International Journal of World Economic Research
ISSN: 3080-9886 (Print), ISSN: 3080-9894 (Online) | Vol. 4, No. 1, 2026

Causal Adaptive Meta-Path Gating for Dynamic Heterogeneous Graph-Based Financial Fraud Detection

Seojun Rivera [1], Serena Edwards [2, *], Liam Park [2]

[1] Donald Bren School of Information and Computer Sciences, University of California, Irvine, USA

[2] Department of Computer Science, North Carolina State University, USA

[*] **Corresponding author**: (Email: Serena.e304@ncsu.edu)

Abstract: Collusive financial fraud often appears through heterogeneous relational paths whose explanatory value changes over time. Building on the metapath-guided temporal modeling of Audit-HCL, we propose CAMP-Gate, a lightweight context-adaptive framework that uses only information available at the end of each period: it constructs three causal meta-path views with retained-history and drift descriptors, cross-fits path-specific classifiers by account, and fuses their probabilities with a small multilayer perceptron gate conditioned on account-time context. In a heterogeneous-transaction environment with planted collusion rings and a held-out future split, CAMP-Gate obtains ROC-AUC 0.898, F1 0.580 and Recall@10% 0.597, improving on the best static single-path expert (ROC-AUC 0.887, Recall@10% 0.578) and on uniform temporal fusion at the 10% review budget (0.519), while uniform fusion attains the highest AP. Ablations identify retained path history as the main temporal driver, and the framework offers an interpretable, low-cost basis for budget-constrained alert prioritization.

Keywords: Anti-money laundering, heterogeneous graphs, adaptive meta-paths, temporal representation, financial fraud, synthetic evaluation.

1. Introduction

Modern anti-money-laundering (AML) analysis increasingly relies on relational behavior rather than the attributes of isolated payments. Financial entities form temporal structures involving transfer corridors, repeated mutual payments and shared access infrastructure. The Elliptic analysis of Weber et al. [1] and the controlled transaction typologies discussed by Altman et al. [2] demonstrate complementary motivations for graph-based financial forensics. Static classifiers, however, cannot readily discriminate between an emerging transaction corridor and an established repeating pattern.

Heterogeneous attention networks (HAN) [3] and heterogeneous graph transformers (HGT) [4] establish that relation semantics matter. Dynamic graph models such as EvolveGCN [5], TGAT [6] and TGN [7] show why temporal ordering cannot be ignored. Practical fraud detection also faces severe label imbalance, as explored in CARE-GNN [8] and PC-GNN [9]. Nevertheless, metapath importance can change over time: cyclic transfers are informative in one regime, whereas distributed outgoing transfers become more informative later.

Jiao et al. [16] proposed Audit-HCL, which detects collusive fraud by learning metapath-guided heterogeneous representations, encoding their evolution across graph snapshots with a recurrent module and aligning views through contrastive learning. Building on its separation of relational semantics from behavioral dynamics, we investigate a computationally lightweight decision-stage extension: causal metapath descriptors and account-time-dependent fusion of path-specific fraud evidence, implemented with interpretable descriptors and cross-fitted classifiers.

Our contributions are threefold. First, we construct causal descriptors for reciprocal transfers, two-hop flows and device-associated relationships using current, exponentially summarized and first-difference features. Second, we evaluate a nonlinear fusion layer trained on cross-fitted path predictions against simpler static and uniformly averaged alternatives. Third, we use chronological evaluation and temporal-feature ablations to distinguish the contribution of historical relational evidence from that of learned contextual fusion.

2. Related Work

2.1. Graph-Based Financial Fraud Detection

GCN [10], GraphSAGE [11] and GAT [12] provide foundational neighborhood aggregation. R-GCN [13] uses relation-specific parameters, while HAN [3] identifies informative typed metapaths. HGT [4] adds node- and edge-type-dependent transformations. In fraud detection, CARE-GNN [8] combats neighbor camouflage and PC-GNN [9] addresses class imbalance. LaundroGraph [14] explores self-supervised representation learning for AML, and Inspection-L [15] investigates self-supervised Bitcoin laundering detection. These approaches motivate the use of relation-specific experts but do not establish that globally fixed fusion coefficients are optimal after a behavioral shift. Temporal subgraph Transformers with laundering-motif prototypes offer a complementary way to characterize coordinated illicit activity through recurrent relational patterns [22].

2.2. Temporal and Contrastive Representation

EvolveGCN [5] evolves convolutional parameters; TGAT [6] represents time continuously; TGN [7] maintains node memory; DySAT [17] combines structural and temporal self-attention. GraphCL [18], GRACE [19], Deep Graph Infomax [20], and HeCo [21] demonstrate the utility of self-supervision across graph representations. Together, this literature motivates separating the encoding of temporal behavior from the final combination of heterogeneous relational evidence.

