.3. Tiền xử lý dữ liệu 16

2.2.4. Kiểm soát chất lượng 19

2.3. Thiết kế mô hình huấn luyện 19

2.3.1. Pipeline huấn luyện chung 19

2.3.2. Thiết kế huấn luyện mô hình học máy 22

2.3.3. Thiết kế huấn luyện mô hình học sâu 25

2.4. Các thước đo đánh giá hiệu năng 29

2.4.1. Ma trận nhầm lẫn 30

2.4.2. Nhóm chỉ số đánh giá xếp hạng 30
2.4.3. Nhóm chỉ số đánh giá quyết định 31
2.4.4. Chiến lược tối ưu hóa tổng hợp 32
2.5. Kết luận chương 33
CHƯƠNG 3. TRIỂN KHAI THỰC NGHIỆM VÀ ĐÁNH GIÁ KẾT QUẢ 34
3.1. Giới thiệu chung 34
3.2. Môi trường và công cụ 34
3.2.1. Cấu hình phần cứng 34
3.2.2. Công cụ và thư viện phần mềm 35
3.3. Kết quả huấn luyện mô hình 36
3.3.1. Mô hình học máy 36
3.3.2. Mô hình học sâu 44
3.4. Đánh giá và so sánh kết quả mô hình 53
3.4.1. Chất lượng xếp hạng 54
3.4.2. Khả năng phân tách tổng quát 55
3.4.3. Hiệu quả tại vùng cảnh báo trọng điểm 56
3.4.4. Năng lực bao phủ và cân bằng phát hiện 57
3.4.5. Đánh giá hiệu năng tính toán và khả năng triển khai thực tế 59
3.5. Kết luận chương 61
KẾT LUẬN VÀ KIẾN NGHỊ 63
DANH MỤC TÀI LIỆU THAM KHẢO 65

DANH MỤC CÁC KÝ HIỆU, CÁC CHỮ VIẾT TẮT

Viết tắt	Tiếng Anh	Tiếng Việt
DeFi	Decentralized Finance	Tài chính phi tập trung
BFT	Byzantine Fault Tolerance	Khả năng chịu lỗi Byzantine
SPOF	Single Point of Failure	Điểm lỗi đơn
MEV	Maximal Extractable Value	Giá trị trích xuất tối đa
ABI	Application Binary Interface	Giao diện nhị phân ứng dụng
EVM	Ethereum Virtual Machine	Máy ảo Ethereum
AMM DEX	Automated Market Maker Decentralized Exchange	Sàn giao dịch phi tập trung dùng bộ tạo lập thị trường tự động
PCA	Principal Component Analysis	Phân tích thành phần chính
ELBO	Evidence Lower Bound	Cận dưới của log–likelihood (cận dưới bằng chứng)
CSV	Comma-Separated Values	Định dạng giá trị phân tách bằng dấu phẩy
BCCC	Blockchain, Cryptocurrency and Cybersecurity Center	Trung tâm Blockchain, Tiền mã hóa và An ninh mạng
NPZ	NumPy Zipped Archive	Định dạng tệp nén của NumPy (.npz)
TPE	Tree-structured Parzen Estimator	Bộ ước lượng Parzen có cấu trúc cây
AP	Average Precision	Độ chính xác trung bình
ROC-AUC	Area Under the Receiver Operating Characteristic Curve	Diện tích dưới đường cong ROC

PR-AUC	Area Under the Precision–Recall Curve	Diện tích dưới đường cong Precision–Recall
MSE	Mean Squared Error	Lỗi bình phương trung bình
MAD	Median Absolute Deviation	Độ lệch tuyệt đối theo trung vị
RNN	Recurrent Neural Network	Mạng nơ-ron hồi quy
TP	True Positive	Dương tính thật
FP	False Positive	Dương tính giả
TN	True Negative	Âm tính thật
FN	False Negative	Âm tính giả
TPR	True Positive Rate	Tỷ lệ dương tính thật (độ nhạy, Recall)
FPR	False Positive Rate	Tỷ lệ dương tính giả
I/O	Input/Output	Vào/ra (nhập/xuất)
VRAM	Video Random Access Memory	Bộ nhớ đồ họa (VRAM)
XLA	Accelerated Linear Algebra	Thư viện đại số tuyến tính tăng tốc XLA
GPU	Graphics Processing Unit	Bộ xử lý đồ họa
CPU	Central Processing Unit	Bộ xử lý trung tâm
RAM	Random Access Memory	Bộ nhớ truy cập ngẫu nhiên
GNNs	Graph Neural Networks	Mạng nơ-ron đồ thị
BAD	Blockchain Anomaly Detection	Phát hiện bất thường về chuỗi khối

DANH MỤC CÁC BẢNG

Bảng 2. 1: Các bản ghi dữ liệu minh họa do BCCC công bố **15**
Bảng 3. 1: Cấu hình phần cứng **35**
Bảng 3. 2: Cấu hình siêu tham số của K-Means **36**
Bảng 3. 3: Cấu hình siêu tham số của Isolation Forest **39**
Bảng 3. 4: Cấu hình siêu tham số của HBOS **42**
Bảng 3. 5: Cấu hình siêu tham số của DAE **44**
Bảng 3. 6: Cấu hình siêu tham số của LSTM-AE **47**
Bảng 3. 7: Cấu hình siêu tham số của Transformer-VAE **50**
Bảng 3. 8: Tổng hợp hiệu năng chi tiết của các mô hình **53**
Bảng 3. 9: Tổng hợp hiệu năng tính toán của các mô hình **59**

DANH MỤC CÁC HÌNH

Hình 2. 1: Phân bố dữ liệu **15**
Hình 2. 2: Top 30 đặc trưng quan trọng nhất **17**
Hình 3. 1: PR Curve và ROC Curve của K-Means **37**
Hình 3. 2: Phân phối điểm bất thường và ma trận nhầm lẫn của K-Means **38**
Hình 3. 3: PR Curve và ROC Curve của Isolation Forest **40**
Hình 3. 4: Phân phối điểm bất thường và ma trận nhầm lẫn của Isolation Forest .. **41**
Hình 3. 5: PR Curve và ROC Curve của HBOS **42**
Hình 3. 6: Phân phối điểm bất thường và ma trận nhầm lẫn của HBOS **43**
Hình 3. 7: PR Curve và ROC Curve của DAE **45**
Hình 3. 8: Phân phối điểm bất thường và ma trận nhầm lẫn của DAE **46**
Hình 3. 9: PR Curve và ROC Curve của LSTM-AE **48**
Hình 3. 10: Phân phối điểm bất thường và ma trận nhầm lẫn của LSTM-AE **49**
Hình 3. 11: PR Curve và ROC Curve của Transformer-VAE **51**
Hình 3. 12: Phân phối điểm bất thường và ma trận nhầm lẫn của Transformer-VAE **52**
Hình 3. 13: So sánh đường cong Precision–Recall giữa các mô hình **54**
Hình 3. 14: So sánh đường cong ROC giữa các mô hình **55**
Hình 3. 15: So sánh Precision@1% giữa các mô hình **56**
Hình 3. 16: So sánh Recall và F1-score tại ngưỡng tối ưu **57**

MỞ ĐẦU

Trong những năm gần đây, sự phát triển nhanh chóng của tài chính phi tập trung (Decentralized Finance – DeFi) trên nền tảng Blockchain, đặc biệt là Ethereum, đã hình thành một lớp hạ tầng tài chính mới với các đặc trưng không cần cấp phép, không lưu ký và có khả năng lập trình cao. Người dùng có thể trực tiếp tham gia hoán đổi tài sản, vay, cho vay, cung cấp thanh khoản, giao dịch phái sinh hay sử dụng cầu nối xuyên chuỗi chỉ với một ví và kết nối Internet. Tuy nhiên, chính tính mở và phi tập trung này cũng khiến DeFi trở thành mục tiêu hấp dẫn của nhiều hoạt động gian lận và tấn công có chủ đích, gây thiệt hại lớn về tài sản và ảnh hưởng tới niềm tin của người dùng.

Các hành vi gian lận trong DeFi xuất hiện dưới nhiều hình thức như vay chớp nhoáng thao túng nguồn giá, khai thác lỗ hổng hợp đồng thông minh, rút thanh khoản và thoát dự án (rug pull), tấn công cầu nối hay lợi dụng cơ chế quản trị. Điểm chung là mọi thao tác đều để lại dấu vết trên chuỗi khối dưới dạng giao dịch, sự kiện và trạng thái hợp đồng, nhưng phân phối dữ liệu lại rất lệch, nhiễu cao, phụ thuộc thời gian và thường thiếu nhãn đầy đủ. Trong bối cảnh đó, các phương pháp giám sát truyền thống dựa trên luật tĩnh hoặc các mô hình đơn giản khó theo kịp tốc độ biến đổi và mức độ tinh vi của các chiến thuật tấn công mới.

Việc nghiên cứu và ứng dụng các phương pháp học máy và học sâu cho bài toán phát hiện bất thường trên dữ liệu giao dịch DeFi vì vậy có ý nghĩa quan trọng cả về mặt khoa học lẫn thực tiễn. Về mặt khoa học, đây là bài toán điển hình của phát hiện bất thường trên dữ liệu tài chính phân tán, mất cân bằng và có trật tự thời gian, đòi hỏi một quy trình xử lý và đánh giá chặt chẽ. Về mặt ứng dụng, một hệ thống có khả năng phát hiện sớm các giao dịch hoặc chuỗi giao dịch bất thường có thể hỗ trợ sàn giao dịch, giao thức DeFi và các đơn vị quản lý rủi ro giảm thiểu thiệt hại, nâng cao an toàn cho người dùng và tăng độ tin cậy của toàn bộ hệ sinh thái.

Xuất phát từ những lý do trên, luận văn lựa chọn đề tài "Phát hiện sự bất thường trong hoạt động của mạng Blockchain dựa trên học máy", tập trung vào bối cảnh giao dịch DeFi trên Ethereum với dữ liệu chuẩn BCCC-DeFiFraudTrans-2025. Thay vì trình diễn một mô hình đơn lẻ, luận văn hướng tới xây dựng một chuẩn so sánh thực nghiệm nhất quán giữa hai nhóm tiếp cận: các mô hình học máy trên dữ liệu bảng đặc trưng, và các mô hình học sâu trên dữ liệu chuỗi thời gian, tất cả cùng vận hành trên một pipeline dữ liệu thống nhất.

Mục tiêu tổng quát của luận văn là khảo sát và đánh giá hiệu quả của một số mô hình tiêu biểu trong phát hiện bất thường/gian lận giao dịch DeFi dưới điều kiện dữ liệu thực tế, từ đó đề xuất cách tiếp cận phù hợp để hỗ trợ các hệ thống giám sát. Các mục tiêu cụ thể gồm:

Xây dựng cơ sở lý thuyết về Blockchain, DeFi, khái niệm hoạt động bất thường và gian lận giao dịch, cùng các hướng tiếp cận phát hiện bất thường dựa trên học máy và học sâu.

Thiết kế quy trình tiền xử lý dữ liệu cho hai dạng đầu vào (bảng đặc trưng và chuỗi thời gian), bảo đảm hạn chế rò rỉ thông tin, xử lý phân phối nặng đuôi và giữ được cấu trúc thời gian.

Triển khai, tối ưu và so sánh một nhóm mô hình học máy và một nhóm mô hình học sâu đại diện trên bộ dữ liệu BCCC-DeFiFraudTrans-2025 trong một khung thực nghiệm thống nhất.

Xây dựng và áp dụng hệ thước đo đánh giá phù hợp với dữ liệu mất cân bằng, tập trung vào chất lượng xếp hạng và hiệu quả ra quyết định (như PR-AUC, ROC-AUC, Precision@1%, F1 và Recall).

Phân tích ưu điểm, hạn chế của từng nhóm mô hình trong bối cảnh DeFi và thảo luận khả năng kết hợp phân tầng giữa mô hình tabular và mô hình chuỗi để nâng cao hiệu quả phát hiện.

Đối tượng nghiên cứu của luận văn là bài toán phát hiện bất thường/gian lận trong giao dịch DeFi trên Blockchain Ethereum dựa trên dữ liệu giao dịch đã được gán nhãn hợp pháp/gian lận. Phạm vi nghiên cứu tập trung vào bộ dữ liệu BCCC-DeFiFraudTrans-2025, với hai dạng biểu diễn chính: bảng đặc trưng ở mức giao dịch và chuỗi giao dịch theo địa chỉ với cửa sổ thời gian cố định. Luận văn chủ yếu xem xét các mô hình không giám sát hoặc bán giám sát dựa trên việc học hành vi bình thường và phát hiện sai lệch, chưa đi sâu vào các kiến trúc đồ thị hoặc các mô hình có giám sát thuần túy.

Về phương pháp, luận văn kết hợp giữa: nghiên cứu lý thuyết để hệ thống hóa các khái niệm về Blockchain, DeFi, phát hiện bất thường và các mô hình tiêu biểu; phân tích dữ liệu thực nghiệm trên bộ BCCC-DeFiFraudTrans-2025, thiết kế hai tuyến tiền xử lý cho dữ liệu bảng và dữ liệu chuỗi; xây dựng pipeline huấn luyện thống nhất cho cả hai nhóm mô hình, sử dụng tối ưu hóa siêu tham số tự động và tách bạch rõ ràng giữa tập huấn luyện, kiểm định, kiểm tra; và phân tích, so sánh kết quả theo nhiều góc nhìn để rút ra nhận xét và kiến nghị.

Những đóng góp chính của luận văn có thể tóm tắt như sau:

Đề xuất một quy trình tiền xử lý và huấn luyện có thể tái lập cho bài toán phát hiện bất thường trong giao dịch DeFi, với hai tuyến dữ liệu riêng cho bảng đặc trưng và chuỗi thời gian, tuân thủ nghiêm ngặt nguyên tắc tách tập theo trật tự khối để hạn chế rò rỉ thông tin.

Thực hiện một so sánh thực nghiệm có hệ thống giữa các mô hình học máy và học sâu trên cùng một bộ dữ liệu DeFi và cùng bộ thước đo đánh giá, qua đó làm rõ đặc điểm vận hành của từng nhóm trong bối cảnh dữ liệu mất cân bằng.

Chỉ ra sự phân hóa vai trò giữa các mô hình trên dữ liệu bảng (phù hợp làm lớp lọc ưu tiên, tập trung vào xếp hạng rủi ro) và các mô hình trên dữ liệu chuỗi (phù hợp tăng độ bao phủ nhờ khai thác động học theo thời gian), từ đó gợi ý hướng tiếp cận phân tầng khi triển khai hệ thống phát hiện gian lận trong thực tế.

Về bố cục, luận văn được tổ chức thành ba chương:

Chương 1. Cơ sở lý thuyết: Trình bày tổng quan về Blockchain và DeFi, khái niệm hoạt động bất thường và các hình thức gian lận điển hình trong DeFi, phát biểu bài toán phát hiện bất thường và giới thiệu các hướng mô hình sử dụng, cùng tổng quan công trình liên quan và khoảng trống nghiên cứu.

Chương 2. Phương pháp nghiên cứu và thiết kế thực nghiệm: Giới thiệu bộ dữ liệu BCCC-DeFiFraudTrans-2025, mô tả chi tiết quy trình tiền xử lý cho dữ liệu bảng và chuỗi thời gian, thiết kế pipeline huấn luyện chung, cấu hình và chiến lược tối ưu cho từng mô hình, cùng hệ chỉ số đánh giá.

Chương 3. Triển khai thực nghiệm và đánh giá kết quả: Trình bày môi trường thực nghiệm, kết quả tối ưu và đánh giá chi tiết cho từng mô hình, so sánh hiệu năng giữa các nhóm tiếp cận và thảo luận ý nghĩa thực tiễn, làm cơ sở cho các đề xuất và kiến nghị ở phần kết luận.

Thông qua cấu trúc này, luận văn kỳ vọng cung cấp một cái nhìn mạch lạc và toàn diện về việc áp dụng các kỹ thuật học máy và học sâu cho bài toán phát hiện bất thường trong giao dịch DeFi, đồng thời đưa ra các gợi ý thiết thực cho việc thiết kế và triển khai hệ thống giám sát gian lận trên nền tảng Blockchain.

CHƯƠNG 1. CƠ SỞ LÝ THUYẾT

1.1. Giới thiệu chung

Chương 1 thiết lập cơ sở lý thuyết và khung khái niệm cho bài toán phát hiện bất thường trong giao dịch Blockchain. Trọng tâm gồm: chuẩn hóa thuật ngữ, xác lập bối cảnh dữ liệu và phạm vi nghiên cứu, trình bày nền tảng về Blockchain và DeFi ở mức cần thiết, làm rõ khái niệm hoạt động bất thường cùng các đặc trưng dữ liệu liên quan, phát biểu khung bài toán và nhóm phương pháp dựa trên học máy, học sâu, đồng thời điểm qua các công trình liên quan để định vị đóng góp. Mục tiêu là cố định hệ thuật ngữ, tạo nền cho thiết kế thực nghiệm và phân tích kết quả ở các chương tiếp theo.

1.2. Tổng quan về Blockchain

1.2.1. Khái niệm của Blockchain

Blockchain là sổ cái phân tán theo mô hình chỉ ghi tổ chức thành các khối (block) liên kết bằng con trỏ băm (hash pointer) theo trật tự thời gian [20], [22]. Mỗi khối chứa tập giao dịch và siêu dữ liệu; băm của khối hiện tại tham chiếu khối trước nên mọi thay đổi quá khứ làm sai lệch dây băm và bị cơ chế đồng thuận bác bỏ. Các bản sao sổ cái được duy trì trên mạng ngang hàng (P2P) và được thống nhất thứ tự khối cùng trạng thái thông qua cơ chế đồng thuận [11]. Tính bất biến đạt được khi khối đạt tính hoàn tất: mang tính xác suất trong cơ chế đồng thuận kiểu Nakamoto và gần quyết định với các giao thức chịu lỗi Byzantine (BFT). Mức độ minh bạch phụ thuộc mô hình: chuỗi công khai cho phép kiểm chứng toàn bộ lịch sử, còn chuỗi cấp phép áp dụng kiểm soát truy cập [27].

1.2.2. Đặc điểm của Blockchain

1.2.2.1. Phi tập trung

Trong chuỗi khối công khai, không tồn tại máy chủ trung tâm. Nhiều nút ngang hàng cùng duy trì và xác thực sổ cái theo quy tắc giao thức, loại bỏ điểm lỗi đơn (SPOF),

tăng khả năng chống kiểm duyệt và nâng độ sẵn sàng [20]. Ở tầng dữ liệu, giao dịch lan truyền qua bộ nhớ chờ giao dịch (mempool) theo nhiều tuyến, hình thành dòng sự kiện đa nguồn, bối cảnh thuận lợi để nhận diện các mẫu hành vi lệch chuẩn theo thời gian và theo địa chỉ [29].

1.2.2.2. Bất biến

Mỗi khối tham chiếu băm của khối trước, nên mọi sửa đổi quá khứ sẽ phá vỡ chuỗi băm về sau và bị cơ chế đồng thuận bác bỏ dưới giả định đa số trung thực [22]. Mức độ hoàn tất phụ thuộc giao thức; có thể cần số xác nhận nhất định hoặc đạt hoàn tất ở tầng giao thức. Tính bất biến hỗ trợ định danh sự kiện bằng số khối và băm giao dịch, giúp gán nhãn và trích xuất đặc trưng ổn định, giảm rủi ro trôi nhãn theo thời gian. Cần lưu ý khả năng tái tổ chức chuỗi trong cửa sổ ngắn, vì vậy tập huấn luyện nên bỏ qua một số khối gần nhất để tránh dữ liệu chưa ổn định [6].

1.2.2.3. Minh bạch

Đa số chuỗi công khai cho phép kiểm chứng toàn bộ lịch sử, đồng thời duy trì ẩn danh bút danh: địa chỉ không gắn định danh thực, nhưng luồng giá trị vẫn quan sát được. Từ đó có thể xây dựng đặc trưng hành vi như tần suất và kích thước giao dịch, gas đã dùng, tổng gas (phí) tích lũy, số thứ tự giao dịch, mức độ kết nối đối tác, cũng như mẫu tương tác hợp đồng thông minh. Các tín hiệu này hữu ích cho mô hình phát hiện bất thường trong bối cảnh DeFi [16], [19].

1.2.2.4. Đồng thuận

Đồng thuận xác lập thứ tự khối và trạng thái đúng của sổ cái. Với bằng chứng công việc, công suất tính toán quyết định quyền đề xuất khối; với bằng chứng cổ phần, cổ phần cùng quy tắc hoàn tất chi phối an toàn và tính hoàn tất. Động học đồng thuận ảnh hưởng trực tiếp tới bộ nhớ chờ giao dịch và chi phí giao dịch: khi tắc nghẽn, cạnh tranh chèn giao dịch làm giá gas và phí ưu tiên biến động theo thời gian [11]. Đây là

nguồn tín hiệu quan trọng cho xếp hạng bất thường, đặc biệt trong các pha biến động cục bộ hoặc khi xuất hiện khai thác giá trị bởi người sản xuất khối (MEV) [7].

1.2.3. Cấu trúc dữ liệu của Blockchain

Blockchain là chuỗi các khối nối tiếp bằng con trỏ băm; mỗi khối là một cấu trúc dữ liệu gồm tiêu đề khối (block header) để ràng buộc toàn vẹn và thân khối (block body) để chứa danh sách giao dịch cùng dữ liệu liên quan. Phần này sử dụng ngữ cảnh Ethereum vì phù hợp với dữ liệu DeFi của luận văn.

1.2.3.1. Tiêu đề khối

Tiêu đề khối tập trung các cam kết mật mã và tham số vận hành, cho phép xác minh độc lập nội dung khối và ràng buộc toàn vẹn chuỗi [32].

