Fraud Detection in E-Commerce and Financial Transactions Using Graph Neural Networks

Karunakar Thammadi
Data Science
Anderson College of Business and Computing
Regis University, Denver, CO, USA
Kthammadi@regis.edu

Abstract

Fraud in internet business and financial systems continues to increase. Current systems consider transactions individually, but this does not look the way fraudsters actually work, because they are involved in webs of accounts that are fake and move money around them to cover their activity. Graph Neural Networks will be my choice of models to detect fraud based on the need to capture these relationships. The project uses three publicly available datasets (PaySim) mobile payment simulator, Elliptic Bitcoin transaction network and UCI credit card data). I will use a full workflow, which will involve cleaning and building of graphs and model training. The accounts will be taken as nodes and the transactions as edges. Conventional models which include XGBoost will be employed as the baseline models to compare with GNNs. The evaluation will consider recall and F1-score since the challenge of detecting fraud is more important than the general accuracy.

I. Introduction/Background

Fraud detection is becoming more difficult. According to Nilson Report, credit card fraud alone will cost business over 28.65 billion in 2023 and the figure keeps rising. Most companies utilize the system that analyzes transactions one by one. This is in simple cases but nowadays fraudsters utilize the networks of fake accounts to transfer money and hide their activity.

This project proposes building a fraud detection system based on Graph Neural Networks. The concept is easy: Relationships between accounts and transactions are no longer ignored, but directly modeled. When account A transacts money to B and the latter transacts the money to C, that is a pattern that could be explored. This project is challenging for several reasons. To begin with, fraud makes up less than 1% of transactions in most data sets and as a result, each modeling choice must carefully handle this imbalance. Second, no canned graph format, I will need to construct directed graphs out of transaction logs, and this requires heavy data engineering.

Overall, this project combines data engineering, advanced modeling, and handling imbalanced data.

II. Problem Statement

I'm trying to answer a straightforward question: when it comes to spotting fraud in e-commerce and financial transactions, do graph neural networks actually help? And if so, when?

The problem breaks down into several pieces. First, there's the data collection - finding public transaction datasets that are realistic enough to be useful. Then cleaning. After that, I need to engineer features both for traditional models and for graph construction. Building the graph is not straightforward: deciding what becomes a node (accounts? transactions? users?), what becomes an edge (money flow? other attributes?), and handling the directionality.

For modeling, I'm comparing two paths. Path one: traditional tabular methods like XGBoost that ignore relationships. Path two: GNN and other graph architectures that exploit them. Both need careful tuning for the real goal: catching fraud without false alarms. That means focusing on recall and F1-score, not just accuracy.

Why does this matter? Take a typical payment processor. They might see fraud in 0.5% of transactions. Missing half those cases means millions in losses. But flagging too much legitimate activity is just as bad - it angers customers and wastes analyst time. These relationship patterns exist in the data but are hard to detect using traditional methods. The question is whether graph methods can find them reliably enough.

III. Related Work

[1]. I'm building on several existing threads of research. The paper by Di Giovanni et al. [2] is my starting point - they showed how to adapt GNNs for directed graphs, which matters because money flows in one direction. The Elliptic dataset paper [3] is crucial because it's one of the few public transaction graphs with labeled fraud. They used a simple GCN and got decent results, but they didn't explore directed architectures or handle the imbalance well. Cheng's 2023 survey on GNNs for fraud detection [4] was eye-opening. It confirmed what I suspected - everyone's excited about graph methods, but most papers use proprietary data and ignore the practical issues: how do you build these graphs at scale, what about the imbalance, how do you explain predictions to investigators?. Chao et al. [5] worked on heterogeneous graphs for financial fraud, modeling users, accounts, and transactions as different node types. It's elegant but adds complexity.

IV. Methodology

In this project, I used experimental research design based on mixed methods approach of graph topological analysis and deep learning. I mainly aimed to assess the performance of direction-aware passing of messages in Graph Neural Networks (GNNs) to identify illicit financial activities.

I have used three different sources of data in order to make a comprehensive assessment. It was the Elliptic Bitcoin Data that were obtained at Kaggle. This actual data is a cryptocurrency network with 203,769 nodes (transactions) and 234,355 directed edges, and 166 anonymized features. It is also characterized by great temporal dynamics in 49 time steps.

The second dataset was PaySim (Online Payments), a synthetic and yet very realistic mobile money dataset with 6.36 million transactions. I used a method of class-balanced downsampling to build a computationally viable graph and still have structural integrity. I retained all 8, 213, cases of fraud and used a random sample of 5 percent of legitimate transactions, which created a graph with 592,288 nodes and 325,933 edges.

The third dataset was a custom synthetic data set that I created to specifically test the model with complex fraud behavior, including low homophily with fraudsters dealing with legit users and repetitive triangular patterns.