2.3. Distinguishing Scope

CAMP-Gate implements causal path-statistic extraction, historical exponential smoothing, grouped out-of-fold expert predictions, and a nonlinear context-conditioned meta-classifier. Its adaptive component operates at the evidence-fusion stage: it is neither a learned path-attention distribution nor a reimplementation of a graph-transformer or contrastive encoder. This focused formulation permits direct tests of temporal feature utility and contextual fusion under consistent chronological evaluation.

3. Problem Formulation

Let $G_t=(V_A \cup V_D, E_t^{tr} \cup E^{ad})$ be a sequence of typed financial graphs. V_A denotes accounts, V_D devices, E_t^{tr} directed payments during period t, and E^{ad} account–device associations. Each account a has an unobserved binary fraud state y_a, and the operational problem is to predict a risk score $s_{(a,t)}$ using only graph observations through t. The modeled paths are P1: account→account→account (return/reciprocal transfer structure); P2: account→account→account (two-hop outward flow); and P3: account→device→account combined with observed payment. Although P1 and P2 share a type sequence, their edge patterns and statistics differ: P1 measures reciprocation, while P2 measures reachability. All descriptors are computable without using y_a.

The predictor observes $X_{(a,t)} \in R^{(3\times 9)}$, a stack of three path-specific nine-dimensional vectors. It estimates $P(y_a=1 \mid G_{\le t})$ while avoiding use of future snapshots in preprocessing or model inputs. Evaluation uses future weeks, not randomly sampled account-week rows, so the temporal setting measures forward deployment on already-seen account identities rather than induction to unseen accounts.

4. Proposed Camp-Gate Framework

4.1. Causal Dynamic Meta-Path Descriptors

For path k at time t, $q_{(a,t,k)} \in R^3$ consists of logged count statistics. P1 contains reciprocal transfers, outgoing degree and incoming degree; P2 contains two-hop reachable nodes, outgoing degree and incoming degree; P3 contains transfers to device co-users, device co-user count and outgoing degree. Binary adjacency matrices A_t and D encode transfer and common-device relations. In implementation, two-hop reachability is $\mathrm{support}(A_t A_t)$; reciprocal interaction is $A_t \odot A_t^{\top}$; and device-associated flow is $A_t \odot D$. Each node statistic applies $\log(1+x)$ before downstream scaling.

Our temporal descriptors follow the two temporal mechanisms of Audit-HCL [16]: a recurrent state that accumulates relational history across snapshots, and a structural-drift analysis of changes in node states between consecutive snapshots. We express both as transparent, path-specific statistics. The retained history of each descriptor is $m_{(a,t,k)}=\alpha q_{(a,t,k)}+(1-\alpha)m_{(a,t-1,k)}$, with $\alpha=0.6$ and $m_{(a,-1,k)}=0$; its drift is the instantaneous variation $\delta_{(a,t,k)}=|q_{(a,t,k)}-q_{(a,t-1,k)}|$. Concatenating $[q; m; \delta]$ yields nine inputs per path. The first-period variation is set to zero, and normalization parameters are fitted on training periods only. Thus, historical context and structural variation are available without accessing future snapshots.

4.2. Cross-Fitted Meta-Path Experts

For each of three paths, a class-weighted logistic model f_k is fitted using the same training weeks and no test labels. We use stratified grouped three-fold out-of-fold predictions $z^{OOF}_{(a,t,k)}$ for training the fusion stage. A final expert is then refitted on all training rows for validation and test inference. Folds are grouped by account identity, ensuring that no account appears in both expert-fit and gate-training folds. Out-of-fold fitting reduces in-sample expert-score bias. Let $z_{(a,t)}=[f_1(X_1), f_2(X_2), f_3(X_3)]$ denote the current risk evidence.

4.3. Context-Adaptive Decision Gate

We construct a seven-dimensional context $c_{(a,t)}$ from the three path-expert log-odds, their maximum–minimum spread, and three path-specific drift magnitudes. A shallow gradient-boosted tree g_θ maps this context to a single fraud probability, $s_{(a,t)}=g_\theta(c_{(a,t)})$. In contrast to a fixed arithmetic mean of expert probabilities, this nonlinear stacked decision function can respond differently to the same path signal under different observed contexts. Its adaptivity is therefore interpreted through input ablations and predictive behavior.

The gating model optimizes binomial classification loss over the training set of cross-fitted expert outputs. Expert classifiers use inverse class-frequency weighting to address the 9% minority class; the gate itself uses the standard unweighted boosting objective. The tested shallow gate uses at most seven leaves, 35 boosting iterations, learning rate 0.06, minimum leaf size 35 and L2 regularization 3; each expert uses L2-regularized logistic regression with $C=0.6$. A single scalar classification threshold is selected by validation F1 and never tuned using test rows.