Liên kết chuỗi: băm của khối trước bảo đảm mọi chỉnh sửa quá khứ sẽ làm đổi băm và làm mất hiệu lực các khối phía sau.

Ba gốc băm chính của cây Merkle-Patricia trie:

- Gốc giao dịch (Transactions root): cam kết danh sách giao dịch của khối.
- Gốc biên nhận (Receipts root): cam kết tập biên nhận thực thi, gồm trạng thái kết quả, lượng gas tiêu thụ và nhật ký sự kiện.
- Gốc trạng thái (State root): cam kết trạng thái toàn cục sau khi áp dụng toàn bộ giao dịch.

Tham số vận hành: số khối, thời điểm tạo, hạn mức gas, lượng gas đã dùng, phí cơ sở theo EIP-1559; các trường này chi phối năng lực chứa giao dịch và động học phí.

Ý nghĩa đối với luận văn: các cam kết trong tiêu đề khối cung cấp định danh bền vững để gắn nhãn và trích chọn đặc trưng; đồng thời cần lưu ý cửa sổ tái tổ chức chuỗi ngắn hạn khi xây dựng tập dữ liệu huấn luyện và đánh giá [12].

1.2.3.2. Thân khối

Thân khối chứa danh sách giao dịch được sắp theo quyết định của thực thể tạo khối; thứ tự này chi phối trình tự thực thi và tiêu thụ tài nguyên. Mỗi giao dịch sau khi thực thi tạo một biên nhận phản ánh kết quả, lượng gas sử dụng và các bản ghi sự kiện để phục vụ truy vết. Tùy phiên bản giao thức, thân khối có thể kèm thêm dữ liệu mở rộng như danh sách rút phần thưởng hoặc các đối tượng liên quan đến cơ chế mở rộng dữ liệu, nhưng hạt nhân vẫn là chuỗi giao dịch và các biên nhận tương ứng, vốn cung cấp nền tảng cho việc trích chọn đặc trưng và gắn nhãn trong bài toán phát hiện bất thường [1].

1.2.4. Giao dịch và vòng đời của giao dịch

1.2.4.1. Khái niệm và cấu trúc giao dịch

Giao dịch là đơn vị thao tác cơ bản làm thay đổi trạng thái sổ cái blockchain, như chuyển tài sản, triển khai hoặc gọi hợp đồng thông minh. Mỗi giao dịch là một thông điệp được ký bằng khóa riêng của người gửi và phát tán trên mạng để được xác thực và ghi vào khối.

Trong Ethereum, giao dịch thường gồm các thành phần chính: địa chỉ gửi và nhận, giá trị chuyển (value), dữ liệu gọi hợp đồng (data), nonce, các tham số phí gas theo EIP-1559 (maxFeePerGas, maxPriorityFeePerGas, gas limit), cùng chainId và chữ ký số để xác thực giao dịch.

1.2.4.2. Vòng đời giao dịch

Tạo và ký: ví tạo giao dịch, ước lượng gas và ký bằng khóa riêng.

Phát tán: giao dịch được gửi lên mạng và vào mempool, nơi các nút kiểm tra chữ ký, số dư và nonce [13].

Lựa chọn: người xác thực chọn giao dịch từ mempool và sắp thứ tự ưu tiên dựa trên phí và điều kiện mạng [7].

Thực thi: giao dịch được thực thi trong EVM, cập nhật trạng thái; nếu lỗi hoặc hết gas thì vẫn mất phí đã tiêu thụ.

Xác nhận: khối chứa giao dịch được lan truyền và thêm vào blockchain, đạt độ tin cậy sau một số xác nhận.

1.2.4.3. Kết quả thực thi, biên nhận và sự kiện

Sau khi thực thi, giao dịch tạo biên nhận (receipt) ghi lại trạng thái thành công/thất bại, lượng gas đã dùng và các sự kiện (events) phát sinh. Các sự kiện là nguồn dữ liệu quan trọng để theo dõi hoạt động DeFi như chuyển token, hoán đổi, vay hoặc cung cấp thanh khoản. Ngoài ra còn có giao dịch nội bộ (internal transactions) phát sinh từ lời gọi giữa các hợp đồng, được ghi lại trong logs và traces để phục vụ phân tích [16], [21].

1.3. Tổng quan về DeFi

1.3.1. Khái niệm của DeFi

Tài chính phi tập trung (DeFi) là lớp dịch vụ tài chính triển khai trực tiếp trên chuỗi khối thông qua hợp đồng thông minh (smart contract) [20]. Thay vì phụ thuộc vào tổ chức trung gian, các quy tắc xử lý giá trị được mã hóa và thực thi quyết định trên máy trạng thái của chuỗi. Đặc trưng vận hành của DeFi gồm không cần cấp phép cho phép bất kỳ địa chỉ ví hợp lệ tương tác giao thức; không lưu ký bảo toàn quyền kiểm soát khóa riêng và tài sản; và khả năng lắp ghép giúp nhiều giao thức liên kết tạo thành chuỗi thao tác trong một hoặc vài khối.

Dữ liệu trong DeFi hình thành từ giao dịch ở lớp thực thi, biên nhận và sự kiện hợp đồng, cùng với trạng thái sau mỗi khối. Từ đó hình thành các bề mặt quan sát giàu tín hiệu cho mô hình: tần suất tương tác, luồng tài sản giữa địa chỉ, cấu trúc gọi hợp đồng lồng nhau, cùng các tham số thị trường như trượt giá hay mức phí. Trong phạm vi luận văn, DeFi được coi là nguồn dữ liệu có cấu trúc, xác thực được, phục vụ xây dựng đặc trưng và gắn nhãn hành vi [16].

1.3.2. Mục tiêu của DeFi

DeFi hướng tới bốn mục tiêu cốt lõi. Thứ nhất là mở rộng tiếp cận dịch vụ tài chính với rào cản gia nhập thấp trên hạ tầng công khai. Thứ hai là tối ưu hiệu quả vốn thông qua tự động hóa quy tắc khớp lệnh, cho vay có thế chấp, và luân chuyển thanh khoản giữa giao thức. Thứ ba là giảm nhu cầu tin cậy tập trung bằng cách thay thế quy trình vận hành thủ công bằng mã nguồn công khai, có thể kiểm chứng trên chuỗi [22]. Thứ tư là lập trình được các sản phẩm tài chính mới bằng cách kết hợp nhiều khối xây dựng.

1.3.3. Ứng dụng phổ biến của DeFi

Những nhóm ứng dụng dưới đây vừa chiếm tỷ trọng lớn trong hoạt động DeFi, vừa cung cấp nhiều tín hiệu rủi ro phục vụ phát hiện bất thường [16], [27]:

- Hoán đổi tài sản (AMM DEX).
- Vay, cho vay thế chấp.
- Cung cấp thanh khoản, khai thác lợi suất.
- Phái sinh phi tập trung.
- Cầu nối và tài sản tổng hợp.

Tổng hợp lại, các ứng dụng này tạo ra không gian đặc trưng giàu tín hiệu ở ba lớp: thống kê giao dịch theo thời gian, đồ thị luồng tài sản giữa địa chỉ, hợp đồng và ngữ nghĩa sự kiện hợp đồng; đây là nền tảng cho thiết kế đặc trưng và lựa chọn mô hình trong các chương tiếp theo.

1.4. Hoạt động bất thường trong Blockchain

1.4.1. Khái niệm về hoạt động bất thường

Hoạt động bất thường được hiểu là các hành vi, chuỗi hành vi hoặc cấu trúc dữ liệu lệch đáng kể khỏi mô hình vận hành thông thường của mạng lưới [3], [11]. Trong bối cảnh Blockchain và DeFi, bất thường có thể xuất hiện ở nhiều cấp độ: tại một giao

dịch riêng lẻ, tại một địa chỉ qua thời gian, tại một hợp đồng thông minh, ở cấp khối hoặc ở cấp một cơ chế giao thức như bể thanh khoản hay cầu nối xuyên chuỗi. Bất thường không đồng nghĩa với gian lận; đây là khái niệm bao trùm, trong đó gian lận là một tập con mang tính chủ ý nhằm chiếm đoạt lợi ích hoặc làm sai lệch trạng thái.

Về hình thái, có thể phân biệt bất thường điểm (một giao dịch hoặc một khối lệch chuẩn), bất thường theo ngữ cảnh (hành vi chỉ trở nên bất thường trong một điều kiện nhất định như giai đoạn tắc nghẽn mạng), bất thường tập thể (một chuỗi lời gọi hợp đồng phối hợp tạo hiệu ứng) và thay đổi chế độ (sự chuyển pha hành vi kéo dài trong một khoảng thời gian) [3]. Các đặc thù của dữ liệu chuỗi khối làm cho bài toán trở nên khó: phân phối nặng đuôi, nhãn khan hiếm và cực mất cân bằng, trôi khái niệm theo thời gian, tác nhân đối kháng có khả năng thích nghi, dữ liệu đa nguồn và dạng dòng thời gian gần thời gian thực [12], [23]. Ngoài ra còn có các yếu tố hạ tầng như hàng đợi giao dịch, tái tổ chức chuỗi trong cửa sổ ngắn, và các cơ chế tối ưu thứ tự khớp lệnh có thể làm méo tín hiệu.

Để nhận diện hiệu quả, cần kết hợp các nhóm đặc trưng bổ trợ nhau: đặc trưng thống kê giao dịch theo thời gian, đặc trưng đồ thị dòng giá trị và quan hệ địa chỉ, đặc trưng thực thi hợp đồng từ nhật ký sự kiện, cùng trạng thái kinh tế của giao thức như thanh khoản, thế chấp và biến động chi phí giao dịch [6]. Phương pháp tiếp cận phù hợp thường là học không giám sát hoặc bán giám sát, ưu tiên xếp hạng nguy cơ để phục vụ rà soát theo ngưỡng hoặc theo tỉ lệ ưu tiên cố định.

1.4.2. Các hình thức gian lận giao dịch trong DeFi

Gian lận trong DeFi là các kịch bản có chủ đích nhằm trục lợi từ cơ chế hợp đồng thông minh, nguồn giá, thanh khoản hoặc quản trị. Một số nhóm điển hình gồm [19], [29]:

- Vay chớp nhoáng thao túng giá.

- Khai thác lỗi hợp đồng thông minh.
- MEV, đón đầu và kẹp giá.
- Rút thanh khoản và thoát dự án.
- Tấn công cầu nối xuyên chuỗi.
- Tấn công quản trị dùng vay chớp nhoáng.
- Lừa đảo cấp quyền chi tiêu, chiếm đoạt ví.
- Thao túng thị trường và tài sản thế chấp.

Tuy cơ chế khác nhau, các hình thức trên đều để lại những mẫu hình chung như: cường độ giao dịch tăng đột biến trong cửa sổ ngắn, chuỗi gọi hợp đồng sâu bất thường, trượt giá và chi phí gas thay đổi mạnh, dòng tài sản di chuyển nhanh qua nhiều ví, sự kiện rút thanh khoản lớn hoặc chuyển quyền quản trị gần thời điểm biến động giá, và quyền biểu quyết tăng đột ngột [29]. Các mẫu hình này là cơ sở để thiết kế đặc trưng ở cấp giao dịch, địa chỉ và giao thức trong mô hình phát hiện bất thường.

1.5. Phương pháp phát hiện bất thường

1.5.1. Bài toán phát hiện bất thường trong dữ liệu giao dịch

Mục tiêu của bài toán là gán cho mỗi thực thể quan sát một điểm rủi ro để xếp hạng ưu tiên hoặc ra quyết định theo một ngưỡng đã hiệu chỉnh [3]. Thực thể có thể là giao dịch đơn lẻ, một chuỗi giao dịch theo thời gian của một địa chỉ, một phiên tương tác với hợp đồng, hoặc một cụm sự kiện gắn với một giao thức DeFi. Đầu ra mong muốn là thang điểm liên tục phản ánh mức độ bất thường; việc phân loại nhị phân được suy ra bằng cách chọn ngưỡng phù hợp với chi phí sai sót và năng lực xử lý hậu kiểm.

Dữ liệu đầu vào thường có hai dạng: bảng đặc trưng ở mức giao dịch và chuỗi thời gian ở mức địa chỉ hoặc phiên tương tác. Nhãn gian lận khan hiếm hoặc không sẵn có, do đó quy trình thực nghiệm ưu tiên học biểu diễn hành vi bình thường và phát hiện

sai lệch so với chuẩn này, kết hợp hiệu chỉnh ngưỡng bằng một phần nhỏ nhãn nếu có [6].

Đặc thù của dữ liệu trên chuỗi đặt ra các ràng buộc kỹ thuật:

- **Mất cân bằng lớp:** tỉ lệ dương tính rất nhỏ, cần các độ đo dựa trên xếp hạng thay vì độ chính xác tổng quát.
- **Trôi khái niệm theo thời gian:** hành vi và chiến thuật tấn công thay đổi, yêu cầu tách tập theo thời gian, tránh rò rỉ tương lai và cho phép tái hiệu chỉnh định kỳ.
- **Phụ thuộc theo chuỗi:** nhiều bất thường hình thành qua một dãy hành động, cần mô hình hóa theo trình tự hoặc gộp theo cửa sổ.
- **Nhiễu và cực trị:** phân phối nặng đuôi do biến động phí và thanh khoản, đòi hỏi chuẩn hóa và quy tắc cắt ngưỡng ổn định.
- **Đối kháng:** kẻ tấn công có thể che giấu tín hiệu đơn lẻ, nên ưu tiên điểm số bền vững dựa trên nhiều đặc trưng và ngữ cảnh.

Theo chế độ nhãn, có ba hướng chính: không giám sát (học trên dữ liệu được giả định chủ yếu là bình thường), bán giám sát (dùng ít nhãn để hiệu chỉnh ngưỡng và ưu tiên), và có giám sát (tạo nhiệm vụ phụ như che, khôi phục để học biểu diễn bền vững) [6], [12].

1.5.2. Mô hình học máy

1.5.2.1. K-Means

Được lựa chọn làm phương pháp cơ sở (baseline) nhờ khả năng phân cụm hiệu quả trên các tập dữ liệu lớn không nhãn. Giả định rằng hành vi giao dịch bình thường sẽ tập trung thành các cụm đậm đặc, trong khi gian lận là các điểm nằm xa tâm cụm [18]. Mô hình này giúp nhận diện nhanh các giao dịch có đặc trưng thống kê lệch chuẩn rõ rệt.

1.5.2.2. Isolation Forest

Được lựa chọn nhờ khả năng xử lý tốt dữ liệu nhiều chiều và nhiễu mà không cần tính toán khoảng cách phức tạp. Thuật toán hoạt động dựa trên nguyên lý "cô lập": các giao dịch gian lận thường "ít và khác biệt", do đó dễ bị cô lập ở độ sâu thấp của cây quyết định hơn so với giao dịch bình thường [17]. Đây là phương pháp mạnh (robust) để xử lý các ngoại lai trong dữ liệu DeFi.

1.5.2.3. Histogram-Based Outlier Score

Được lựa chọn vì tốc độ tính toán cực nhanh và phù hợp với giả định các đặc trưng độc lập cục bộ. HBOS hiệu quả trong việc phát hiện các bất thường cục bộ trên các phân phối nặng đuôi (heavy-tailed) của dữ liệu tài chính mà các mô hình dựa trên khoảng cách (như K-Means) có thể bỏ sót [8].

1.5.3. Mô hình học sâu

1.5.3.1. Dense Autoencoder

Khắc phục hạn chế của các mô hình tuyến tính bằng cách học các biểu diễn phi tuyến của dữ liệu thông qua mạng nơ-ron nhiều lớp. DAE phát hiện bất thường dựa trên lỗi tái tạo (Reconstruction Error): mô hình được huấn luyện để tái tạo tốt các giao dịch bình thường, do đó các giao dịch gian lận với cấu trúc lạ sẽ sinh ra lỗi tái tạo lớn [9].

1.5.3.2. Long Short-term Memory Autoencoder

Được lựa chọn để giải quyết bài toán chuỗi thời gian. Khác với DAE chỉ nhìn vào từng giao dịch, LSTM-AE có khả năng ghi nhớ trạng thái quá khứ thông qua các cổng nhớ, từ đó phát hiện các bất thường về mặt trình tự (ví dụ: một chuỗi lệnh nạp - rút - hoán đổi diễn ra với tốc độ bất thường) mà các mô hình tĩnh không thể nhìn thấy [21].

1.5.3.3. Transformer Variational Autoencoder

Là mô hình tiên tiến nhất được áp dụng trong luận văn. Cơ chế Self-Attention giúp mô hình nắm bắt các phụ thuộc dài hạn tốt hơn LSTM, đồng thời không gian tiềm ẩn biến phân (Variational Latent Space) giúp tăng tính bền vững trước nhiễu dữ liệu on-chain. Mô hình này được kỳ vọng phát hiện các mẫu tấn công phức tạp kéo dài qua nhiều khối [16].

1.6. Các công trình liên quan

1.6.1. Tổng quan các công trình

Nghiên cứu phát hiện bất thường trên Blockchain khởi đầu từ bối cảnh Bitcoin, sau đó mở rộng sang Ethereum và đặc biệt là DeFi với mức độ tương tác hợp đồng và động học thị trường phức tạp hơn. Một khảo sát gần đây tổng hợp hệ phương pháp không giám sát thường gặp trong bối cảnh thiếu nhãn, bao gồm các kỹ thuật phân cụm như K-means, các phương pháp dựa trên cây như Isolation Forest, các phương pháp dựa trên mật độ, và các tiếp cận một lớp (ví dụ One-Class SVM) [6]. Khảo sát này đồng thời nêu bật các thách thức thực nghiệm đặc thù của dữ liệu trên chuỗi như mất cân bằng nghiêm trọng, trôi khái niệm theo thời gian, và nhu cầu đánh giá bằng các thước đo phù hợp xếp hạng ưu tiên. Những kết luận này định hình nền tảng phương pháp cho các nghiên cứu sau đó [5], [27].

Bên cạnh lớp phương pháp tổng quát, nhiều công trình đã kiểm chứng tính khả thi của học không giám sát trên dữ liệu giao dịch thực tế. Shafiq thực hiện một nghiên cứu so sánh trên dữ liệu Bitcoin, áp dụng K-means, Isolation Forest, Histogram-based Outlier Score và các biến thể autoencoder để nhận diện mẫu hành vi lệch chuẩn ở cấp giao dịch hoặc địa chỉ, qua đó khẳng định hiệu lực của các mô hình này trong môi trường thiếu nhãn [26]. Một hướng tiếp cận đồng thời khác đề xuất khung áp dụng K-means và Isolation Forest, nhấn mạnh vai trò của trích xuất đặc trưng từ dấu vết giao dịch như kích thước, dấu thời gian và địa chỉ, xem đây là đầu vào thiết yếu cho mô hình không giám sát [28].

Ngoài khai thác trực tiếp đặc trưng giao dịch, một số công trình tập trung vào siêu dữ liệu và kiến trúc mạng Blockchain. Giải pháp BAD (Blockchain Anomaly Detection) mô tả một hệ thống phân tán thu thập, đối chiếu và phân tích siêu dữ liệu trên chuỗi (bao gồm cả hiện tượng phân nhánh chuỗi), từ đó xây dựng kho tri thức phục vụ phát hiện bất thường theo thời gian thực [29]. Công trình nhấn mạnh khả năng tích hợp các kỹ thuật học máy và trí tuệ nhân tạo để tinh chỉnh hiệu quả nhận diện trong tương lai.

Khi DeFi trở thành bối cảnh trung tâm, yêu cầu mô hình hóa phụ thuộc dài hạn và tương tác hợp đồng tăng mạnh. Một nghiên cứu tiêu biểu đề xuất kết hợp bộ mã hóa tự động biến thiên với kiến trúc Transformer để vừa học biểu diễn tiềm ẩn, vừa nắm bắt phụ thuộc dài hạn trong chuỗi sự kiện của một giao thức DeFi cụ thể, qua đó xếp hạng bất thường gắn với cụm giao dịch hoặc phiên tương tác hợp đồng [16]. Kết quả cho thấy tiềm năng của các mô hình học sâu chuỗi trong nhận diện sự kiện bất thường nội sinh của DeFi.

1.6.2. Khoảng trống nghiên cứu

Tổng hợp các công trình cho thấy còn ba khoảng trống nổi bật. Thứ nhất, phần lớn so sánh mô hình học máy cổ điển tập trung vào Bitcoin [26], trong khi DeFi có cấu trúc tương tác đa hợp đồng, thanh khoản biến động và nhiều cơ chế sinh rủi ro (ví dụ vay chớp nhoáng, thao túng nguồn giá) chưa được khảo sát toàn diện bằng các mô hình không giám sát học máy cổ điển. Thứ hai, các nghiên cứu học sâu trong DeFi còn mới và thường trình diễn một kiến trúc đơn lẻ có độ phức tạp cao (như Transformer-VAE), thiếu các so sánh hệ thống giữa nhiều biến thể autoencoder để lượng hóa rõ ràng lợi ích theo chuỗi thời gian và theo ngữ cảnh sự kiện hợp đồng. Thứ ba, hiện chưa có một chuẩn so sánh đồng nhất đặt các mô hình học máy cổ điển cạnh các mô hình học sâu trên cùng một bộ dữ liệu DeFi với quy trình tách theo thời gian và bộ chỉ số đánh giá phù hợp tính mất cân bằng [12].