One of the major constraints of such datasets is severe imbalance in classes. As an example, the portion of fraudulent transactions is less than half in the raw PaySim data (0.12%). To address this, I used strategic down Sampling, a weighted Cross-Entropy loss during training, and stratified split used to make sure that the distribution of fraud was similar to the distribution of fraud in training, validation, and testing.

I developed the end-to-end data architecture and the model architecture using Python. PyTorch and PyTorch Geometric (PyG) were the main technical foundations of the core technical infrastructure to perform tensor operations and construct graph neural networks. Pandas and Scikit-learn were used in preprocessing data and baseline models creation. Topological visualization was done with NetworkX in combination with Matplotlib and Seaborn and an interactive dashboard with Streamlit was created to display node-level fraud probabilities and model summary statistics.

My methodology analysis consisted of the construction of a custom Directed Graph Convolutional Network (Directed GCN). Upon understanding that the directionality and flow of funds are important factors in money laundering, I implemented a tailored PyTorch modules that differentiate between incoming and outgoing aggregation of messages.

In order to improve performance even more, I optimized structural feature enrichment. I derived the topological features of every node (size) algorithmically, which are: in-degree, out-degree, flow asymmetry,

degree imbalance and multi-hop neighborhood characteristics. These features were then concatenated with the baseline features, and then they were inputted into the neural network.

In order to rigorously prove my findings, I did not use accuracy because there was a great imbalance of the classes represented in fraud datasets. In lieu, my evaluation metrics were centered on F1-Score, Precision, Recall and ROC-AUC which offer a more meaningful evaluation of performance on imbalanced classification issues.

I also did large-scale robustness testing of the synthetic dataset, where the Directed GCN is tested at different levels of noise effects. Such experiments involved edge removal, flipped direction, label noise, and masked features to model adversarial or corrupted data.

Lastly, I carried out a systematic hyperparameter tuning pipeline on hidden dimensions, learning rates, and the weight decay values to decompose the best model sets of the Elliptic and PaySim datasets. This guaranteed that the reported final results are the most steady and well-behaving architectures that were found in the course of experimentation.

V. Data Description

I have put in place a firm organizational system of data management. Raw data were stored unaltered in data/raw, and processed data were stored in a data/processed directory in a serialize form of the PyTorch objects. This guaranteed a smooth loading and a high level of reproducibility of experiments. The versions of code were followed in the course of the project.

My quality data pipeline was well-designed. First I confirmed that there were no missing values in the core feature matrices and edge lists. I also programmatically checked the edge lists to make sure that no edge was pointing to non-existent node which eliminated any orphaned transactions that would cause inconsistencies.

PaySim data is extreme skewed in terms of transactions and balance differences. I used log transformation of the transaction amounts and clipped extreme balance differences to the 1st and 99th percentiles to stabilize the model gradients.

Since difference in variance among features was very big, particularly in the Elliptic data dataset that has 166 local and aggregate features, I used standardization to be able to feed all the input matrices into the neural networks with an average of 0 and a standard deviation of 1.

Since the given project involves financial fraud detection, I was guided by a high level of ethical conduct concerning the privacy of data. There is no personal identifiable information in this study. The data providers cryptographically anonymize the Elliptic dataset, which codes real-world objects to meaningless integer identifiers. In the same way, PaySim is produced by a multi-agent simulation model, that is, there are no real people and real bank accounts.

Another serious ethical issue in fraud detection is the problem of algorithmic bias, and this issue can be explained in the following way. It can happen that a model can treat a legitimate user as a fraudster. False positives like these can be very detrimental such as freezing the accounts of innocent persons. To alleviate this risk, I did not optimize models with the aim of global accuracy only.

I instead tuned the models with F1-score and used a close attention on the precision-recall trade-off, to make the detection strategy balanced. The interactive Streamlit dashboard additionally leads to increased transparency since users can view the specific probabilities of fraud that are assigned to each node instead of using some opaque binary classification.

VI. Expected Outcomes

Here's what I'm aiming to produce:

First, trained GNN and baseline models on all three datasets: working models and metrics of performance would indicate whether the graph approach actually helps or not. I expect graph-based models to improve recall compared to XGBoost. Second, an interactive dashboard built using Streamlit. Not only the static plots, but something that a fraud investigator may actually utilize to investigate the findings,

compare different models, and understand why a specific transaction was flagged. Third, a project report that explains all major decisions, including graph construction choices, how I handled data imbalance, which hyperparameters I used, and what did not work.

VII. Conclusion

The proposed practicum proposal is a complete end-to-end model of identifying financial fraud with the help of advanced graph deep learning approaches. The main issue with this project is the weakness of the classical tabular machine learning models which model transactions as events and do not reflect relation dependencies.