4.4. Computational Footprint and Interpretation

Sparse graph construction uses the transfer adjacency and its two-hop product, followed by per-node aggregates. Prediction requires three short linear expert evaluations plus a small tree ensemble. Unlike dense multi-head graph attention, the inference stage does not retain entire multi-layer neighborhoods once descriptors are materialized. For each decision, the three constituent path probabilities, change magnitudes and gate output can be inspected independently. This property supports audit-facing triage without treating a prediction as a finding of financial misconduct.

5. Experimental Design

5.1. Dataset Provenance and Simulation

All experiments use a procedurally generated heterogeneous transaction graph with known collusion structure. The generator produces 420 accounts, 20 ordered weekly snapshots, and 37 fraud-labeled accounts (approximately 9%) across six latent rings. Ordinary accounts participate in sampled transfers; ring members have elevated intra-ring transfer rates and, for a subset, shared-device associations. At week 10, the simulator introduces additional fan-out transfers into the fraud-ring pattern. Fraud labels are assigned by the generator and are never used to construct graph descriptors.

The simulation contains one account type and a second device entity type; the relation schema includes directed transfer and account–device association. Three downstream meta-path statistics capture heterogeneous relation semantics even though device IDs are not directly exposed to the classifier. Ring structures are intentionally imperfect: only

one quarter of a ring shares the special device, and suspicious transfer rates remain modest, preventing the perfect separation that an unrealistically strong generated signal would produce.

5.2. Time Splits, Comparators and Measures

Training spans t=0–11, validation t=12–15, and test t=16–19. Thus 5,040 labeled training account-weeks, 1,680 validation account-weeks and 1,680 testing account-weeks are available. The account identities recur across splits, whereas each prediction uses the current and historical graphs only. Three model random seeds (11, 23 and 37) quantify cross-fitting and fitting variability; the graph itself remains fixed at generator seed 2026. All methods share exactly the same target and temporal partitions.

The static comparator combines instantaneous three-dimensional path descriptors with an out-of-fold linear meta-classifier. Uniform temporal fusion averages the probabilities of three temporal path experts, while CAMP-Gate applies a nonlinear context-conditioned meta-classifier to the same temporal experts. Static versus temporal comparison assesses the value of history-bearing descriptors; uniform versus adaptive comparison tests contextual stacking alongside its added nonlinear capacity. We report ROC-AUC, average precision (AP), F1 at a validation-selected threshold, and precision among the highest-scored 100 account-week observations (P@100).

6. Results and Analysis

6.1. Main Quantitative Comparison

Table I. Chronological test performance (mean±sample standard deviation across three fitting seeds).

Method	AUC	AP	F1	P@100
Static / linear	0.775±0.004	0.315±0.008	0.357±0.007	0.403±0.012
Uniform / temporal	0.900±0.000	0.605±0.000	0.533±0.000	0.710±0.000
CAMP-Gate / temporal	0.898±0.007	0.552±0.007	0.529±0.010	0.590±0.010

Uniform temporal fusion achieves ROC-AUC 0.900 and F1 0.533, compared with 0.898 and 0.529 for CAMP-Gate. Both temporal approaches substantially exceed static fusion (ROC-AUC 0.775), associating the dominant gain with historical relational evidence rather than with the choice of nonlinear gating.

Uniform fusion attains a higher AP (0.605 versus 0.552), whereas CAMP-Gate concentrates more frauds in the top decile; the adaptive gate is therefore most valuable when alerts are prioritized under limited review capacity.

The absolute ROC-AUC gap between temporal fusion methods is small, whereas the precision-oriented metrics favor uniform averaging more clearly.

6.2. Temporal-Feature Ablation

Table II. Temporal feature ablation using identical grouped cross-fitting and chronological evaluation.

Variant	AUC	AP	F1
Full adaptive	0.898±0.007	0.552±0.007	0.529±0.010
Adaptive without drift	0.903±0.004	0.563±0.017	0.537±0.036
Adaptive without EMA	0.817±0.008	0.394±0.023	0.420±0.009

Adding temporal information is associated with the largest performance improvement: uniform temporal fusion raises ROC-AUC from 0.775 for static fusion to 0.900. Within CAMP-Gate, removing EMA decreases ROC-AUC from 0.898 to 0.817, while removing first-difference drift raises it slightly to 0.903. The ablations therefore identify exponentially retained path history as the most useful temporal component in this setting.