1.6.3. Đóng góp của luận văn

Luận văn này hướng tới lấp đầy các khoảng trống trên bằng một nghiên cứu thực nghiệm nhất quán và tái lập. Cụ thể, luận văn xây dựng một chuẩn so sánh trên dữ liệu DeFi, triển khai sáu mô hình gồm K-means, Isolation Forest, Histogram-based Outlier Score, Dense Autoencoder, LSTM-AE và Transformer-VAE trên cùng một pipeline tiền xử lý và đặc trưng, bảo đảm tách theo thời gian để phản ánh trôi khái niệm, và đánh giá bằng các thước đo phù hợp dữ liệu cực mất cân bằng. Thiết kế này vừa mở rộng hướng tổng quan phương pháp không giám sát được nêu trên, vừa đối chiếu có hệ thống hai họ mô hình (học máy và học sâu) trong bối cảnh DeFi vốn chưa được chuẩn hóa so sánh ở các công trình trước. Ngoài ra, luận văn nhấn mạnh mô hình học sâu theo chuỗi (LSTM và Transformer) như một đường hướng phù hợp để nắm bắt phụ thuộc theo thời gian và cấu trúc lời gọi hợp đồng, điểm đã được gợi mở nhưng chưa được so sánh rộng trong tài liệu hiện có.

1.7. Kết luận chương

Chương 1 đã đặt nền tảng lý thuyết cho đề tài: trình bày bản chất chuỗi khối, cấu trúc khối, cơ chế vận hành đặc thù của môi trường công khai, cũng như khái niệm và mục tiêu của tài chính phi tập trung. Trên cơ sở đó, chương đã làm rõ khái niệm hoạt động bất thường trong dữ liệu chuỗi khối, các hình thức gian lận điển hình trong hệ sinh thái DeFi và những tín hiệu có thể quan sát để nhận diện rủi ro. Phần phương pháp đã giới thiệu các hướng tiếp cận phát hiện bất thường theo cả học máy và học sâu, kèm nguyên lý vận hành cốt lõi của từng mô hình.

Từ các phân tích trên, luận văn xác định cách tiếp cận thực nghiệm cho các chương tiếp theo như sau: sử dụng hai góc nhìn dữ liệu bổ trợ nhau gồm bảng đặc trưng tabular và chuỗi thời gian giao dịch theo địa chỉ; áp dụng nhóm mô hình học máy không giám sát; áp dụng nhóm mô hình học sâu tái tạo để làm thực nghiệm.

CHƯƠNG 2. PHƯƠNG PHÁP NGHIÊN CỨU VÀ THIẾT KẾ THỰC NGHIỆM

2.1. Giới thiệu chung

Chương 2 chuyển hóa khung lý thuyết ở Chương 1 thành một quy trình thực nghiệm có thể tái lập nhằm phát hiện bất thường trong giao dịch DeFi. Trọng tâm là xác định rõ nguồn dữ liệu và cách biểu diễn phù hợp với mục tiêu phát hiện, chuẩn hóa các bước xử lý để bảo toàn tín hiệu hành vi, và xây dựng kiến trúc mô hình cùng chiến lược tối ưu bảo đảm tính khách quan. Các quyết định thiết kế đều bám sát đặc thù dữ liệu mất cân bằng, phụ thuộc thời gian và có trôi khái niệm, đồng thời ưu tiên khả năng so sánh công bằng giữa các phương pháp.

2.2. Dữ liệu

2.2.1. Bộ dữ liệu

Luận văn sử dụng bộ dữ liệu chuẩn BCCC-DeFiFraudTrans-2025 trên nền tảng Ethereum. Dữ liệu tập trung vào các đặc trưng hành vi ở cấp độ giao dịch (Gas, Nonce, dòng Token, trạng thái lỗi) thay vì lịch sử ví, cho phép phát hiện hiệu quả các kịch bản tấn công mới (zero-day). Nhãn gian lận được xác thực qua thẻ Etherscan và trải qua quy trình kiểm tra chéo nghiêm ngặt để đảm bảo tính nhất quán và loại bỏ trùng lặp.

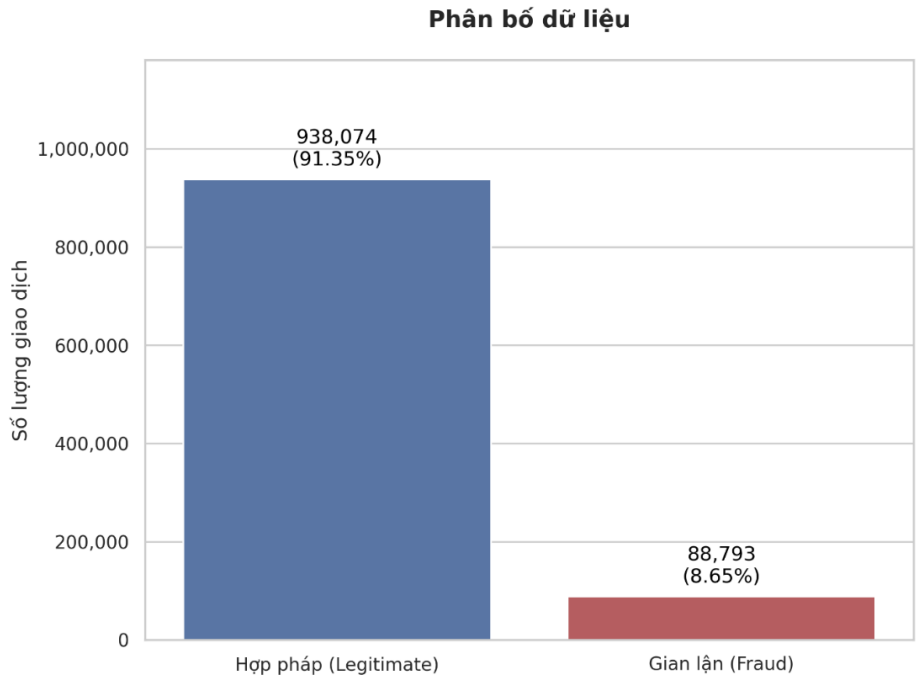
2.2.2. Khai phá dữ liệu

Bản dữ liệu dùng trong nghiên cứu được gộp từ hai tệp CSV do BCCC công bố: DeFiTransLyzer_fraud.csv và DeFiTransLyzer_legitimate.csv. Sau gộp, tập dữ liệu có 1.026.867 dòng và 94 cột, trong đó lớp gian lận chiếm khoảng 8,65%. Phân bố nhãn ở mức mất cân bằng vừa phải, phù hợp với các thước đo dựa trên xếp hạng như PR-AUC và các chỉ số tại ngưỡng ưu tiên cố định. Một số bản ghi tiêu biểu từ bộ dữ liệu được minh họa trong bảng 2.1 như sau:

Bảng 2. 1: Các bản ghi dữ liệu minh họa do BCCC công bố

address	block	nonce_med	cumuGas_avg	...	gasUsed_sum	cumuGas_sum	label
0x3bc...2b04e	2,279,473	1,196.5	1,158,476	...	58,856,326	2,905,457,455	0
0x38a...76e3a	5,046,713	161,815.5	2,054,311	...	26,436,938	1,992,681,231	0
0x2d3...5b098	2,311,909	347,050.0	864,696	...	43,497,586	1,429,341,801	0
0x32E...d180E	5,750,367	7.0	2,357,627	...	735,000	82,516,931	1
0x5be...A3986	5,329,061	512.5	1,943,765	...	1,218,000	112,738,394	1

Về thời gian, dữ liệu bao phủ dải block_number từ 46.468 đến 14.999.970, cho phép tách theo trật tự khối để hạn chế rò rỉ tương lai. Cấu trúc trường gồm 88 cột số và 5 cột phân loại/chuỗi (không tính cột nhãn), phản ánh các thống kê tổng hợp theo nhiều cấp (ví dụ nhóm nonce.*, cumulativeGasUsed.*). Kiểm định Kolmogorov–Smirnov cho thấy nhiều đặc trưng có khác biệt phân phối rõ giữa hai lớp (đặc biệt ở nhóm nonce và cumulativeGasUsed). Một số trường mang tính kỹ thuật hoặc có nguy cơ rò rỉ nhãn như flag hay Unnamed: 0 được ghi nhận để loại bỏ ở mục tiền xử lý.



Hình 2. 1: Phân bố dữ liệu

2.2.3. Tiền xử lý dữ liệu

Quy trình tiền xử lý chuẩn hóa dữ liệu đầu vào cho hai nhóm mô hình (Dữ liệu bảng và Dữ liệu chuỗi), tuân thủ nguyên tắc bảo toàn trật tự thời gian của các khối. Các bước thực hiện chính bao gồm:

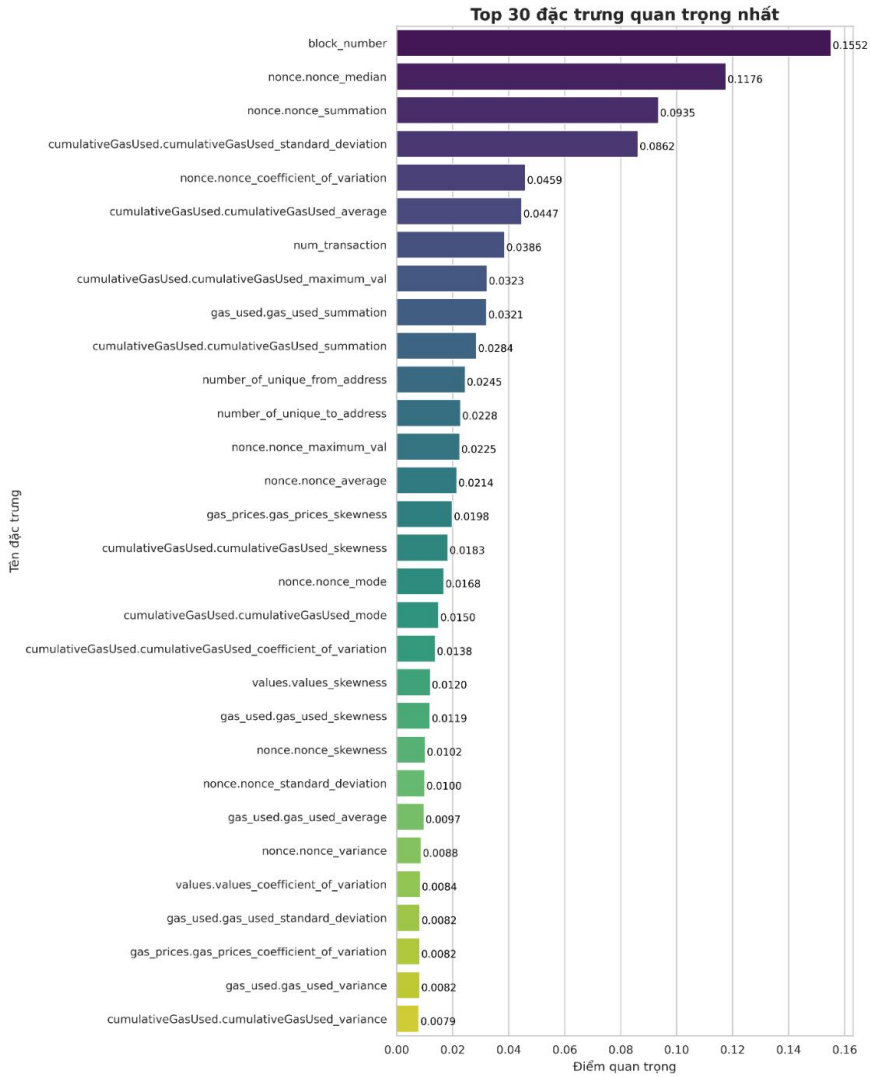
- **Làm sạch và chọn lọc:** Loại bỏ các trường gây rò rỉ dữ liệu (leakage) hoặc không mang thông tin dự báo. Đối với dữ liệu chuỗi, trường address được giữ lại để gom nhóm giao dịch nhưng không đưa vào làm đặc trưng huấn luyện.
- **Xử lý giá trị:** Điền giá trị khuyết bằng trung vị (với biến số) hoặc giá trị phổ biến nhất (với biến phân loại); các biến định tính được mã hóa số học (LabelEncoder).
- **Chuẩn hóa phân phối:** Xử lý dữ liệu nặng đuôi bằng phương pháp cắt ngoại lai (Clipping) trong khoảng phân vị [0,1%; 99,9%]. Đồng thời, loại bỏ các đặc trưng hằng số hoặc có độ tương quan dư thừa (> 0,95) để giảm chiều dữ liệu.

2.2.3.1. Tuyến bảng đặc trưng cho học máy

Tuyến này cấp dữ liệu cho nhóm mô hình học máy, đồng thời cũng là đầu vào thuận tiện cho Dense Autoencoder. Sau bước làm sạch, tập đặc trưng ban đầu được lựa chọn lại bằng một mô hình rừng ngẫu nhiên (Random Forest) huấn luyện trên mẫu con phân tầng nhằm xếp hạng tầm quan trọng. Danh mục cuối cùng giữ lại ba mươi đặc trưng có đóng góp cao nhất, trong đó nổi bật là các thống kê của nonce.*, cumulativeGasUsed.*, các thước đo gas_used.*, biến thời gian block_number và một số đặc trưng mô tả độ phân tán như values.* và gas_prices.*. Danh sách này được cố định cho toàn bộ thực nghiệm để bảo đảm tính nhất quán giữa các mô hình.

Chuẩn hóa được áp dụng ở giai đoạn huấn luyện bằng bộ chuẩn hóa bền vững với ngoại lai (RobustScaler), bộ chuẩn hóa này chỉ được khớp trên phần huấn luyện rồi suy rộng sang kiểm định và kiểm thử. Cách làm này phản ánh đúng kịch bản phát hiện bất thường không giám sát, tránh lan truyền thông tin từ tương lai về quá khứ.

Định dạng xuất cho tuyến bảng đặc trưng. Dữ liệu được xuất đồng thời ở hai định dạng. Định dạng văn bản phân cách cột (CSV) bảo đảm tính tương thích rộng với mọi thư viện học máy cổ điển, thuận tiện cho việc tái tạo thí nghiệm hoặc kiểm tra chéo trên các nền tảng khác nhau. Bên cạnh đó, dữ liệu cũng được xuất dưới dạng cột theo chuẩn Parquet. Định dạng Parquet lưu trữ theo cột và nén hiệu quả nên giảm kích thước, tăng tốc đọc chọn lọc cột, giữ chắc kiểu dữ liệu và hạn chế trôi kiểu khi chuyển đổi giữa các môi trường.



Hình 2. 2: Top 30 đặc trưng quan trọng nhất

2.2.3.2. Tuyến chuỗi thời gian cho học sâu theo trình tự

Tuyến này cấp dữ liệu cho LSTM-AE và Transformer-VAE, nơi cấu trúc thời gian và ngữ cảnh theo địa chỉ là trọng yếu. Dữ liệu được nhóm theo địa chỉ, sắp xếp theo block_number và trượt cửa sổ độ dài hai mươi bước để tạo các mẫu chuỗi. Nhãn của một chuỗi bằng một nếu trong cửa sổ có ít nhất một giao dịch gán nhãn gian lận, cách định nghĩa này giúp mô hình học được tín hiệu bối cảnh trước khi sự kiện xảy ra. Trường block_number chỉ dùng cho việc sắp xếp và tách theo thời gian, không được đưa vào danh mục đặc trưng để tránh mô hình dựa máy móc vào xu hướng thời gian. Sau làm sạch, số đặc trưng đưa vào mỗi bước thời gian là bảy mươi sáu.

Chuẩn hóa không được thực hiện ở bước tiền xử lý mà được dời sang notebook huấn luyện để khớp trên phần huấn luyện chỉ gồm các chuỗi hợp lệ, rồi áp dụng cho kiểm định và kiểm thử. Cách bố trí này tránh ô nhiễm thống kê do trộn lẫn các cửa sổ có sự kiện nghi vấn vào quá trình ước lượng tham số chuẩn hóa.

Định dạng xuất cho tuyến chuỗi thời gian. Dữ liệu được đóng gói dưới dạng NPZ của NumPy, bao gồm mảng X có hình dạng $N \times L \times F$, trong đó N là số chuỗi, L là độ dài cửa sổ và F là số đặc trưng, kèm theo mảng y chứa nhãn chuỗi, mảng features ghi danh sách tên đặc trưng, mảng addresses lưu định danh địa chỉ của mỗi chuỗi và mảng end_block chỉ mốc khối kết thúc cửa sổ. Định dạng NPZ đặc biệt phù hợp với mô hình học sâu vì loại bỏ bước trung gian chuyển đổi dữ liệu khung dữ liệu sang mảng số, tải nhanh trực tiếp vào bộ nhớ của thư viện huấn luyện, bảo toàn thứ tự thời gian và siêu thông tin cần thiết cho tách theo thời gian cũng như phân tích sau huấn luyện. Việc chuẩn hóa kiểu số thống nhất ngay từ lúc xuất (ví dụ dùng float32) còn giúp giảm dấu chân bộ nhớ và tăng thông lượng khi huấn luyện trên thiết bị tăng tốc.

2.2.4. Kiểm soát chất lượng

Sau khi xuất, dữ liệu được kiểm tra về tính toàn vẹn: tệp tồn tại và không rỗng, giá trị số hữu hạn, nhãn nhị phân hợp lệ, số cột khớp với danh mục đặc trưng đã định, tỉ lệ dương tính ổn định quanh mức quan sát ở dữ liệu gốc. Toàn bộ phép biến đổi như mã hóa nhãn, chuẩn hóa và lựa chọn đặc trưng đều được cố định hạt giống ngẫu nhiên, chỉ khớp trên phần huấn luyện, sau đó áp dụng nguyên vẹn cho kiểm định và kiểm thử. Các bước này bảo đảm kết quả đánh giá phản ánh đúng năng lực mô hình trong bối cảnh dữ liệu có trật tự thời gian và hạn chế rủi ro rò rỉ thông tin.

2.3. Thiết kế mô hình huấn luyện

2.3.1. Pipeline huấn luyện chung

Mục tiêu của pipeline là thiết lập một quy trình nhất quán, áp dụng đồng bộ cho cả sáu mô hình, từ khâu nạp dữ liệu đến báo cáo kết quả. Tất cả mô hình đều trả về điểm bất thường cùng chiều (điểm càng lớn càng đáng nghi), được tối ưu siêu tham số trên tập kiểm định và đánh giá mù trên tập kiểm tra.

Bước 1: Nạp dữ liệu và kiểm tra chất lượng

Nhóm học máy sử dụng dữ liệu bảng ở định dạng CSV đã qua chọn đặc trưng. Dense Autoencoder sử dụng cùng cấu trúc dữ liệu bảng nhưng ở Parquet để ổn định kiểu và tốc độ I/O. LSTM-AE và Transformer-VAE sử dụng NPZ tuần tự: mảng *X* kích thước (số chuỗi, độ dài chuỗi, số đặc trưng), mảng nhãn, danh sách tên đặc trưng và cột end_block cho tách theo thời gian.

Bước 2: Tách tập dữ liệu

- **Phương pháp chia theo thời gian:** Đối với dữ liệu có mốc thời gian tin cậy (end_block), áp dụng phân chia theo trật tự khối: 60% đầu cho huấn luyện (Train), 20% tiếp theo cho kiểm định (Validation), và 20% cuối cùng cho kiểm tra (Test).

Cách này mô phỏng chính xác kịch bản triển khai thực tế: mô hình học từ quá khứ để dự báo tương lai.

- **Phương pháp chia ngẫu nhiên:** Trong trường hợp không có mốc thời gian, áp dụng tỉ lệ 60/20/20 có phân tầng để duy trì tỉ lệ nhãn gian lận. Mọi thông tin về ranh giới tách đều được ghi lại để đảm bảo tính tái lập.

Bước 3: Làm sạch và ổn định phân phối

Các quy tắc xử lý giá trị thiếu, vô hạn và ngoại lai được mô tả chi tiết tại mục 2.2.3. Tại bước này, điều quan trọng là mọi thống kê phục vụ làm sạch (như trung vị, phân vị cắt biên, giá trị thay thế) đều chỉ được ước lượng trên tập Train (hoặc Train-Legit), sau đó giữ cố định và áp dụng nguyên vẹn cho Validation và Test nhằm ngăn rò rỉ thông tin và bảo toàn kịch bản triển khai thực tế.

Bước 4: Chuẩn hoá thang đo đặc trưng

- **Dữ liệu bảng:** Sử dụng RobustScaler (dựa trên trung vị và khoảng tứ phân vị) để giảm thiểu tác động của nhiễu. Scaler được khớp trên Train và biến đổi cho Validation/Test.
- **Dữ liệu chuỗi:** Scaler được khớp trên tập Train-Legit (chỉ các chuỗi bình thường) bằng cách làm phẳng dữ liệu theo thời gian, sau đó áp dụng đồng nhất cho các tập còn lại. Điều này giúp các mô hình Autoencoder học chính xác phân phối của hành vi bình thường.

Bước 5: Sinh điểm bất thường và quy ước cùng chiều

Mọi mô hình đều được chuẩn hóa đầu ra theo quy ước: Điểm càng lớn, khả năng gian lận càng cao.

- **Nhóm học máy:** Chuyển đổi các điểm gốc (khoảng cách cụm, độ cô lập, mật độ) về chiều dương quy ước.

- **Nhóm học sâu:** Sử dụng lỗi tái tạo (Reconstruction Error). Đối với Transformer-VAE, điểm số có thể kết hợp thêm thành phần năng lượng từ không gian ẩn (Latent Energy) để tăng độ nhạy.

Lưu ý: Có thể áp dụng phép biến đổi đơn điệu (như Z-score) trên tập Validation để đưa điểm số của các mô hình về cùng một thang đo phục vụ trực quan hóa, nhưng đảm bảo không làm thay đổi thứ hạng và kết quả đánh giá.