The proposed method utilizes the topological flow of funds through transaction networks by using a custom Directed Graph Convolutional Network. This functionality enables the model to reveal advanced money laundering rings and organized cash-out plans many of which may be missed in the complicated financial graphs.

The originality of the work is that it is empirically proven that directionality is a vital factor in financial transaction networks. Experiments of robustness indicated that when edge directions are randomly inverted, the performance of the model was greatly impaired, which demonstrated that even the flow of capital possesses powerful information about some ill intentionality.

Also, the model used to generate structurally designed elements (e.g., flow asymmetry) and incorporated them into the network, which, at the Elliptic dataset, a challenging dataset, the model achieved a competitive F1-score of 0.4544, which is significantly higher than baseline undirected graph convolutional models.

VIII. Timeline

Week 1: Finalize project scope and proposal, select datasets, and initialize github repository and dev environment

Week 2: Exploratory data analysis on all datasets, document quality issues

Week 3: Data cleaning, feature engineering for both tabular and graph inputs

Week 4: Build directed graphs, train tabular baseline models

Week 5: Train GNN baseline on PaySim and Elliptic, tune basic hyperparameters

Week 6: Train enhanced models (gated GNN, GAT, GraphSAGE), extensive hyperparameter tuning

Week 7: Cross-model evaluation, error analysis, build Streamlit dashboard

References

[1] W. Zheng, B. Xu, E. Lu, Y. Li, Q. Cao, X. Zong, and H. Shen, "Midlg: Mutual information based dual-level graph neural network for transaction fraud complaint verification," in *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '23)*. ACM, 2023, pp. 5685–5694. [Online]. Available: https://doi.org/10.1145/3580305.3599865

[2] F. Di Giovanni, J. Rowbottom, B. P. Chamberlain, T. Markovich, and M. M. Bronstein, "Understanding convolution on graphs via energies," *Transactions on Machine Learning Research (TMLR)*, 2023, arXiv:2206.10991. [Online]. Available: https://openreview.net/forum?id=9oP6f7fX5X

[3] M. Weber, G. Domeniconi, J. Chen, D. K. I. Weidele, C. Bellei, T. Robinson, and C. E. Leiserson, "Anti-money laundering in bitcoin: Experimenting with graph convolutional networks for financial forensics," in *Proceedings of the KDD Workshop on Anomaly Detection in Finance*, 2019. [Online]. Available: https://arxiv.org/abs/1908.02591

[4] D. Cheng, Y. Zou, S. Xiang, and C. Jiang, "Graph neural networks for financial fraud detection: A survey," *Frontiers of Computer Science*, vol. 19, no. 1, 2025. [Online]. Available: https://doi.org/10.1007/s11704-024-40474-y

[5] B. Wu, K.-M. Chao, and Y. Li, "Heterogeneous graph neural networks for fraud detection and explanation in supply chain finance," *Information Systems*, vol. 121, p. 102335, 2024. [Online]. Available: https://doi.org/10.1016/j.is.2023.102335

[6] X. Liu, X. Wu, and Y. Ren, "efraudcom: An e-commerce fraud detection system via competitive graph neural networks," in *Proceedings of the 15th ACM International Conference on Web Search and Data Mining (WSDM '22)*, 2022, pp. 1438–1441. [Online]. Available: https://doi.org/10.1145/3488560.3502194

[7] Y. Wang, Y. Liu, Y. Zhang, and H. Zha, "Bright – graph neural networks in real-time fraud detection," in *Proceedings of the 31st ACM International Conference on Information & Knowledge Management (CIKM '22)*, 2022, pp. 4575–4579. [Online]. Available: https://doi.org/10.1145/3511808.3557088

[8] Y. Tian, G. Liu, J. Wang, and M. Zhou, "Transaction fraud detection via an adaptive graph neural network," *arXiv preprint*, 2023, arXiv:2307.05633. [Online]. Available: https://arxiv.org/abs/2307.05633

[9] G. Goyal, R. Tyagi, and S. Tyagi, "Graph neural networks for fraud detection in e-commerce transactions," in *2024 International Conference on Computing, Sciences and Communications (ICCSC)*, 2024, pp. 1–6. [Online]. Available: https://doi.org/10.1109/ICCSC62048.2024.10830450

[10] A. Dauletov, K. Bakhrieva, A. Azamatov, M. Babajanov, N. M. Yaacob, and S. A. Abd, "Graph contrastive learning for fraud detection in financial transactions using ai-powered anomaly detection in banking and e-commerce," in *2025 3rd International Conference on Cyber Resilience (ICCR)*, 2025. [Online]. Available: https://doi.org/10.1109/ICCR67387.2025.11292552