Bước 6: Tối ưu siêu tham số

Sử dụng thư viện Optuna (với thuật toán TPE và Median Pruner) để tìm cấu hình tối ưu trên tập Validation. Hàm mục tiêu tổng hợp J được thiết kế để cân bằng các yếu tố:

$$J = 0.5 \times AP + 0.3 \times ROC - AUC + 0.2 \times Precision@1\%$$

Trong đó:

- AP (Average Precision): Ưu tiên khả năng xếp hạng trên dữ liệu mất cân bằng.
- ROC-AUC: Đánh giá khả năng phân tách tổng quát giữa hai lớp.
- Precision@1%: Đảm bảo độ chính xác cao ở đỉnh danh sách cảnh báo (quan trọng cho vận hành thực tế).

Để tăng tốc, quá trình này sử dụng tập con (Subsample): tối đa 300.000 mẫu cho dữ liệu bảng và 100.000 chuỗi cho dữ liệu tuần tự.

Bước 7: Huấn luyện toàn cục

Sau khi xác định bộ tham số tối ưu, mô hình được huấn luyện lại trên toàn bộ tập Train (hoặc Train-Legit). Các kỹ thuật kiểm soát quá khớp (Overfitting) như Early Stopping và ReduceLROnPlateau được áp dụng nghiêm ngặt.

Bước 8: Chọn ngưỡng

Ngưỡng quyết định (Threshold) được cố định tại điểm cho chỉ số F1 cao nhất trên tập Validation.

Bước 9: Đánh giá

Ngưỡng này được áp dụng trực tiếp lên tập Test (chưa từng được mô hình nhìn thấy) để tính toán các chỉ số cuối cùng:

- Nhóm không phụ thuộc ngưỡng: AP, ROC-AUC, Precision@1%.
- Nhóm phụ thuộc ngưỡng: F1, Precision, Recall, Accuracy, Confusion Matrix.

Bước 10: Lưu và tái lập thực nghiệm

Toàn bộ thành phần thực nghiệm bao gồm: Scaler, mô hình, cấu hình tham số tốt nhất và các thống kê tham chiếu đều được lưu trữ để phục vụ việc tái lập kết quả hoặc triển khai sau này.

2.3.2. Thiết kế huấn luyện mô hình học máy

2.3.2.1. K-Means

Thiết lập dữ liệu và không gian tìm kiếm

Dữ liệu đầu vào là bảng đặc trưng (CSV) đã qua xử lý, được chia tập theo thời gian (60-20-20) và chuẩn hóa bằng RobustScaler (khớp trên tập huấn luyện). Không gian siêu tham số được thiết kế mở rộng để thuật toán Optuna tự động khám phá cấu trúc dữ liệu:

- **Số cụm (K):** Khảo sát trong khoảng [1, 500]. Khoảng tìm kiếm rộng cho phép mô hình tự cân bằng giữa độ phân giải cụm và chi phí tính toán.
- **Khởi tạo (n_init, init):** Số lần khởi tạo lại trong khoảng [8, 40] với các phương pháp k-means++ hoặc random nhằm giảm thiểu rủi ro rơi vào cực trị địa phương.

- **Tích hợp PCA (Principal Component Analysis):** Đây là điểm nhấn trong thiết kế. Optuna sẽ quyết định có áp dụng PCA hay không (use_pca), và nếu có thì số chiều (n_components) nằm trong khoảng [8, 32]. Việc này giúp cải thiện điều kiện số và loại bỏ nhiễu trước khi tính khoảng cách Euclid.
- **Thuật toán:** Sử dụng biến thể Elkan để tăng tốc độ tính toán khoảng cách trên tập dữ liệu lớn.

Tối ưu và huấn luyện

Quá trình tối ưu sử dụng bộ lấy mẫu TPE (Tree-structured Parzen Estimator) của Optuna với hàm mục tiêu tổng hợp J (đã mô tả tại mục 2.3.1). Điểm đặc biệt trong quy trình này là bước tinh chỉnh cục bộ (Local Refinement): sau khi Optuna tìm ra cấu hình tốt nhất, một vòng lặp nhỏ sẽ thử thay đổi nhẹ các tham số (K, n_init) xung quanh điểm tối ưu và chỉ cập nhật nếu hàm mục tiêu J tăng lên rõ rệt.

Đánh giá

Cấu hình tối ưu cuối cùng được huấn luyện lại trên toàn bộ tập huấn luyện (bao gồm cả bước PCA nếu được chọn). Ngưỡng quyết định được cố định dựa trên F1-score cao nhất trên tập kiểm định. Các thành phần thực nghiệm (Scaler, PCA model, K-Means model) được lưu trữ đầy đủ để phục vụ đánh giá mù trên tập kiểm tra và tái lập kết quả.

2.3.2.2. Isolation Forest

Thiết lập dữ liệu và không gian tìm kiếm

Dữ liệu đầu vào tuân thủ quy trình chung (CSV, tách tập theo thời gian 60-20-20, chuẩn hóa RobustScaler). Quá trình tối ưu siêu tham số tập trung vào việc cân bằng giữa độ ổn định và khả năng phát hiện mẫu hiếm:

- **Số cây (n_estimators):** Khảo sát trong khoảng [200, 800]. Số lượng cây lớn giúp điểm số hội tụ ổn định hơn.

- **Tỷ lệ mẫu con (max_samples):** Từ 0.3 đến 1.0. Việc lấy mẫu con (subsampling) là yếu tố then chốt để tạo sự đa dạng cho các cây và tránh hiện tượng che lấp (swamping).
- **Tỷ lệ đặc trưng (max_features):** Từ 0.3 đến 1.0.
- **PCA (Tùy chọn):** Optuna tự động quyết định có áp dụng giảm chiều dữ liệu (xuống 8-32 chiều) trước khi đưa vào rừng cây hay không, nhằm loại bỏ nhiễu và tăng tốc độ tính toán.

Tối ưu và huấn luyện

Sử dụng Optuna với chiến lược lấy mẫu TPE để tối đa hóa hàm mục tiêu J trên tập kiểm định. Tham số contamination được thiết lập bằng tỷ lệ gian lận thực nghiệm (fraud_ratio) chỉ để phục vụ việc chuẩn hóa điểm số nội bộ, không dùng để cắt ngưỡng cứng (hard threshold).

Đánh giá

Sau khi xác định cấu hình tối ưu, mô hình được huấn luyện lại toàn cục trên tập huấn luyện. Ngưỡng vận hành được xác định dựa trên F1-score tốt nhất trên tập kiểm định và áp dụng cho tập kiểm tra. Toàn bộ mô hình và bộ tham số được lưu trữ để đảm bảo tính tái lập.

2.3.2.3. Histogram-Based Outlier Score

Thiết lập dữ liệu và không gian tìm kiếm

Dữ liệu đầu vào tuân thủ quy trình chung (CSV, tách tập theo thời gian 60-20-20, chuẩn hóa RobustScaler). Không gian siêu tham số tập trung vào việc kiểm soát độ mịn của histogram và xử lý tương quan đặc trưng:

- **Số lượng thùng (n_bins):** Khảo sát trong khoảng [20, 200]. Số thùng quyết định độ phân giải của mô hình hóa mật độ.

- **Hệ số làm trơn (alpha):** Từ [0.05, 0.999]. Tham số này giúp điều hòa phân phối, tránh hiện tượng zero-probability cho các giá trị hiếm gặp.
- **Ngưỡng cắt (tol):** Từ [0.001, 0.5] để loại bỏ ảnh hưởng của các bin có mật độ quá thấp (nhiễu).
- **PCA (Tùy chọn):** Do HBOS giả định các đặc trưng độc lập, việc tích hợp PCA (tìm kiếm 8-32 chiều) là bước xử lý quan trọng để khử tương quan (decorrelation) giữa các biến đầu vào, giúp thỏa mãn giả định lý thuyết của mô hình.

Tối ưu và huấn luyện

Sử dụng Optuna với chiến lược TPE để tối đa hóa hàm mục tiêu J trên tập kiểm định. Một bước tinh chỉnh cục bộ (local refinement) được áp dụng sau khi tìm được vùng tham số hứa hẹn để đạt hiệu năng cao nhất.

Đánh giá

Cấu hình tối ưu được huấn luyện lại trên toàn bộ tập huấn luyện. Ngưỡng quyết định được xác định dựa trên F1-score tốt nhất trên tập kiểm định. Toàn bộ mô hình, bộ tham số và scaler được lưu trữ. Kết quả đánh giá bao gồm cả các chỉ số phụ thuộc và không phụ thuộc ngưỡng, đảm bảo tính khách quan khi so sánh với các phương pháp phức tạp hơn.

2.3.3. Thiết kế huấn luyện mô hình học sâu

2.3.3.1. Dense Autoencoder

Thiết lập dữ liệu và kiến trúc mạng

- **Dữ liệu:** Khác với các mô hình trước, Dense Autoencoder chỉ sử dụng tập Train-Legit (các mẫu nhãn 0) để huấn luyện và khớp RobustScaler. Dữ liệu Parquet được ưu tiên sử dụng để tăng tốc độ.

- **Kiến trúc:** Mạng fully-connected đối xứng qua nút cổ chai. Để tăng tính bền vững (robustness), một lớp GaussianNoise(0.01) được đặt ngay đầu vào (Denoising Autoencoder). Sau mỗi lớp ẩn đều áp dụng BatchNormalization và Dropout.
- **Hàm mất mát (Loss):** Sử dụng Huber Loss thay vì MSE để giảm tác động của các giá trị ngoại lai trong quá trình cập nhật trọng số.

Tối ưu và huấn luyện

Quá trình tối ưu sử dụng Optuna với 40 thử nghiệm trên tập con Train-Legit. Không gian tìm kiếm siêu tham số được thiết kế như sau:

- **Kiến trúc (w_1, w_2, bottleneck):** Số nơ-ron lớp mã hóa lần lượt chọn trong các tập {256, 384, 512}, {128, 192, 256} và nút cổ chai {8, 16, 32}.
- **Tỷ lệ Dropout:** Khảo sát liên tục trong khoảng [0.08, 0.25].
- **Hệ số L2 Regularization:** Tìm kiếm theo thang logarit từ 10^{-7} đến 10^{-4}.
- **Tốc độ học (Learning Rate):** Từ 2×10^{-4} đến 2×10^{-3}.
- **Hàm kích hoạt:** Lựa chọn giữa ReLU và GELU.

Đánh giá

Cấu hình tốt nhất được huấn luyện lại (refit) trên toàn bộ tập Train-Legit trong 80 epochs với cơ chế EarlyStopping. Các thống kê tham chiếu (median, MAD) được tính toán lại trên tập Val-Legit và lưu trữ. Ngưỡng quyết định được chọn theo F1-score tối đa trên Validation và áp dụng mù cho Test.

2.3.3.2. Long Short-term Memory Autoencoder

Thiết lập dữ liệu và chuẩn hóa

- **Dữ liệu:** Sử dụng file NPZ chứa các chuỗi thời gian tuần tự.
- **Tách tập:** Tuân thủ phân chia theo thời gian (60-20-20) dựa trên end_block.
- **Chuẩn hóa:** Quy trình chuẩn hóa phức tạp hơn so với dữ liệu bảng:

1. Thay thế vô cực/thiếu bằng trung vị học từ Train-Legit.
2. Cắt biên (clipping) theo phân vị [0.1%-99.9%].
3. Áp dụng RobustScaler (được khớp trên dữ liệu Train-Legit đã làm phẳng) cho toàn bộ các tập dữ liệu để đảm bảo tính nhất quán.

Kiến trúc mạng

- **Encoder:** Sử dụng 1-2 lớp LSTM (tùy chọn Bidirectional do Optuna quyết định) để nén chuỗi đầu vào thành một vector ẩn (latent vector) tại bước thời gian cuối cùng.
- **Decoder:** Sử dụng lớp RepeatVector để lặp lại vector ẩn L lần, làm đầu vào cho các lớp LSTM đối xứng nhằm khôi phục lại trình tự thời gian.
- **Output:** Lớp TimeDistributed (Dense) được sử dụng ở tầng cuối cùng để tái tạo lại không gian đặc trưng ban đầu F tại mỗi bước thời gian. Đầu ra được ép kiểu tường minh về float32 để đảm bảo tính ổn định số học.

Tối ưu hóa và huấn luyện

Sử dụng Huber Loss để bền vững với nhiễu ngoại lai. Để ổn định quá trình huấn luyện với độ chính xác hỗn hợp (mixed precision fp16), đầu ra cuối cùng được ép kiểu về float32.

Optuna tìm kiếm cấu hình tối ưu trên tập con 100.000 chuỗi với không gian tham số:

- **Số đơn vị ẩn (Units):** Lớp LSTM thứ nhất thuộc {64, 128, 256}, lớp thứ hai thuộc {32, 64, 128}.
- **Cấu trúc:** Tùy chọn sử dụng LSTM hai chiều (Bidirectional) hoặc một chiều.
- **Tỷ lệ Dropout:** Khảo sát trong khoảng [0.1, 0.4].
- **Tốc độ học:** Tìm kiếm thang logarit từ 10^{-4} đến 3×10^{-3}.

- **Kích thước Batch:** Lựa chọn giữa các mức lớn {256, 512, 1024} để tận dụng khả năng tính toán song song của GPU.

Đánh giá

Mô hình tối ưu được huấn luyện lại trên toàn bộ tập Train-Legit. Ngưỡng vận hành được xác định dựa trên F1-score tốt nhất trên tập Validation và giữ nguyên khi đánh giá trên tập Test. Kết quả được báo cáo đầy đủ theo bộ chỉ số chuẩn đã định nghĩa.

2.3.3.3. Transformer Variational Autoencoder

Thiết lập dữ liệu và chuẩn hóa

Tương tự các mô hình chuỗi khác, dữ liệu được chuẩn hóa nghiêm ngặt trên tập Train-Legit (impute median, clip outliers, RobustScaler). Tuy nhiên, Transformer-VAE có tùy chọn sử dụng cơ chế tự chuẩn hóa nội bộ (LayerNorm) và bỏ qua scaler bên ngoài (EXTERNAL_SCALED=False) nếu cấu hình tối ưu yêu cầu, giúp mô hình học trực tiếp từ dữ liệu thô đã làm sạch.

Kiến trúc mạng chuyên biệt

Mô hình được thiết kế với các khối đặc thù cho dữ liệu chuỗi thời gian tài chính:

- **Embedding:** Sử dụng lớp Time2Vec để học các đặc trưng chuỗi thời gian (tính chu kỳ) kết hợp với LearnablePositionalEmbedding để mã hóa vị trí trong chuỗi.
- **Encoder & Decoder:** Sử dụng các khối Transformer (Multi-Head Attention + Feed Forward) thay vì RNN, cho phép xử lý song song và bắt các phụ thuộc dài hạn hiệu quả hơn.
- **Không gian ẩn (Latent Space):** Tầng giữa sử dụng cơ chế VAE (Reparameterization Trick) để lấy mẫu vector ẩn $z = \mu + \sigma \odot \varepsilon$ với $\varepsilon \sim N(0,1)$.

Tối ưu và huấn luyện

Tối ưu hóa (Optuna): Tìm kiếm số lượng Attention Heads, số khối Transformer, kích thước vector ẩn, tỷ lệ Dropout và tốc độ học trên tập con 100.000 chuỗi.

Không gian tìm kiếm cho kiến trúc Transformer bao gồm:

- **Kích thước nhúng (embed_dim):** Chọn trong tập {64, 96, 128}.
- **Số đầu chú ý (num_heads):** {4, 8} (đảm bảo chia hết cho kích thước nhúng).
- **Cấu trúc khối:** Số khối Transformer (1 hoặc 2) và hệ số mở rộng Feed-Forward (2× hoặc 4×).
- **Không gian ẩn (latent):** Kích thước vector biến phân thuộc {8, 12, 16, 24}.

Huấn luyện: Tỷ lệ Dropout [0.15, 0.35] và tốc độ học từ 10^{-4} đến 2×10^{-3}.

Đánh giá

Điểm số cuối cùng trên tập Test được tính toán bằng cách lấy trung bình qua nhiều lần chạy (Monte Carlo sampling) với các mặt nạ che ngẫu nhiên khác nhau (repeats=3) để tăng độ tin cậy. Ngưỡng quyết định F1 trên Validation được dùng để đánh giá hiệu năng cuối cùng.

2.4. Các thước đo đánh giá hiệu năng

Việc đánh giá mô hình phát hiện gian lận trên dữ liệu tài chính đối mặt với thách thức lớn về sự mất cân bằng dữ liệu (lớp gian lận chỉ chiếm khoảng 8,65%). Trong bối cảnh này, các chỉ số truyền thống như độ chính xác toàn cục (Accuracy) trở nên sai lệch và thiếu ý nghĩa (một mô hình dự đoán tất cả giao dịch là hợp pháp vẫn đạt độ chính xác >91% nhưng thất bại hoàn toàn trong việc phát hiện gian lận).

Do đó, nghiên cứu xây dựng một hệ thống đánh giá đa chiều, tập trung vào hai khía cạnh: năng lực xếp hạng rủi ro (Ranking Quality) và hiệu quả ra quyết định (Decision Quality).

2.4.1. Ma trận nhầm lẫn

Mọi chỉ số quyết định đều được dẫn xuất từ ma trận nhầm lẫn, bao gồm 4 thành phần cơ bản khi so sánh nhãn dự đoán với nhãn thực tế:

- True Positive (TP - Dương tính thật): Giao dịch gian lận được phát hiện chính xác.
- False Positive (FP - Dương tính giả): Giao dịch hợp pháp bị báo nhầm là gian lận (Báo động giả). Sẽ gây phiền toái cho người dùng, lãng phí chi phí điều tra.
- True Negative (TN - Âm tính thật): Giao dịch hợp pháp được phân loại đúng.
- False Negative (FN - Âm tính giả): Giao dịch gian lận bị bỏ lọt. Sẽ gây thiệt hại tài chính trực tiếp.

2.4.2. Nhóm chỉ số đánh giá xếp hạng

Nhóm chỉ số này đánh giá chất lượng của điểm số bất thường (anomaly score) mà không phụ thuộc vào việc chọn ngưỡng cắt cụ thể. Đây là cơ sở để so sánh khả năng xếp hạng giữa các mô hình.

2.4.2.1. Precision-Recall Curve (PR-AUC / AP)

Đây là thước đo quan trọng nhất đối với dữ liệu mất cân bằng. PR-AUC tóm tắt sự đánh đổi giữa độ chính xác (Precision) và độ phủ (Recall) tại mọi ngưỡng cắt có thể.

- Công thức: $AP = \sum_n (R_n - R_{n-1}) P_n$
- Ý nghĩa: Giá trị AP càng gần 1.0 chứng tỏ mô hình có khả năng đẩy các giao dịch gian lận lên đầu danh sách xếp hạng mà không tạo ra quá nhiều báo động giả. Khác với ROC, đường PR không bị sai lệch bởi số lượng lớn các mẫu True Negative.

2.4.2.2. ROC Curve (ROC-AUC)

Đánh giá khả năng phân tách tổng quát giữa hai phân phối điểm số của lớp Gian lận và Hợp pháp.

- Công thức: Diện tích dưới đường cong biểu diễn mối quan hệ giữa Tỷ lệ dương tính thật (TPR) và Tỷ lệ dương tính giả (FPR).
- Ý nghĩa: ROC-AUC = 0.5 tương đương với đoán ngẫu nhiên; ROC-AUC = 1.0 là phân tách hoàn hảo. Chỉ số này giúp đánh giá độ ổn định tổng quát của mô hình.

2.4.2.3. Precision@1%

Đây là chỉ số định hướng vận hành thực tế. Trong thực tế, nguồn lực (nhân sự, thời gian) để kiểm tra thủ công các cảnh báo là hữu hạn.

- Định nghĩa: Tỷ lệ gian lận thực sự nằm trong top k% các giao dịch có điểm bất thường cao nhất.
- Công thức: $Precision@1\% = \frac{Số\ lượng\ mẫu\ gian\ lận\ trong\ top\ k\%}{Tổng\ số\ mẫu\ trong\ top\ k\%}$
- Ý nghĩa: Với k=1%, chỉ số này cho biết: Nếu chỉ điều tra 1% số giao dịch nghi ngờ nhất, thì bao nhiêu phần trăm trong đó là bắt đúng. Precision@1%cao giúp tối ưu hóa chi phí vận hành.

2.4.3. Nhóm chỉ số đánh giá quyết định

Để triển khai hệ thống, một ngưỡng quyết định cụ thể được xác định dựa trên việc tối đa hóa điểm F1-Score trên tập kiểm định. Tại ngưỡng này, các chỉ số sau được tính toán:

2.4.3.1. Precision (Độ chính xác dương tính)

- Công thức: $Precision = \frac{TP}{TP + FP}$

- Ý nghĩa: Phản ánh độ tin cậy của cảnh báo. Precision thấp đồng nghĩa với tỷ lệ báo động giả (False Alarm Rate) cao, gây ra hiện tượng quá tải cảnh báo cho hệ thống giám sát.

2.4.3.2. Recall (Độ nhạy / Độ phủ)

- Công thức: $Recall = \frac{TP}{TP + FN}$
- Ý nghĩa: Phản ánh khả năng bao quát rủi ro. Recall thấp đồng nghĩa với việc bỏ lọt tội phạm (False Negatives), dẫn đến rủi ro tài chính.

2.4.3.3. F1-Score

- Công thức: $F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$
- Ý nghĩa: Là trung bình điều hòa giữa Precision và Recall. F1-Score được chọn làm tiêu chí chính để tìm điểm cắt tối ưu, đảm bảo sự cân bằng giữa việc bắt được nhiều gian lận và giữ độ tin cậy của cảnh báo ở mức chấp nhận được.

2.4.3.4. Balanced Accuracy (Độ chính xác cân bằng)

- Công thức: $Balanced\ Accuracy = \frac{Recall + Specificity}{2}$
- Ý nghĩa: Là trung bình cộng của độ nhạy trên từng lớp. Chỉ số này khắc phục nhược điểm của Accuracy truyền thống, giúp đánh giá công bằng ngay cả khi lớp gian lận rất thiểu số.

2.4.4. Chiến lược tối ưu hóa tổng hợp

Trong giai đoạn tinh chỉnh siêu tham số (Hyperparameter Tuning) bằng Optuna, nghiên cứu không dựa vào một chỉ số đơn lẻ mà sử dụng Hàm mục tiêu lai ghép (Composite Objective Function) để định hướng mô hình hội tụ về điểm tối ưu toàn diện:

$$J = 0.5 \times AP + 0.3 \times ROC_AUC + 0.2 \times Precision@1\%$$

Trọng số:

- w_{AP} = 0.5 (Trọng số cao nhất): Ưu tiên tối đa cho chất lượng xếp hạng trên dữ liệu lệch, đảm bảo mô hình mạnh mẽ ở mọi điểm cắt.
- w_{ROC_AUC} = 0.3: Đảm bảo khả năng phân tách tổng quát, tránh việc mô hình bị quá khớp vào một vùng cục bộ của dữ liệu.
- $w_{Precision@1\%}$ = 0.2: Cưỡng bức mô hình phải có độ chính xác cao ở phần đỉnh của danh sách cảnh báo, phục vụ trực tiếp cho kịch bản phát hiện sớm trong vận hành.

2.5. Kết luận chương

Chương 2 đã thiết lập hoàn chỉnh khung phương pháp luận và quy trình thực nghiệm cho bài toán phát hiện gian lận DeFi. Các đóng góp chính bao gồm:

- **Quy trình xử lý dữ liệu:** Xây dựng pipeline tiền xử lý theo nguyên tắc phân chia theo thời gian và chuẩn hóa trên tập huấn luyện, giúp tránh rò rỉ dữ liệu và xử lý các đặc trưng phân phối lệch.
- **Thiết kế mô hình:** Xây dựng và chuẩn hóa kiến trúc, cơ chế sinh điểm và không gian siêu tham số cho 6 mô hình học máy và học sâu, bảo đảm khả năng so sánh trực tiếp.
- **Chiến lược tối ưu và đánh giá:** Việc sử dụng hàm mục tiêu lai ghép trong Optuna và chọn ngưỡng dựa trên F1-score giúp mô hình không chỉ tốt về mặt lý thuyết (xếp hạng) mà còn tối ưu cho kịch bản triển khai thực tế (ra quyết định).

Những thiết kế này tạo nền tảng vững chắc để tiến hành thực nghiệm. Chương 3 tiếp theo sẽ trình bày chi tiết các kết quả thực nghiệm, so sánh hiệu năng giữa các mô hình và phân tích sâu về các trường hợp phát hiện thành công cũng như sai sót điển hình.

CHƯƠNG 3. TRIỂN KHAI THỰC NGHIỆM VÀ ĐÁNH GIÁ KẾT QUẢ

3.1. Giới thiệu chung

Chương 3 chuyển hóa các thiết kế phương pháp luận tại Chương 2 thành quy trình thực nghiệm cụ thể nhằm kiểm chứng hiệu quả của các mô hình phát hiện gian lận. Mục tiêu cốt lõi là thực hiện huấn luyện, tối ưu hóa và đánh giá sáu mô hình đề xuất trên bộ dữ liệu BCCC-DeFiFraudTrans-2025. Nội dung chương được triển khai theo trình tự logic của quy trình thực nghiệm:

- Thiết lập môi trường: Công bố chi tiết cấu hình phần cứng và các thư viện phần mềm để đảm bảo tính minh bạch và khả năng tái lập kết quả.
- Thực nghiệm mô hình: Trình bày quá trình tìm kiếm siêu tham số (Hyperparameter Tuning) bằng thuật toán TPE (Optuna) và kết quả huấn luyện cho hai nhóm mô hình: Học máy và học sâu.
- Đánh giá và So sánh: Phân tích hiệu năng dựa trên hệ thống thước đo đa chiều đã định nghĩa (như PR-AUC, F1-Score, Precision@1%, v.v.), từ đó làm rõ ưu nhược điểm của từng phương pháp trong bối cảnh phát hiện gian lận giao dịch tài chính.

3.2. Môi trường và công cụ

Để đảm bảo tính nhất quán, hiệu quả tính toán và khả năng tái lập của các kết quả thực nghiệm, toàn bộ quy trình từ tiền xử lý đến đánh giá mô hình được thực hiện trên một hệ thống máy trạm hiệu năng cao (Workstation) với cấu hình phần cứng và môi trường phần mềm thống nhất.

3.2.1. Cấu hình phần cứng

Quá trình thực nghiệm đòi hỏi tài nguyên tính toán lớn để xử lý bộ dữ liệu bảng và huấn luyện các mô hình học sâu với kiến trúc phức tạp. Hệ thống thực nghiệm được

trang bị bộ nhớ RAM lớn để giảm thiểu độ trễ I/O và GPU mạnh mẽ hỗ trợ các kỹ thuật tăng tốc hiện đại. Chi tiết cấu hình được trình bày trong bảng dưới đây:

Bảng 3. 1: Cấu hình phần cứng

Thành phần	**Thông số kỹ thuật**	**Vai trò chính**
CPU	Dual Intel Xeon Gold 6148	Chịu trách nhiệm chính cho các tác vụ tiền xử lý dữ liệu đa luồng, tính toán thống kê và huấn luyện song song các mô hình học máy
RAM	256 GB DDR4	Đảm bảo khả năng nạp toàn bộ dữ liệu huấn luyện vào bộ nhớ, tránh hiện tượng nghẽn cổ chai khi truy xuất dữ liệu.
GPU	NVIDIA GeForce RTX 4090	Tăng tốc huấn luyện các mô hình học sâu. Dung lượng VRAM lớn cho phép thiết lập kích thước Batch lớn để tối ưu hóa tốc độ hội tụ.
Lưu trữ	SSD NVMe 1TB	Lưu trữ bộ dữ liệu để tối ưu tốc độ đọc tuần tự và lưu trữ các trọng số mô hình.

3.2.2. Công cụ và thư viện phần mềm

Thực nghiệm được triển khai hoàn toàn trên nền tảng Python 3.10, tận dụng hệ sinh thái mã nguồn mở mạnh mẽ cho khoa học dữ liệu.

Quản lý và xử lý dữ liệu:

- NumPy, Pandas: Thao tác trên ma trận và khung dữ liệu.
- PyArrow: Hỗ trợ đọc/ghi định dạng Parquet tốc độ cao.
- Scikit-learn: Cung cấp các công cụ chuẩn hóa (RobustScaler), giảm chiều (PCA) và chia tập dữ liệu.

Mô hình học máy:

- Scikit-learn: Triển khai thuật toán K-Means và các metric đánh giá cơ bản.
- PyOD (Python Outlier Detection): Triển khai các thuật toán chuyên biệt Isolation Forest và HBOS.

Mô hình học sâu:

- TensorFlow 2.10+ / Keras: Xây dựng và huấn luyện các kiến trúc mạng nơ-ron.
- Tối ưu hóa: Kích hoạt chế độ Mixed Precision (float16) và trình biên dịch XLA (Accelerated Linear Algebra) để khai thác tối đa sức mạnh của GPU.

Tối ưu siêu tham số và đánh giá:

- Optuna: Tự động hóa quá trình tìm kiếm siêu tham số với chiến lược TPE.
- Matplotlib, Seaborn: Trực quan hóa kết quả, biểu đồ Loss và các đường cong đánh giá (ROC/PR).

3.3. Kết quả huấn luyện mô hình

3.3.1. Mô hình học máy

3.3.1.1. K-Means

a) Phân tích kết quả tối ưu hóa tham số

Quá trình tối ưu Optuna hội tụ về cấu hình phản ánh sự phân mảnh mạnh trong hành vi giao dịch DeFi. Bảng dưới trình bày các tham số và ý nghĩa:

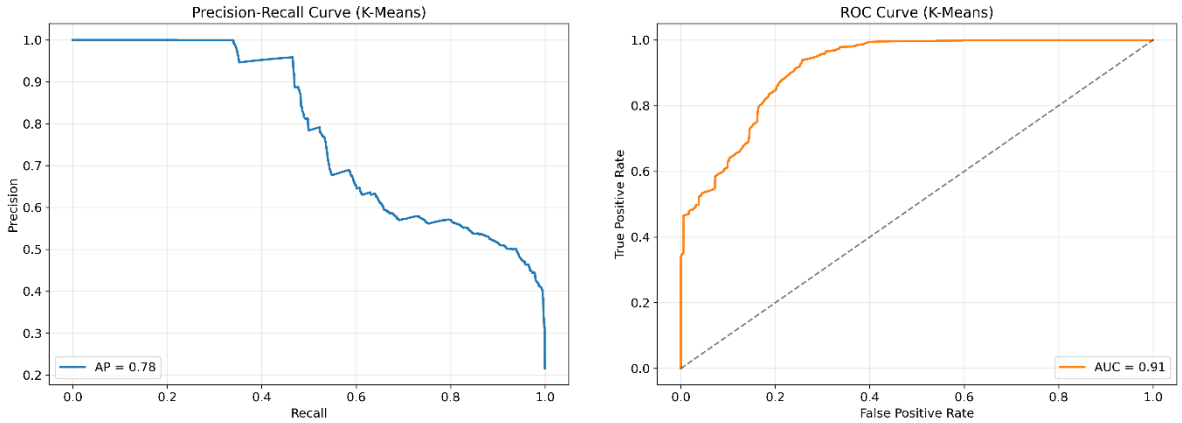
Bảng 3. 2: Cấu hình siêu tham số của K-Means

Tham số	Giá trị tối ưu	Phân tích
Số cụm (n_clusters)	483	Gần sát cận trên (500), cho thấy không gian hành vi rất phân mảnh, đòi hỏi nhiều cụm nhỏ để mô tả.

Số lần khởi tạo (n_init)	34	Giảm khả năng rơi vào cực tiểu địa phương trong không gian nhiều chiều.
Sử dụng PCA (use_pca)	True	Loại bớt nhiễu và tương quan, giúp khoảng cách Euclid phản ánh mức độ bất thường tốt hơn.
Số chiều PCA (n_components)	10	Cô đọng thông tin từ 30 đặc trưng xuống 10 thành phần chính hiệu quả.

b) Phân tích hiệu năng trên tập kiểm tra

- PR-AUC, ROC-AUC, Precision@1%
  - PR-AUC ≈ 0.7801, ROC-AUC ≈ 0.9146, Precision@1% = 1.0000.
  - Các chỉ số này cho thấy K-Means xếp hạng và phân tách giao dịch gian lận rất tốt; đặc biệt trong nhóm 1% giao dịch nghi ngờ nhất, toàn bộ đều là gian lận thật, nên mô hình cực kỳ hữu ích cho bước ưu tiên kiểm tra thủ công.

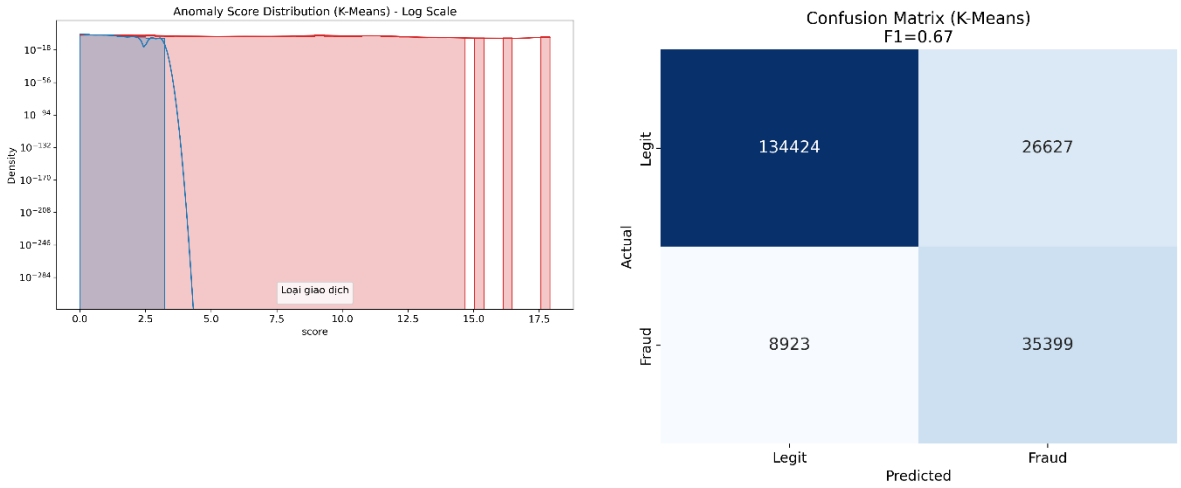


Hình 3. 1: PR Curve và ROC Curve của K-Means

- Phân phối điểm bất thường

- Giao dịch hợp lệ tập trung chủ yếu ở vùng điểm thấp, trong khi gian lận trải rộng về phía điểm cao, với một vùng chồng lấn ở phần đuôi phân phối lệch phải của lớp hợp lệ.
- Điều này phản ánh mô hình nhận diện khá rõ các hành vi gian lận lộ liễu, nhưng vẫn có một số giao dịch hợp lệ hiếm nằm ở biên cụm bị gán nhầm là bất thường.

- Ma trận nhầm lẫn tại ngưỡng tối ưu (threshold ≈ 0.5529)
  - F1-score ≈ 0.6657, Precision ≈ 0.5707, Recall ≈ 0.7987.
  - TN = 134 424, FP = 26 627, FN = 8 923, TP = 35 399.
  - Như vậy, mô hình phát hiện được phần lớn giao dịch gian lận với F1 tương đối cao, nhưng đổi lại vẫn tạo ra một lượng đáng kể cảnh báo giả.



Hình 3. 2: Phân phối điểm bất thường và ma trận nhầm lẫn của K-Means

c) Nhận xét

- Ưu điểm
  - PR-AUC và ROC-AUC cao, thể hiện khả năng xếp hạng và phân biệt gian lận tốt trên toàn dải ngưỡng.

- Precision@1% = 1.0000, giúp mô hình đặc biệt phù hợp cho việc khoanh vùng một tập rất nhỏ giao dịch rủi ro cao để phân tích sâu.

- Hạn chế
  - Precision chung chỉ khoảng 0.57, số lượng false positive còn lớn, nhất là ở các giao dịch hợp lệ nhưng hiếm.
  - Mô hình phụ thuộc mạnh vào khoảng cách Euclid trong không gian PCA, nên khó phân biệt giữa hợp lệ hiếm gặp và gian lận.

3.3.1.2. Isolation Forest

a) Phân tích kết quả tối ưu hóa tham số

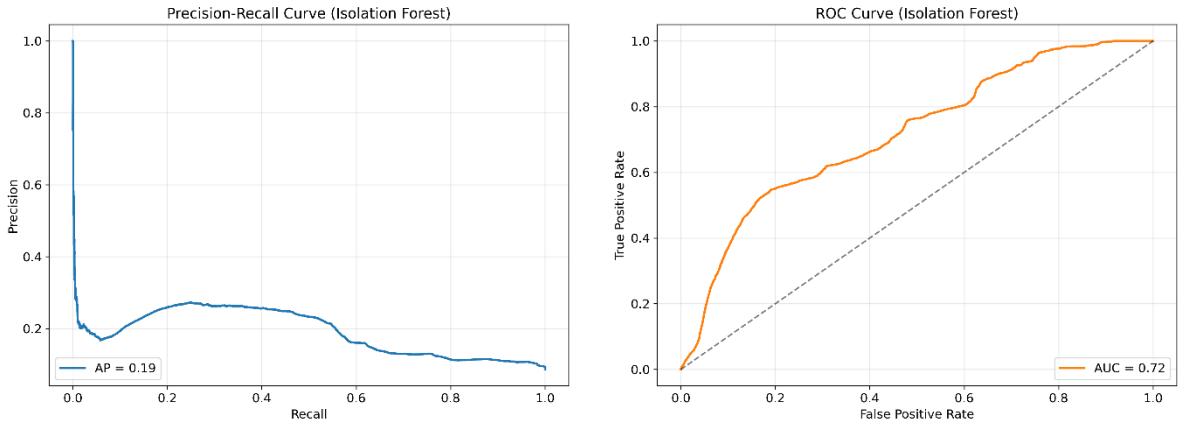
Quá trình tối ưu bằng Optuna đã chọn ra cấu hình phù hợp với bản chất rời rạc, nhiều nhiễu và phân phối lệch của giao dịch DeFi. Bảng dưới tóm tắt các siêu tham số và ý nghĩa của chúng:

Bảng 3. 3: Cấu hình siêu tham số của Isolation Forest

Tham số	Giá trị tối ưu	Phân tích
Sử dụng PCA (use_pca)	True	Giảm nhiễu và giúp rừng phân tách không gian hiệu quả hơn.
Số chiều PCA (n_components)	9	Giữ cấu trúc tổng quát, loại bỏ nhiễu.
Số cây (n_estimators)	606	Số cây lớn tăng độ ổn định khi phát hiện bất thường.
max_samples	0.8125	Mỗi cây dùng ≈ 81% dữ liệu, tăng tính ngẫu nhiên.
max_features	0.7244	Mỗi cây dùng 72% đặc trưng, tránh lệ thuộc vào feature mạnh.

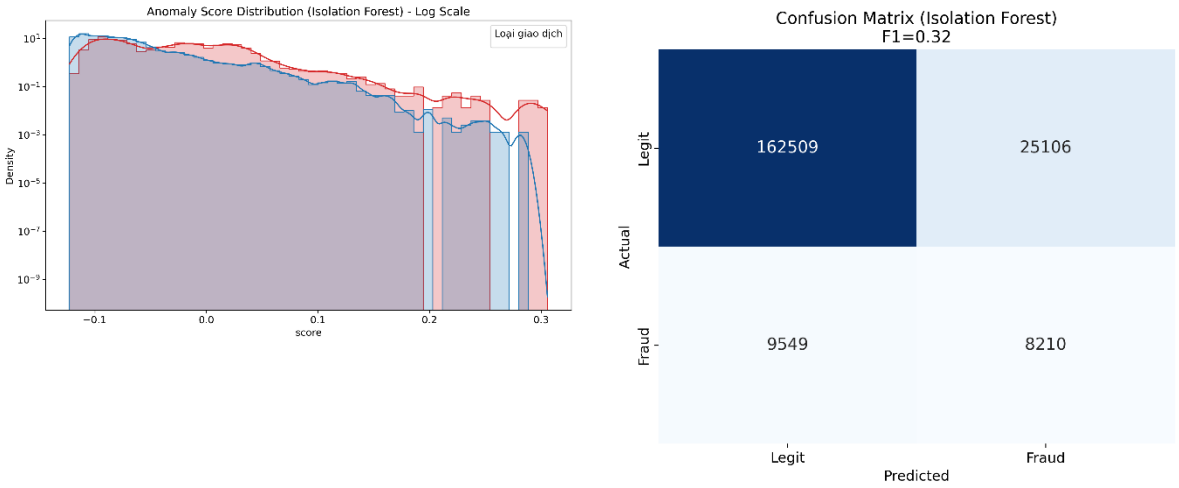
b) Phân tích hiệu năng trên tập kiểm tra

- PR-AUC, ROC-AUC, Precision@1%
    - PR-AUC ≈ 0.1902, ROC-AUC ≈ 0.7168, Precision@1% ≈ 0.2065.
    - Bộ chỉ số cho thấy mô hình tốt hơn ngẫu nhiên nhưng khả năng ưu tiên đúng các giao dịch gian lận trong top điểm cao còn yếu, nên kém phù hợp cho mục tiêu xếp hạng để kiểm tra thủ công.



Hình 3. 3: PR Curve và ROC Curve của Isolation Forest

- Phân phối điểm bất thường
    - Phân phối điểm của hai lớp gần như chồng lấn; cả giao dịch hợp lệ hiếm và gian lận đều có thể xuất hiện ở vùng điểm rất cao.
    - Điều này dẫn đến việc một số lượng lớn giao dịch hợp lệ bị gán nhãn bất thường, trong khi không phải tất cả gian lận đều được đẩy lên vùng điểm cao.
- Ma trận nhầm lẫn tại ngưỡng tối ưu (threshold ≈ –0.0299)
    - F1-score ≈ 0.3215, Precision ≈ 0.2464, Recall ≈ 0.4623.
    - TN = 162 509, FP = 25 106, FN = 9 549, TP = 8 210.
    - Mô hình bỏ sót hơn một nửa giao dịch gian lận và đồng thời tạo khá nhiều cảnh báo giả, cho thấy ranh giới được học chưa phù hợp với cấu trúc dữ liệu DeFi.



Hình 3. 4: Phân phối điểm bất thường và ma trận nhầm lẫn của Isolation Forest

c) Nhận xét

- Ưu điểm
  - Không yêu cầu giả định phân phối, vận hành được trên dữ liệu nhiễu và nhiều chiều.
  - ROC-AUC trên 0.7 cho thấy vẫn có mức phân tách nhất định giữa hai lớp.
- Hạn chế
  - PR-AUC và Precision@1% thấp, chất lượng xếp hạng top giao dịch nghi ngờ không đáp ứng yêu cầu cho tác vụ ưu tiên kiểm tra.
  - F1 thấp và Recall ≈ 0.46 khiến mô hình vừa bỏ sót nhiều gian lận, vừa tạo ra nhiều false positive do phân phối điểm chồng lấn mạnh.

3.3.1.3. Histogram-Based Outlier Score

a) Phân tích kết quả tối ưu hóa tham số

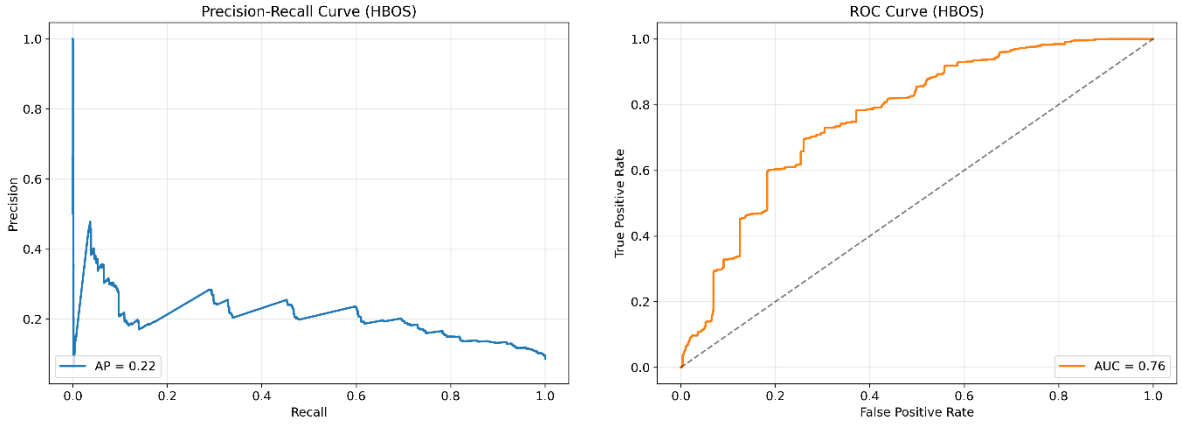
Quá trình tối ưu siêu tham số bằng Optuna đã tìm được cấu hình phù hợp với bản chất phân phối một chiều của HBOS – vốn giả định các đặc trưng độc lập và đánh giá điểm bất thường theo histogram. Bảng dưới đây trình bày chi tiết:

Bảng 3. 4: Cấu hình siêu tham số của HBOS

Tham số	Giá trị tối ưu	Phân tích
Số lượng bins (n_bins)	92	Bins lớn mô tả phân phối biên chi tiết hơn, hữu ích với đặc trưng heavy-tailed.
Hệ số smoothing (alpha)	0.2091	Làm trơn vừa phải, giảm nhiễu ở tail nhưng không che mất outlier thật.
Sử dụng PCA (use_pca)	True	Giảm tương quan giữa đặc trưng phù hợp với giả định độc lập của HBOS.
Số chiều PCA (n_components)	10	Giảm chiều giúp histogram hoạt động ổn định và đỡ overfit.

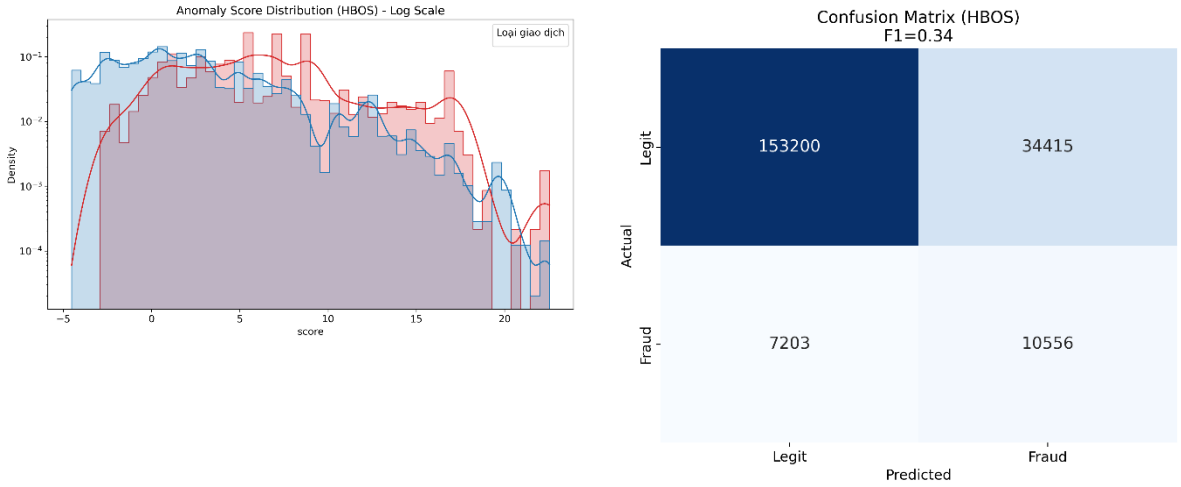
b) Phân tích hiệu năng trên tập kiểm tra

- PR-AUC, ROC-AUC, Precision@1%
  - PR-AUC = 0.2213, ROC-AUC = 0.7642, Precision@1% ≈ 0.3877.
  - Mô hình xếp hạng tốt hơn ngẫu nhiên và nhận diện được một phần các điểm outlier rõ rệt, nhưng vẫn kém các mô hình phức tạp hơn về chất lượng xếp hạng tổng thể.



Hình 3. 5: PR Curve và ROC Curve của HBOS

- Phân phối điểm bất thường
    - Hai lớp có tách biệt nhất định ở vùng điểm cao, nhưng chồng lấn nhiều ở vùng điểm trung bình.
    - Do chỉ mô hình hóa từng đặc trưng độc lập, HBOS dễ coi các giao dịch hợp lệ nhưng có giá trị cực trị là bất thường, dẫn đến số lượng cảnh báo giả lớn.
- Ma trận nhầm lẫn tại ngưỡng tối ưu (threshold ≈ 5.5533)
    - F1-score ≈ 0.3366, Precision ≈ 0.2347, Recall ≈ 0.5944.
    - TN = 153 200, FP = 34 415, FN = 7 203, TP = 10 556.
    - Mô hình phát hiện được khoảng 59% gian lận nhưng đồng thời sinh ra nhiều false positive, phản ánh trực tiếp hạn chế từ giả định độc lập giữa các đặc trưng.



Hình 3. 6: Phân phối điểm bất thường và ma trận nhầm lẫn của HBOS

c) Nhận xét

- Ưu điểm
    - Cấu trúc mô hình đơn giản, suy luận nhanh, dễ triển khai.

- ROC-AUC (≈ 0.76) và PR-AUC (≈ 0.22) ở mức chấp nhận được đối với một phương pháp histogram một chiều, đặc biệt khi có bước PCA hỗ trợ.

- Hạn chế
  - Precision và Precision@1% còn thấp, số cảnh báo giả lớn (FP ≈ 34 nghìn).
  - Không nắm bắt được quan hệ giữa các đặc trưng giao dịch, trong khi đây là yếu tố quan trọng trong phát hiện gian lận DeFi.

3.3.2. Mô hình học sâu

3.3.2.1. Dense Autoencoder

a) Phân tích kết quả tối ưu hóa tham số

Quá trình tối ưu siêu tham số bằng Optuna đã tìm ra cấu hình phản ánh đúng bản chất phi tuyến và phân phối phức tạp của dữ liệu giao dịch DeFi. Dense Autoencoder, vốn học biểu diễn nén thông qua tái tạo, đặc biệt nhạy với cách chọn kích thước tầng ẩn, hàm kích hoạt và mức regularization. Bảng dưới đây tổng hợp cấu hình tối ưu và ý nghĩa:

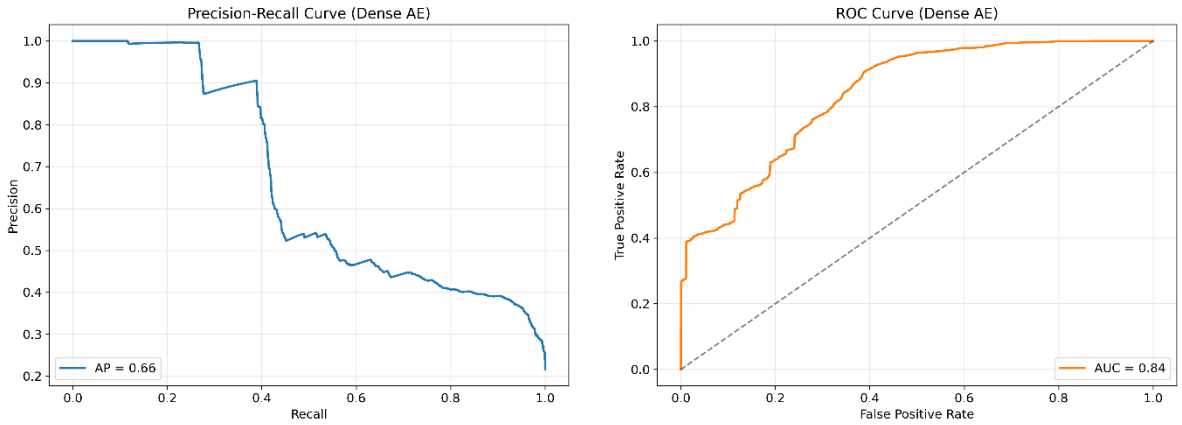
Bảng 3. 5: Cấu hình siêu tham số của DAE

Tham số	Giá trị tối ưu	Phân tích
Kích thước tầng 1 (w1)	512	Cho phép mạng học các biểu diễn bậc cao ở không gian đầu tiên, phù hợp với dữ liệu đa chiều và nhiễu.
Kích thước tầng 2 (w2)	256	Giảm dần chiều một cách mượt, tránh nghẽn cổ chai quá sớm gây mất thông tin.
Kích thước bottleneck (bott)	24	Biểu diễn nén xuống 24 chiều là vừa đủ để tách biệt hành vi hợp lệ – gian lận mà không làm mất cấu trúc quan trọng.

Dropout (drop)	≈ 0.111	Tạo nhiễu có kiểm soát giúp mô hình học robust hơn, hạn chế overfitting vào giao dịch hợp lệ.
Regularization L2 (l2)	6.6e-7	Mức phạt nhỏ, phù hợp vì reconstruction-based model cần tránh ràng buộc quá mạnh.
Learning Rate (lr)	≈ 1.8e-3	LR tương đối cao, giảm thời gian hội tụ và ổn định tốt với batch rất lớn.

b) Phân tích hiệu năng trên tập kiểm tra

- PR-AUC, ROC-AUC, Precision@1%
  - PR-AUC = 0.6574, ROC-AUC = 0.8395, Precision@1% = 1.0000.
  - Điều này cho thấy DAE xếp hạng và phân tách gian lận rất tốt, đặc biệt trong top 1% giao dịch có điểm bất thường cao nhất thì toàn bộ đều là gian lận thật, cực kỳ phù hợp cho bước ưu tiên kiểm tra thủ công.

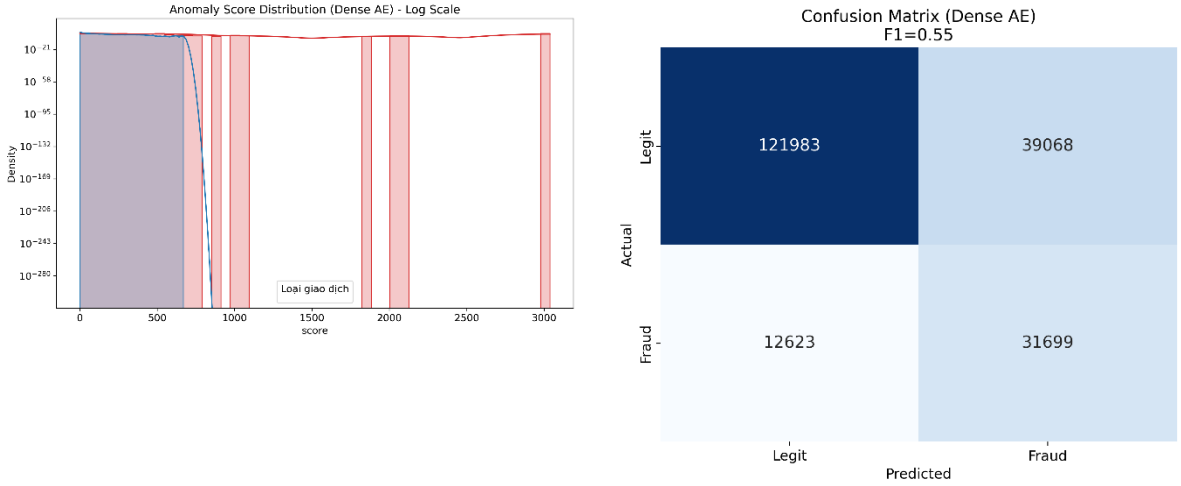


Hình 3. 7: PR Curve và ROC Curve của DAE

- Phân phối điểm bất thường
  - Giao dịch hợp lệ tập trung mạnh ở vùng điểm thấp (khoảng 0–800), trong khi giao dịch gian lận tạo thành một dải điểm rộng kéo dài tới hàng nghìn.

- Khoảng tách biệt lớn giữa hai phân phối phản ánh việc mô hình tái tạo tốt cấu trúc giao dịch hợp lệ, nhưng thất bại rõ rệt với các mẫu gian lận, đúng kỳ vọng của một mô hình autoencoder dựa trên lỗi tái tạo.

- Ma trận nhầm lẫn tại ngưỡng tối ưu (threshold ≈ 16.2069)
  - F1-score ≈ 0.5509, Precision ≈ 0.4479, Recall ≈ 0.7152.
  - TN = 121 983, FP = 39 068, FN = 12 623, TP = 31 699.
  - Mô hình giữ lại được phần lớn gian lận với F1 tương đối cao, nhưng vẫn tạo ra một lượng đáng kể cảnh báo giả do một số giao dịch hợp lệ hiếm cũng có lỗi tái tạo lớn.



Hình 3. 8: Phân phối điểm bất thường và ma trận nhầm lẫn của DAE

c) Nhận xét

- Ưu điểm
  - PR-AUC và ROC-AUC cao, thể hiện khả năng phân tách mạnh giữa giao dịch hợp lệ và gian lận trên dữ liệu tabular.
  - Precision@1% = 1.0000, giúp DAE đặc biệt hiệu quả cho việc khoanh vùng một tập rất nhỏ giao dịch rủi ro nhất để kiểm tra sâu.
  - Recall > 0.71, giảm nguy cơ bỏ sót gian lận trên tập kiểm tra.

- Hạn chế
    - Precision chung (≈ 0.45) và số FP (≈ 39 nghìn) cho thấy vẫn còn nhiều cảnh báo giả, nhất là với các giao dịch hợp lệ nhưng hiếm hoặc biến thiên mạnh.
    - Chi phí huấn luyện và suy luận cao hơn các mô hình học máy truyền thống.

3.3.2.2. Long Short-term Memory Autoencoder

a) Phân tích kết quả tối ưu hóa tham số

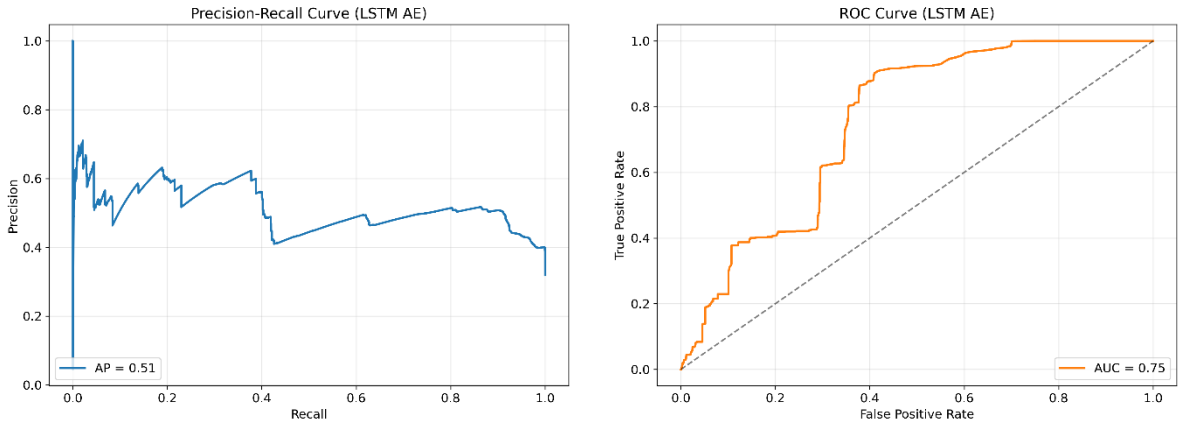
Quá trình tối ưu bằng Optuna đã lựa chọn một cấu hình phản ánh đúng đặc trưng chuỗi giao dịch và tính phụ thuộc thời gian của dữ liệu DeFi. Các siêu tham số tối ưu được tóm tắt trong bảng sau:

Bảng 3. 6: Cấu hình siêu tham số của LSTM-AE

Tham số	Giá trị tối ưu	Phân tích
units1	256	Đây là số đơn vị LSTM ở tầng encoder đầu tiên. Giá trị lớn giúp mô hình nắm bắt được các phụ thuộc dài hạn trong chuỗi giao dịch.
units2	128	Số đơn vị ở tầng encoder thứ hai nhỏ hơn units1, tạo cấu trúc nén dần đặc trưng thời gian trước khi vào latent.
drop	0.288	Dropout giúp giảm overfitting vì dữ liệu chuỗi thường có nhiễu và các pattern ngắn hạn không ổn định.
bidirectional	True	Sử dụng LSTM hai chiều cho phép mô hình học cả quan hệ trước và sau trong chuỗi — rất quan trọng trong phân tích hành vi giao dịch.

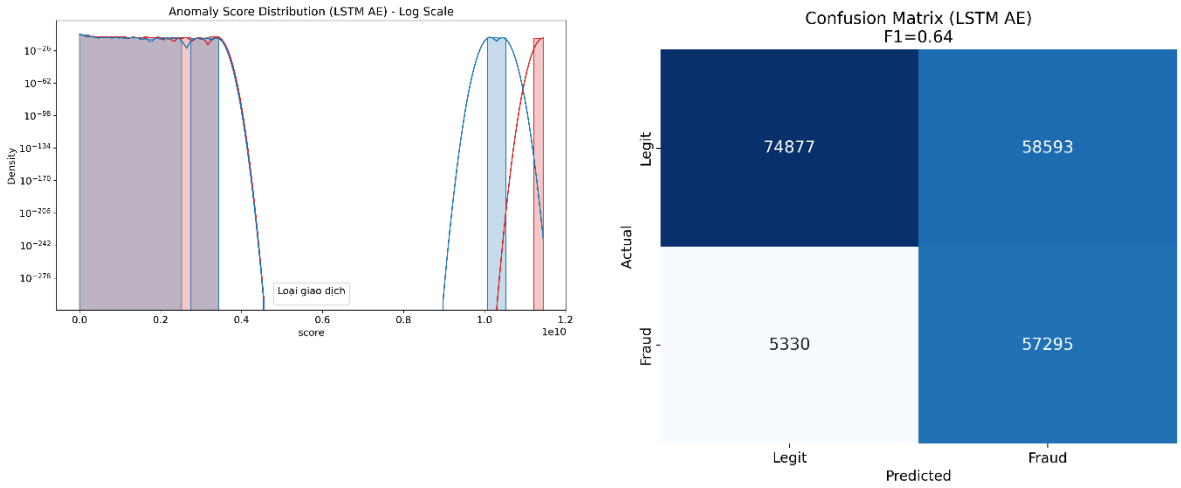
b) Phân tích hiệu năng trên tập kiểm tra

- PR-AUC, ROC-AUC, Precision@1%
    - PR-AUC = 0.5137, ROC-AUC = 0.7514, Precision@1% ≈ 0.6765.
    - Các chỉ số này cho thấy LSTM-AE xếp hạng gian lận trên dữ liệu chuỗi tốt hơn rõ rệt so với các mô hình thuần tabular, đặc biệt ở nhóm chuỗi có điểm bất thường cao nhất.



Hình 3. 9: PR Curve và ROC Curve của LSTM-AE

- Phân phối điểm bất thường
    - Chuỗi hợp lệ chủ yếu nằm ở vùng điểm thấp, trong khi chuỗi gian lận tập trung ở các giá trị rất cao (tới khoảng 1e10 do tổng residual theo chuỗi).
    - Điều này cho thấy lỗi tái tạo theo thời gian phản ánh tốt mức độ lệch chuẩn trong chuỗi hành vi giao dịch, dù vẫn có một số chuỗi hợp lệ hiếm mang điểm cao.
- Ma trận nhầm lẫn tại ngưỡng tối ưu (threshold ≈ 145.3797)
    - F1-score ≈ 0.6419, Precision ≈ 0.4944, Recall ≈ 0.9149.
    - TN = 74 877, FP = 58 593, FN = 5 330, TP = 57 295.
    - Mô hình phát hiện được hơn 91% chuỗi gian lận, ưu tiên độ phủ rất cao nhưng đi kèm số lượng false positive đáng kể.



Hình 3. 10: Phân phối điểm bất thường và ma trận nhầm lẫn của LSTM-AE

c) Nhận xét

- Ưu điểm
  - Recall rất cao (≈ 0.91), phù hợp với mục tiêu hạn chế bỏ sót các chuỗi gian lận.
  - F1 ≈ 0.64 với Precision gần 0.5 là mức cân bằng tốt trong bối cảnh dữ liệu chuỗi biến động mạnh.
  - Khai thác được động học thời gian mà các mô hình tabular không nắm bắt được.
- Hạn chế
  - Precision chung (≈ 0.49) và FP (≈ 58 nghìn) cho thấy tỷ lệ cảnh báo giả còn lớn, đặc biệt với các chuỗi hợp lệ nhưng hiếm gặp.
  - Biên độ điểm bất thường rất rộng (tới 1e10), đòi hỏi bước chuẩn hóa và diễn giải cẩn trọng khi triển khai.
  - Chi phí tính toán cho huấn luyện và suy luận cao hơn đáng kể so với mô hình tabular.

3.3.2.3. Transformer Variational Autoencoder

a) Phân tích kết quả tối ưu hóa tham số

Quá trình tối ưu siêu tham số bằng Optuna đã chọn được một cấu hình Transformer VAE khá gọn nhưng đủ khả năng mô hình hóa chuỗi giao dịch DeFi. Bảng dưới tóm tắt các tham số chính:

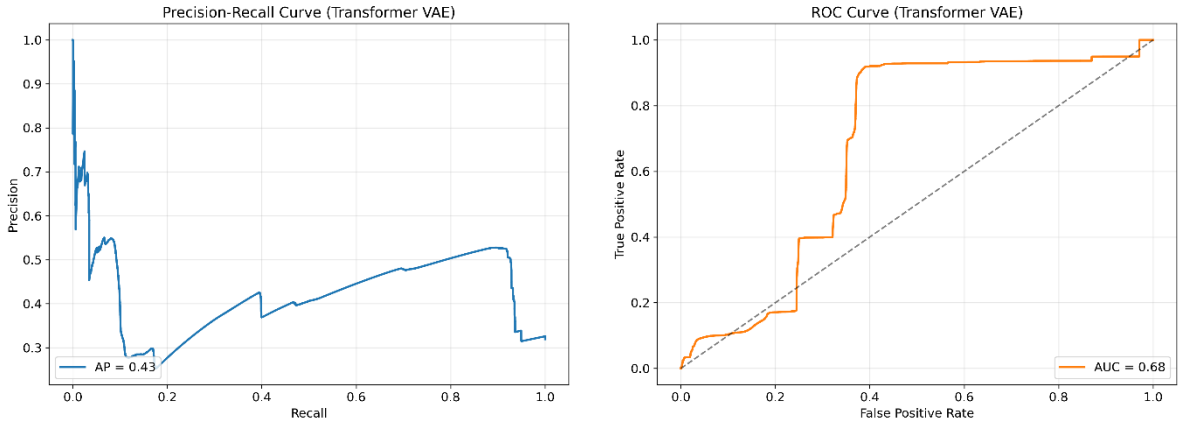
Bảng 3. 7: Cấu hình siêu tham số của Transformer-VAE

Tham số	Giá trị tối ưu	Phân tích
Kích thước embedding (embed_dim)	64	Mã hóa mỗi bước thời gian vào vector 64 chiều, đủ giàu thông tin nhưng vẫn giữ chi phí tính toán và nguy cơ overfitting ở mức chấp nhận được.
Số đầu attention (num_heads)	8	Multi-head self-attention với 8 heads cho phép mô hình học song song nhiều kiểu phụ thuộc khác nhau trong chuỗi.
Số block Transformer (num_blocks)	1	Cấu hình gọn, tránh dư thừa; thêm nhiều block không cải thiện rõ rệt mà chỉ tăng thời gian huấn luyện.
Tỷ lệ dropout (dropout)	≈ 0.33	Dropout khá cao giúp regularization mạnh, hạn chế overfitting khi số chiều latent lớn và phân phối giao dịch rất đa dạng.
Kích thước latent (latent)	24	Không gian latent 24 chiều cho phép VAE nén chuỗi giao dịch thành biểu diễn gọn nhưng vẫn giữ cấu trúc bất thường cần thiết để tính điểm bất thường.

b) Phân tích hiệu năng trên tập kiểm tra

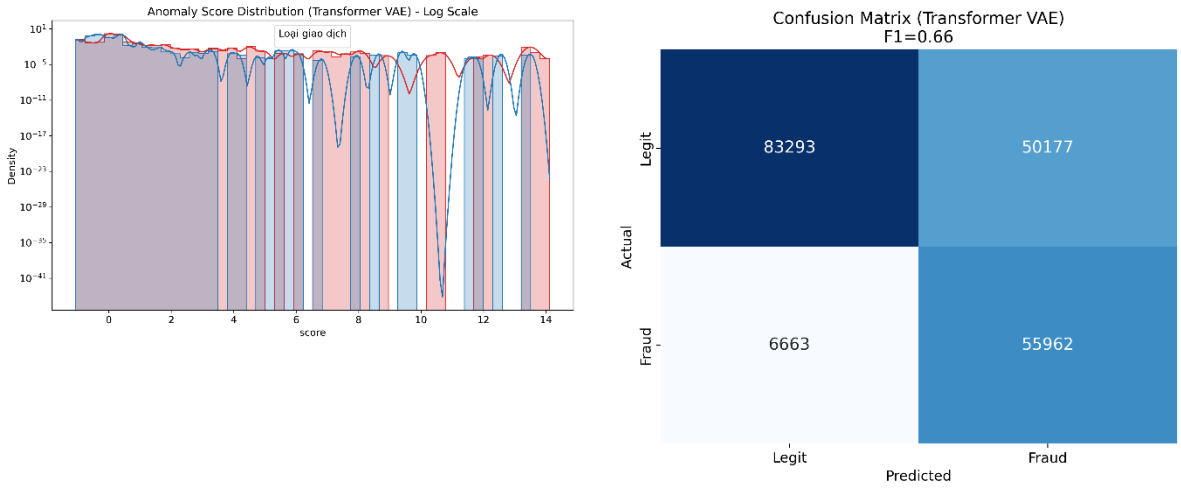
- PR-AUC, ROC-AUC, Precision@1%

- PR-AUC = 0.4258, ROC-AUC = 0.6768, Precision@1% ≈ 0.7332.
- PR-AUC ở mức khá trong bối cảnh dữ liệu chuỗi nhiều nhiễu, còn P@1% cho thấy mô hình ưu tiên tương đối chính xác nhóm 1% chuỗi rủi ro nhất, dù ROC-AUC chỉ ở mức trung bình.



Hình 3. 11: PR Curve và ROC Curve của Transformer-VAE

- Phân phối điểm bất thường
  - Phân phối điểm của chuỗi gian lận dịch rõ sang bên phải so với chuỗi hợp lệ, nhưng vẫn tồn tại vùng chồng lấn ở khoảng điểm trung bình.
  - Điều này cho thấy mô hình đã học được một số mẫu latent đặc trưng cho hành vi tấn công, song với các chuỗi gian lận tinh vi hoặc chuỗi hợp lệ hiếm, điểm bất thường đôi khi chưa đủ tách bạch.
- Ma trận nhầm lẫn tại ngưỡng tối ưu (threshold ≈ 0.0007)
  - F1-score ≈ 0.6632, Precision ≈ 0.5273, Recall ≈ 0.8936.
  - TN = 83 293, FP = 50 177, FN = 6 663, TP = 55 962.
  - Mô hình phát hiện gần 90% chuỗi gian lận với hơn một nửa cảnh báo là gian lận thật, nhưng số lượng false positive vẫn lớn do vùng chồng lấn phân phối.



Hình 3. 12: Phân phối điểm bất thường và ma trận nhầm lẫn của Transformer-VAE

c) Nhận xét

- Ưu điểm
  - F1 ≈ 0.66 và Recall ≈ 0.89 cho thấy Transformer VAE có khả năng bao phủ tốt các chuỗi gian lận.
  - Precision@1% ≈ 0.73, phù hợp để ưu tiên tương đối chính xác danh sách nhỏ các chuỗi nghi ngờ nhất.
  - Kiến trúc Transformer cho phép mô hình hóa tốt các phụ thuộc dài hạn và phức tạp trong chuỗi giao dịch.
- Hạn chế
  - ROC-AUC chỉ khoảng 0.68, khả năng phân tách tổng thể giữa hai lớp chưa thật sự nổi bật.
  - FP khoảng 50 nghìn, phản ánh vùng chồng lấn điểm bất thường giữa chuỗi hợp lệ hiếm và chuỗi gian lận tinh vi.
  - Kiến trúc phức tạp, yêu cầu tài nguyên tính toán và triển khai cao.

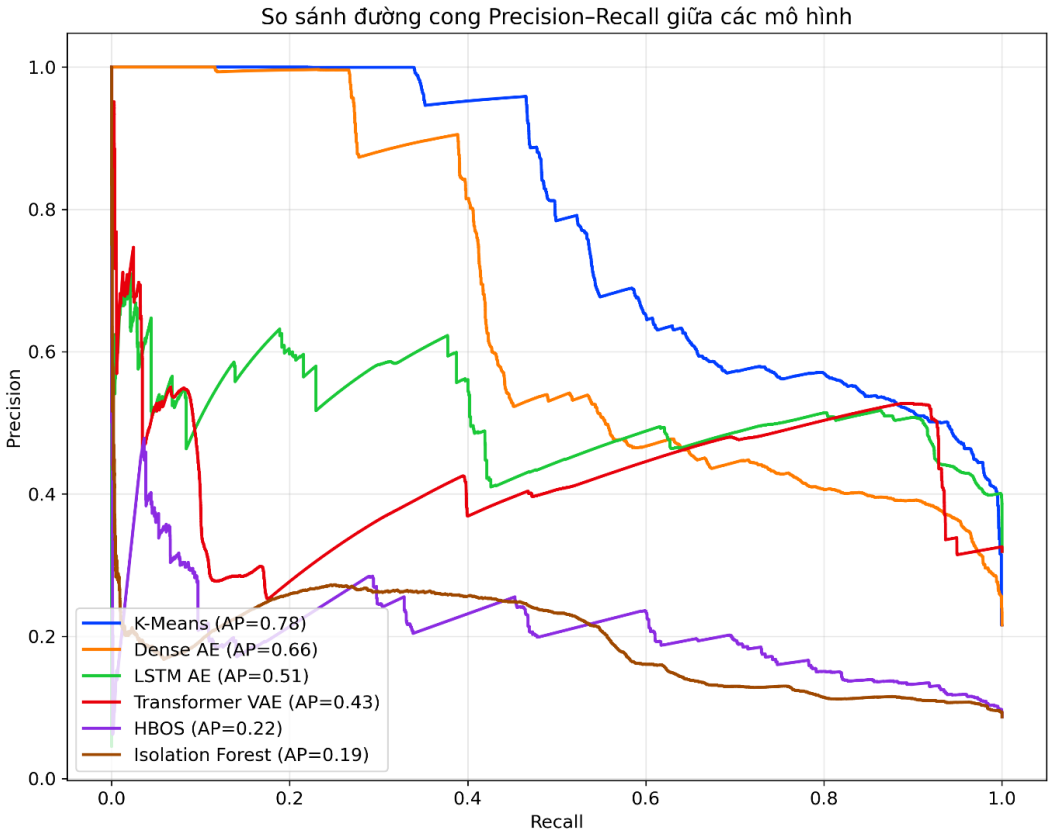
3.4. Đánh giá và so sánh kết quả mô hình

Trước khi đi vào phân tích chi tiết từng khía cạnh, bảng 3.8 tổng hợp các chỉ số đánh giá chính của sáu mô hình trên tập kiểm tra, bao gồm PR-AUC, ROC-AUC, Precision@1%, Precision, Recall, F1-score và các phần tử của ma trận nhầm lẫn (TN, FP, FN, TP) tại ngưỡng tối ưu. Các mục 3.4.1–3.4.4 phía sau sẽ lần lượt phân tích sâu hơn từng nhóm chỉ số này.

Bảng 3. 8: Tổng hợp hiệu năng chi tiết của các mô hình

Mô hình	PR-AUC	ROC-AUC	P@1%	Precision	Recall	F1-Score	TP	FP	FN	TN
K-Means	**0.7801**	**0.9146**	**1.0000**	0.5707	0.7987	**0.6657**	35,399	26,627	8,923	134,424
DAE	0.6574	0.8395	**1.0000**	0.4479	0.7152	0.5509	31,699	39,068	12,623	121,983
LSTM-AE	0.5137	0.7514	0.6765	0.4944	**0.9149**	0.6419	**57,295**	58,593	**5,330**	74,877
Trans-VAE	0.4258	0.6768	0.7332	0.5273	0.8936	0.6632	55,962	50,177	6,663	83,293
HBOS	0.2213	0.7642	0.3877	0.2347	0.5944	0.3366	10,556	34,415	7,203	153,200
Isolation Forest	0.1902	0.7168	0.2065	0.2464	0.4623	0.3215	8,210	25,106	9,549	**162,509**

3.4.1. Chất lượng xếp hạng



Hình 3. 13: So sánh đường cong Precision–Recall giữa các mô hình

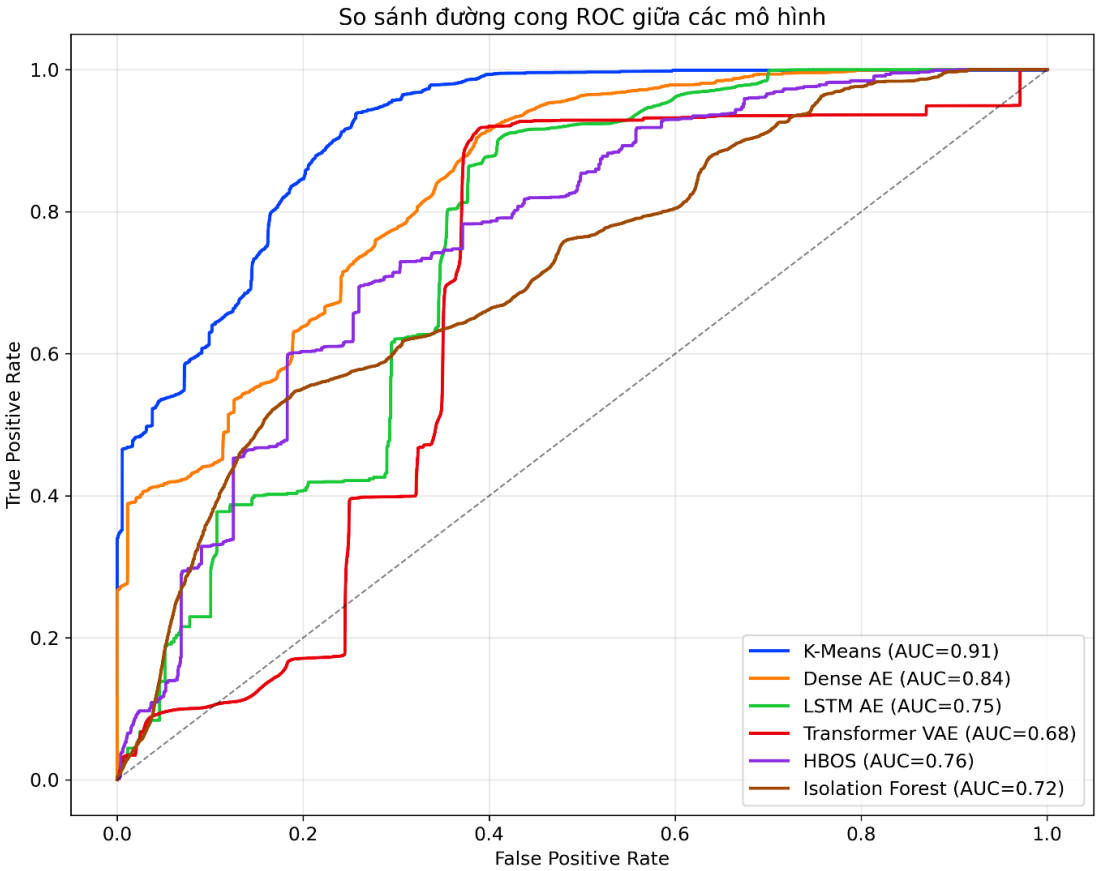
Chỉ số PR-AUC phản ánh khả năng xếp hạng đúng các giao dịch gian lận trong bối cảnh dữ liệu mất cân bằng mạnh, vốn là đặc trưng phổ biến của dữ liệu DeFi. Giá trị PR-AUC cao đồng nghĩa với việc mô hình có thể đưa phần lớn giao dịch gian lận lên các vị trí ưu tiên cao, điều đặc biệt quan trọng trong các hệ thống cảnh báo sớm.

Kết quả cho thấy K-Means (PR-AUC ≈ 0,78) và Dense Autoencoder – DAE (PR-AUC ≈ 0,66) là hai mô hình vượt trội nhất. Điều này cho thấy, trong không gian đặc trưng đã được tiền xử lý và giảm chiều, khoảng cách đến tâm cụm và lỗi tái tạo là hai tín hiệu rất mạnh để nhận diện hành vi bất thường.

Các mô hình chuỗi đạt kết quả trung bình: LSTM-AE (≈ 0,51) và Transformer-VAE (≈ 0,43). Dù PR-AUC chưa cao bằng các mô hình tabular, kết quả này phản ánh rằng thông tin động học theo thời gian giúp mô hình phát hiện được các gian lận kéo dài, nhưng cũng làm tăng độ chồng lấn điểm giữa hai lớp.

Ngược lại, HBOS và Isolation Forest có PR-AUC thấp (≈ 0,22 và ≈ 0,19), cho thấy các phương pháp dựa trên giả định thống kê đơn giản hoặc phân tách ngẫu nhiên khó nắm bắt được cấu trúc phức tạp, đa chiều và nhiều nhiễu của giao dịch DeFi.

3.4.2. Khả năng phân tách tổng quát



Hình 3. 14: So sánh đường cong ROC giữa các mô hình

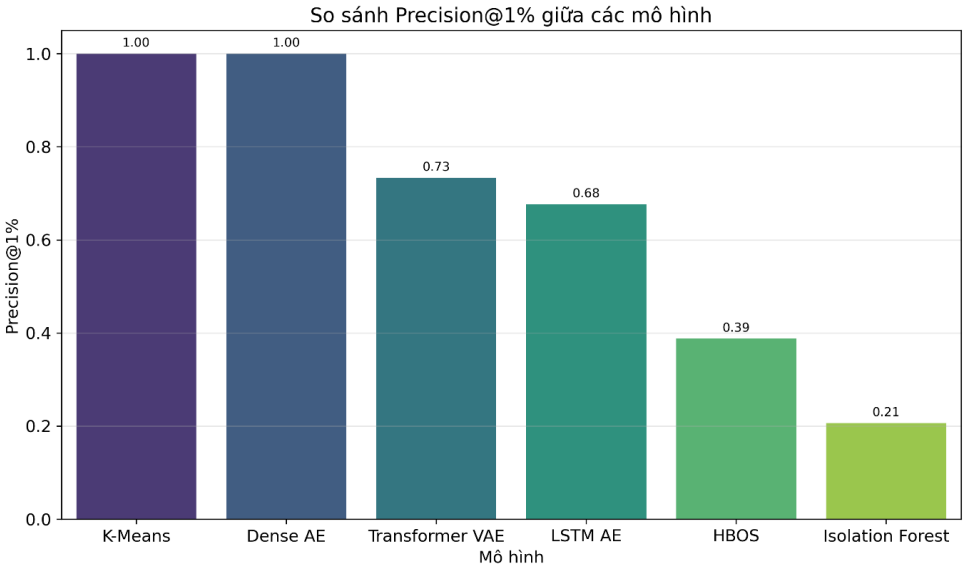
ROC-AUC đánh giá năng lực phân biệt hai lớp trên toàn bộ miền ngưỡng, không tập trung riêng vào lớp thiểu số. Do đó, chỉ số này phản ánh khả năng học cấu trúc tổng quát của dữ liệu.

K-Means đạt ROC-AUC cao nhất (≈ 0,91), cho thấy việc phân cụm trong không gian PCA đã tạo ra biên phân tách rất rõ giữa giao dịch hợp lệ và gian lận. Dense Autoencoder (≈ 0,84) xếp thứ hai, xác nhận khả năng học biểu diễn phi tuyến hiệu quả.

Nhóm mô hình chuỗi đạt ROC-AUC thấp hơn (LSTM-AE ≈ 0,75, Transformer-VAE ≈ 0,68), phản ánh sự đánh đổi giữa việc học phụ thuộc thời gian và khả năng phân tách rạch ròi hai lớp.

Đáng chú ý, HBOS đạt ROC-AUC ≈ 0,76, tương đương LSTM-AE, cho thấy mô hình này vẫn có khả năng phân tách tổng quát khá tốt, dù hiệu quả xếp hạng ở vùng gian lận lại hạn chế. Isolation Forest (≈ 0,72) cho kết quả khiêm tốn, phù hợp với nhận định rằng mô hình này khó thích nghi với dữ liệu có cấu trúc phức tạp.

3.4.3. Hiệu quả tại vùng cảnh báo trọng điểm



Hình 3. 15: So sánh Precision@1% giữa các mô hình

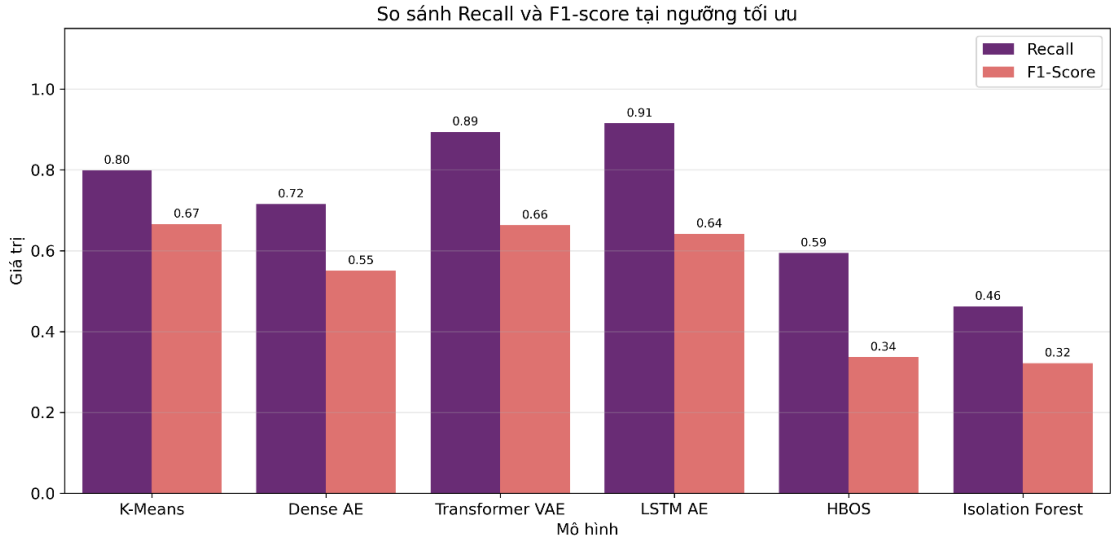
Trong triển khai thực tế, hệ thống thường chỉ có thể xử lý hoặc điều tra một tỷ lệ nhỏ các giao dịch có điểm bất thường cao nhất. Do đó, Precision@1% là chỉ số phản ánh trực tiếp giá trị ứng dụng của mô hình.

Kết quả cho thấy K-Means và DAE đạt P@1% = 1,0, tức toàn bộ giao dịch nằm trong top 1% nguy hiểm nhất đều là gian lận. Đây là kết quả đặc biệt quan trọng, chứng minh hai mô hình này có thể đóng vai trò bộ lọc tuyến đầu với độ tin cậy rất cao.

Transformer-VAE (≈ 0,73) và LSTM-AE (≈ 0,68) đạt mức khá, cho thấy các mô hình chuỗi có khả năng tập trung vào các hành vi bất thường mạnh, nhưng vẫn tồn tại một số chuỗi hợp lệ hiếm bị xếp nhầm vào top.

Trong khi đó, HBOS và Isolation Forest có Precision@1% thấp (< 0,4), cho thấy hai mô hình này khó xác định chính xác vùng rủi ro cao, hạn chế khả năng sử dụng làm bộ cảnh báo ưu tiên.

3.4.4. Năng lực bao phủ và cân bằng phát hiện



Hình 3. 16: So sánh Recall và F1-score tại ngưỡng tối ưu

3.4.4.1. So sánh Recall

LSTM-AE đạt Recall ≈ 0.915, cao nhất trong toàn bộ sáu mô hình: Điều này cho thấy mô hình chuỗi này có khả năng bao phủ gian lận rất tốt, ít bỏ sót, phù hợp cho các hệ thống ưu tiên độ an toàn cao.

Transformer-VAE đạt Recall ≈ 0.894, gần tương đương LSTM-AE nhưng với phân phối điểm tập trung hơn: Mô hình học sâu dựa trên attention tuy chưa đạt độ sắc sảo cao nhất ở top-tier, nhưng lại rất mạnh trong việc gom được phần lớn gian lận vào dải điểm cao.

Dense Autoencoder đạt Recall ≈ 0.715: Tuy không đạt mức bao phủ như mô hình chuỗi, DAE vẫn phát hiện được phần lớn gian lận mà không cần cấu trúc thời gian, chứng minh tính mạnh mẽ của mô hình tái tạo phi tuyến trên dữ liệu tabular.

K-Means đạt Recall ≈ 0.799, thuộc nhóm khá mạnh: Đây là mức hiệu quả tốt với mô hình phân cụm, đặc biệt khi không sử dụng thông tin thời gian và không được huấn luyện phân biệt hai lớp.

HBOS đạt Recall ≈ 0.594: mô hình histogram một chiều chỉ nhận diện tốt các outlier rõ rệt nhưng bỏ sót nhiều giao dịch gian lận có cấu trúc phức tạp.

Isolation Forest chỉ đạt Recall ≈ 0.462, thấp nhất trong nhóm: Phản ánh sự hạn chế của thuật toán khi phân phối điểm của lớp hợp lệ và lớp gian lận bị chồng lấn quá lớn.

3.4.4.2. So sánh F1-score

K-Means và Transformer-VAE là hai mô hình có F1 cao nhất (lần lượt ≈ 0.666 và ≈ 0.663), trong đó K-Means nhỉnh hơn nhẹ dù là mô hình phân cụm đơn giản.

LSTM-AE đạt F1 ≈ 0.642: phù hợp cho kịch bản ưu tiên hạn chế bỏ sót, dù Precision chưa cao.

DAE đạt F1 ≈ 0.551: Precision thấp kéo F1 xuống, dù Recall vẫn khá tốt. Mô hình phù hợp cho việc lọc ưu tiên hơn là tối ưu F1.

HBOS và Isolation Forest đều có F1 thấp (≈ 0.33 và ≈ 0.32): cho thấy hiệu năng hạn chế trong cả bao phủ lẫn độ chính xác.

3.4.5. Đánh giá hiệu năng tính toán và khả năng triển khai thực tế

Để đánh giá tính khả thi khi triển khai trong môi trường sản xuất (production), luận văn tiến hành đo lường ba nhóm chỉ số hiệu năng cốt lõi cho sáu mô hình: HBOS, Isolation Forest, K-Means, Dense Autoencoder (DAE), LSTM Autoencoder và Transformer-VAE. Các thực nghiệm được thực hiện thống nhất trên cùng một cấu hình phần cứng (như đã trình bày ở phần 3.2.1.) với dữ liệu đã tiền xử lý ở chương 3. Kết quả được tóm tắt trong bảng 3.9 như sau:

Bảng 3. 9: Tổng hợp hiệu năng tính toán của các mô hình

Mô hình	**Độ trễ đơn lẻ (ms/mẫu)**	**Thông lượng (mẫu/giây)**	**Thời gian xử lý Batch (s)**	**Kích thước (MB)**
HBOS	**0,29**	**1.277.705**	0,039	9,42
K-Means	0,53	262.249	0,192	2,39
Isolation Forest	25,07	25.774	1,941	**1.448,07**
DAE	119,42	264.592	0,193	3,63
LSTM-AE	131,00	15.496	0,653	32,40
Trans-VAE	120,00	20.076	0,500	**0,59**

3.4.5.1. Độ trễ suy luận đơn lẻ

Chỉ số này mô phỏng kịch bản xử lý thời gian thực, nơi mỗi giao dịch mới phát sinh cần được đánh giá ngay lập tức.

- Nhóm siêu tốc (Ultra-low Latency): HBOS và K-Means đạt hiệu năng vượt trội với độ trễ dưới 1 ms (0,29 ms và 0,53 ms). Đây là mức gần như không đáng kể,

đáp ứng tốt yêu cầu của hệ thống chặn gian lận thời gian thực (thường yêu cầu < 50 ms). Cả hai mô hình đều hoạt động hiệu quả trên CPU mà không cần GPU.

- Nhóm trung bình: Isolation Forest đạt khoảng 25 ms/mẫu. Mức này vẫn dùng được cho hệ thống thời gian thực, nhưng biên an toàn thấp. Khi cộng thêm độ trễ mạng và truy vấn cơ sở dữ liệu, tổng thời gian phản hồi dễ vượt ngưỡng 50 ms.
- Nhóm độ trễ cao: DAE, LSTM-AE và Transformer-VAE đều có độ trễ > 100 ms do chi phí tính toán lớn khi batch size = 1. Nhóm này không phù hợp cho xử lý chặn trực tiếp.

3.4.5.2. Thông lượng xử lý

Chỉ số này đánh giá khả năng xử lý khối lượng dữ liệu lớn, phục vụ cho các tác vụ phân tích định kỳ hoặc hậu kiểm.

- HBOS dẫn đầu tuyệt đối với ≈ 1,28 triệu mẫu/giây, cho phép quét toàn bộ dữ liệu lịch sử trong thời gian rất ngắn.
- Dense Autoencoder đạt ≈ 264.000 mẫu/giây — cho thấy mô hình tận dụng GPU cực tốt khi suy luận theo batch, thậm chí nhanh hơn hẳn Isolation Forest.
- K-Means đạt ≈ 262.000 mẫu/giây, tương đương DAE.
- Isolation Forest thấp nhất (≈ 25.000 mẫu/giây) do phải duyệt nhiều cây quyết định tuần tự, khó song song hóa trên GPU.
- Với mô hình chuỗi:
    - Transformer-VAE: ≈ 20.000 chuỗi/giây,
    - LSTM-AE: ≈ 15.500 chuỗi/giây.

  Thông lượng này là tốt đối với dữ liệu chuỗi 20 bước/giao dịch.

3.4.5.3. Dung lượng mô hình và yêu cầu bộ nhớ

- Nhẹ và tối ưu:

- Transformer-VAE gây ấn tượng khi chỉ chiếm 0,59 MB — rất nhỏ cho một mô hình xử lý chuỗi có encoder–decoder và tái tham số hóa.
- DAE và K-Means cũng nhẹ (< 4 MB), phù hợp để triển khai trên container hoặc thiết bị biên.

- Kích thước lớn: Isolation Forest có dung lượng tới 1,45 GB, gây tốn bộ nhớ đĩa lẫn RAM khi nạp mô hình (cold-start). Điều này bất lợi trong môi trường microservice có yêu cầu scale nhanh hoặc serverless.

3.4.5.4. Kết luận khả năng triển khai

Từ các phân tích đa chiều trên, luận văn đề xuất kiến trúc triển khai đa tầng để tận dụng ưu điểm của từng nhóm mô hình:

- Tầng lọc nhanh (Real-time Layer): HBOS hoặc K-Means nên được dùng ở tuyến đầu để kiểm tra 100% giao dịch. Với độ trễ < 1 ms và thông lượng cao, hai mô hình này giữ vai trò chặn ban đầu.
- Tầng phân tích sâu (Near Real-time/Offline Layer): Các giao dịch nghi ngờ được chuyển sang bộ phân tích chuyên sâu gồm Transformer-VAE hoặc LSTM-AE. Nhóm này có độ chính xác cao và thích hợp để phát hiện các hành vi bất thường tinh vi mà mô hình tuyến đầu có thể bỏ sót.
- Hạn chế sử dụng Isolation Forest: Mặc dù hiệu quả ở một số kịch bản, kích thước lớn và thông lượng thấp khiến mô hình này kém hấp dẫn khi so sánh với HBOS (nhanh, nhẹ) và Autoencoder (chính xác cao).

3.5. Kết luận chương

Chương 3 đã trình bày quy trình thực nghiệm áp dụng cho sáu mô hình phát hiện gian lận trên bộ dữ liệu BCCC-DeFiFraudTrans-2025, bao gồm tối ưu siêu tham số, huấn luyện và đánh giá theo bộ thước đo đã thiết kế. Mục tiêu là so sánh tương quan giữa các

mô hình và xác định vai trò phù hợp của từng nhóm trong bài toán phát hiện bất thường trên DeFi.

Kết quả cho thấy nhóm mô hình tabular (đặc biệt là K-Means và Dense Autoencoder) đạt hiệu năng nổi bật về chất lượng xếp hạng, với PR-AUC, ROC-AUC và Precision@1% ở mức cao, phù hợp đóng vai trò lớp lọc ưu tiên để khoanh vùng tập giao dịch rủi ro nhất. Ngược lại, nhóm mô hình chuỗi (LSTM-AE, Transformer-VAE) tỏ ra ưu thế về khả năng bao phủ, đạt Recall cao nhờ khai thác động học theo chuỗi giao dịch, thích hợp làm lớp bổ trợ nhằm giảm bỏ sót gian lận. HBOS và Isolation Forest chủ yếu giữ vai trò baseline tham chiếu, phản ánh hạn chế của các phương pháp dựa trên giả định phân phối đơn giản trong bối cảnh dữ liệu DeFi nhiều nhiễu và đa chiều.

Từ các quan sát này, chương 3 cho thấy không có một mô hình đơn lẻ nào tối ưu cho mọi tiêu chí, mà cấu hình hợp lý hơn là tiếp cận phân tầng: lớp tabular ưu tiên xếp hạng và lớp chuỗi tăng độ bao phủ. Đây là cơ sở để chương kết luận đề xuất kiến trúc hệ thống phát hiện gian lận tổng thể và các hướng mở rộng trong triển khai thực tế.

KẾT LUẬN VÀ KIẾN NGHỊ

Luận văn đã tiến hành nghiên cứu toàn diện bài toán phát hiện bất thường trong giao dịch DeFi dựa trên các mô hình học máy và học sâu, kết hợp bộ dữ liệu BCCC-DeFiFraudTrans-2025 với quy mô lớn, cấu trúc phức tạp và phân phối lệch mạnh theo thời gian. Trên cơ sở quy trình tiền xử lý được thiết kế riêng cho hai dạng dữ liệu bảng và chuỗi, luận văn đã triển khai, tối ưu và đánh giá sáu mô hình đại diện gồm K-Means, Isolation Forest, HBOS, Dense Autoencoder, LSTM-AE và Transformer-VAE. Hệ thống thước đo đa chiều (PR-AUC, ROC-AUC, Precision@1%, F1 và Recall) được sử dụng nhằm phản ánh đầy đủ khả năng xếp hạng, mức độ bao phủ và độ tin cậy cảnh báo của từng mô hình trong bối cảnh dữ liệu có tỉ lệ gian lận thấp và nhiễu cao.

Kết quả thực nghiệm cho thấy các mô hình tabular và mô hình chuỗi thể hiện thế mạnh ở những khía cạnh khác nhau. K-Means và Dense Autoencoder là hai phương pháp nổi bật nhất trong nhiệm vụ xếp hạng ưu tiên, khi đều đạt Precision@1% ở mức tuyệt đối và khả năng phân tách tổng quát tốt. Ngược lại, LSTM-AE và Transformer-VAE cho thấy hiệu quả vượt trội về mức độ bao phủ, với Recall xấp xỉ hoặc trên 0.9, phù hợp cho việc phát hiện các hành vi gian lận diễn tiến theo thời gian. Hai mô hình HBOS và Isolation Forest thể hiện năng lực hạn chế và chủ yếu phù hợp cho mục đích tham chiếu hoặc so sánh cơ sở. Từ sự phân hóa hiệu năng này có thể thấy rằng không tồn tại một mô hình duy nhất tối ưu cho mọi tiêu chí, và cách tiếp cận phân tầng giữa mô hình tabular (ưu tiên xếp hạng) và mô hình chuỗi (ưu tiên độ phủ) mang lại giá trị thực tiễn cao hơn khi triển khai trong các hệ thống giám sát DeFi.

Bên cạnh các kết quả đạt được, luận văn cũng đưa ra những kiến nghị cho hướng phát triển tiếp theo. Một mặt, hệ thống phát hiện gian lận có thể được nâng cao bằng việc mở rộng dữ liệu theo thời gian, bổ sung dữ liệu nguồn giá, thanh khoản hoặc thông tin cross-chain nhằm mô tả đầy đủ hơn bối cảnh giao dịch. Mặt khác, các mô hình hiện đại như mạng nơ-ron đồ thị (GNNs) hoặc các kiến trúc Transformer dành cho chuỗi dài có

tiềm năng cải thiện khả năng học quan hệ và phát hiện các hành vi tấn công đa bước. Việc kết hợp các mô hình học máy với luật nghiệp vụ, cũng như xây dựng cơ chế phản hồi từ phía người phân tích, có thể giảm thiểu cảnh báo giả và duy trì hiệu năng khi hành vi gian lận thay đổi theo thời gian. Cuối cùng, quá trình triển khai thực tế cần được hỗ trợ bởi cơ chế giải thích mô hình và kiểm đánh định kỳ để đảm bảo hệ thống duy trì độ tin cậy lâu dài.

Nhìn chung, luận văn đã cung cấp một đánh giá thực nghiệm có hệ thống, làm rõ đặc điểm vận hành của các mô hình trong bối cảnh DeFi và đề xuất hướng triển khai có tính thực tiễn cao. Các kết quả thu được không chỉ góp phần củng cố hiểu biết về phát hiện bất thường trên dữ liệu Blockchain, mà còn tạo nền tảng cho các nghiên cứu và ứng dụng tiếp theo trong lĩnh vực tài chính phi tập trung.

DANH MỤC TÀI LIỆU THAM KHẢO

Tiếng Anh

[1] Adeyemo, A. (2025), "Anomaly Detection in Cryptocurrency and Blockchain Transactions Using AI," ResearchGate Preprint, pp. 1–15.

[2] Buterin, V., Conner, E., Dudley, R., Slipper, M., Norden, I., Bakhta, A. (2019), "EIP-1559: Fee market change for ETH 1.0 chain," Ethereum Improvement Proposals, no. 1559.

[3] Chandola, V., Banerjee, A., Kumar, V. (2009), "Anomaly detection: A survey," ACM Computing Surveys 41(3), pp. 1–58.

[4] Chen, T., Guestrin, C. (2016), "XGBoost: A Scalable Tree Boosting System," Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining 22, pp. 785–794.

[5] Chithanuru, V., Ramaiah, M. (2023), "An anomaly detection on blockchain infrastructure using artificial intelligence techniques: Challenges and future directions – A review," Concurrency and Computation: Practice and Experience 35(22), e7724.

[6] Cholevas, C., Angeli, E., Sereti, Z., Mavrikos, E., Tsekouras, G. E. (2024), "Anomaly Detection in Blockchain Networks Using Unsupervised Learning: A Survey," Algorithms.

[7] Daian, P., et al. (2020), "Flash Boys 2.0: Frontrunning, Transaction Reordering, and Consensus Instability in Decentralized Exchanges," IEEE Symposium on Security and Privacy (SP), pp. 910–927.

[8] Goldstein, M., Dengel, A. (2012), "Histogram-based Outlier Score (HBOS): A fast unsupervised anomaly detection algorithm," KI-2012: Poster and Demo Track, pp. 59–63.

[9] Goodfellow, I., Bengio, Y., Courville, A. (2016), Deep Learning, MIT Press, Cambridge, MA, USA.

[10] Hochreiter, S., Schmidhuber, J. (1997), “Long short-term memory,” Neural Computation 9(8), pp. 1735–1780.

[11] Hussain, W., Hussain, T., Zeadally, S. (2023), “Anomaly Detection in Blockchain Networks: A Comprehensive Survey,” IEEE Communications Surveys & Tutorials 25(1), pp. 289–318.

[12] Jumani, F., Raza, M. (2025), “Machine Learning for Anomaly Detection in Blockchain: A Critical Analysis, Empirical Validation, and Future Outlook,” Computers 14(7), 247.

[13] Kim, J., Lee, S., Hong, J. (2021), “Anomaly Detection based on Traffic Monitoring for Secure Blockchain Networking,” IEEE International Conference on Blockchain and Cryptocurrency (ICBC), pp. 1–3.

[14] Kwon, D., Kim, H., Kim, J., Suh, S. C., Kim, I., Kim, K. J. (2019), “A Survey of Deep Learning-based Network Anomaly Detection,” Cluster Computing 22, pp. 949–961.

[15] LeCun, Y., Bengio, Y., Hinton, G. (2015), “Deep learning,” Nature 521, pp. 436–444.

[16] Lee, S., Kim, H., Kim, J. (2023), “Anomaly VAE-Transformer: A Deep Learning Approach for Anomaly Detection in Decentralized Finance,” IEEE Access 11, pp. 98115–98131.

[17] Liu, F. T., Ting, K. M., Zhou, Z.-H. (2008), “Isolation Forest,” Eighth IEEE International Conference on Data Mining, pp. 413–422.

[18] MacQueen, J. (1967), “Some methods for classification and analysis of multivariate observations,” Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability 1, pp. 281–297.

[19] Martin, K., Rahouti, M., Ayyash, M., Alsmadi, I. (2021), “Anomaly Detection in Crypto Currency,” Security and Privacy 4(6), e192.

[20] Monrat, A. A., Schelén, O., Andersson, K. (2019), “A Survey of Blockchain from the Perspectives of Applications, Challenges, and Opportunities,” IEEE Access 7, pp. 117134–117151.

[21] Nassar, M., Al Bouna, B., Malli, M. (2020), “Deep Learning for Anomaly Detection in Blockchain: A Review,” Journal of Information Security and Applications.

[22] Nofer, M., Gomber, P., Hinz, O., Schiereck, D. (2017), “Blockchain,” Business & Information Systems Engineering 59(3), pp. 183–187.

[23] Pang, G., Shen, C., Cao, L., van den Hengel, A. (2021), “Deep Learning for Anomaly Detection: A Review,” ACM Computing Surveys 54(2), pp. 1–38.

[24] Podgorelec, B., Turkanović, M., Karakatič, S. (2020), “A Novel Anomaly Detection Method for Blockchain-Based Systems,” International Conference on Information Intelligence Systems and Applications.

[25] Qin, K., Zhou, L., Livshits, B., Gervais, A. (2021), “Attacking the DeFi Ecosystem with Flash Loans for Fun and Profit,” Financial Cryptography and Data Security, pp. 3–32.

[26] Shafiq, O. (2023), Anomaly Detection in Blockchain, Master’s Thesis, Tampere University, Tampere, Finland.

[27] Shevchuk, R., Martsenyuk, V., Adamyk, B., Benson, V., Melnyk, A. (2025), "Anomaly Detection in Blockchain: A Systematic Review of Trends, Challenges, and Future Directions," Applied Sciences 15(15), 8330.

[28] Siddamsetti, S., Tejaswi, C., Maddula, P. (2024), "Anomaly Detection in Blockchain Using Machine Learning," Journal of Electrical Systems.

[29] Signorini, M., Pontecorvi, M., Kanoun, W., Di Pietro, R. (2020), "BAD: A Blockchain Anomaly Detection Solution," IEEE INFOCOM 2020 – IEEE Conference on Computer Communications Workshops, pp. 1–6.

[30] Vaswani, A., et al. (2017), "Attention is all you need," Advances in Neural Information Processing Systems (NIPS), pp. 5998–6008.

[31] Wang, J., Wu, P., Wang, X., Shou, W. (2021), "The Outlook of Blockchain Technology for Construction Engineering Management," Frontiers of Engineering Management.

[32] Wood, G. (2014), "Ethereum: A secure decentralised generalised transaction ledger," Ethereum Project Yellow Paper 151, pp. 1–32.