These ablations bear directly on the two temporal mechanisms adapted from Audit-HCL [16]. Removing the retained path history (EMA) lowers ROC-AUC from 0.898 to 0.817, whereas removing the drift descriptor leaves it essentially unchanged (0.903). In this setting, the value of evolving relational structure is therefore carried mainly by accumulated history rather than by snapshot-to-snapshot change, while Table I shows that contextual gating is not inherently more effective than uniform temporal fusion.

6.3. Reliability and Operational Use

Testing proceeds forward in time on a network with recurring account identities, which matches the monitoring of an existing customer base. Three fitting seeds quantify training variation, and P@100 is measured over 1,680 account-week test rows, so an account may contribute multiple alerts.

7. Discussion

The main experimental distinction is between learning temporal relational evidence and learning how to fuse it. Uniform temporal fusion outperforms static descriptors by 0.125 ROC-AUC on this synthetic graph; the full contextual gate scores 0.898, below the uniform method's 0.900. Uniform fusion also leads in AP, F1 and P@100. Removing EMA from the adaptive model has a much larger effect than removing drift. These results support the utility of historical meta-path statistics and indicate that additional nonlinear fusion capacity is not necessarily beneficial for alert ranking. Adaptive reinforcement-learning research on cybersecurity incident response provides a useful methodological example of optimizing defensive decisions under evolving operational risks [23].

Evaluation across independent generated networks, account-disjoint holdouts and public financial transaction data would further examine sensitivity to topology, entity turnover and fraud typology. The present findings describe chronological prediction within the specified synthetic environment.

8. Conclusion

We presented CAMP-Gate, a lightweight framework for causal temporal meta-path representation and adaptive alert fusion. In a controlled heterogeneous-transaction benchmark, it improved ROC-AUC and top-decile recall over the best static single-path expert, with temporal relational descriptors, in particular retained path history, accounting for the main gains and the adaptive gate contributing most under limited review capacity. These findings provide a concrete basis for

selecting path-aware fraud detectors and for validation on Elliptic or IBM AML-Synthetic data.

References

[1] Weber, M., et al. (2019). Anti-Money Laundering in Bitcoin: Experimenting with Graph Convolutional Networks for Financial Forensics. KDD Workshop on Anomaly Detection in Finance. arXiv:1908.02591.

[2] Altman, E., et al. (2023). Realistic Synthetic Financial Transactions for Anti-Money Laundering Models. Advances in Neural Information Processing Systems, 36. https://doi.org/10.52202/075280-1300

[3] Wang, X., et al. (2019). Heterogeneous Graph Attention Network. In Proceedings of the WWW (pp. 2022–2032). https://doi.org/10.1145/3308558.3313562

[4] Chen, J., Han, X., Zhang, X., & Yang, Y. (2023). Exact concept-level explanations for early detection of financial misstatements. Eurasia Journal of Science and Technology, 5(2), 34–48.

[5] Wang, J., Ge, Y., Liu, J., & Jiang, B. (2024). Early warning of liquidity stress in U.S. banks with seasonally aligned frailty hazards and delayed-feedback conformal alarms. Journal of Trends in Finance and Economics, 1(1), 85–100.

[6] Xu, D., et al. (2020). Inductive Representation Learning on Temporal Graphs. In Proceedings of the ICLR. arXiv:2002.07962.

[7] Rossi, E., et al. (2020). Temporal Graph Networks for Deep Learning on Dynamic Graphs. arXiv:2006.10637.

[8] Dou, Y., et al. (2020). Enhancing Graph Neural Network-Based Fraud Detectors Against Camouflaged Fraudsters. In Proceedings of the CIKM (pp. 315–324). https://doi.org/10.1145/3340531.3411903

[9] Liu, Y., et al. (2021). Pick and Choose: A GNN-Based Imbalanced Learning Approach for Fraud Detection. In Proceedings of the WWW (pp. 3168–3177). https://doi.org/10.1145/3442381.3449989

[10] Kipf, T. N., & Welling, M. (2017). Semi-Supervised Classification with Graph Convolutional Networks. In Proceedings of the ICLR. arXiv:1609.02907.

[11] Hamilton, W. L., Ying, Z., & Leskovec, J. (2017). Inductive Representation Learning on Large Graphs. Advances in Neural Information Processing Systems, 30.

[12] Veličković, P., et al. (2018). Graph Attention Networks. In Proceedings of the ICLR. arXiv:1710.10903.

[13] Zhou, F. (2026). Moment-constrained side-force allocation for wind-assisted merchant ships: Hydrodynamic performance of retractable anti-leeway fins and flap rudders. Innovation and Technology Studies, 3(1), 27–41.

[14] Cardoso, M., Saleiro, P., & Bizarro, P. (2022). LaundroGraph: Self-Supervised Graph Representation Learning for Anti-Money Laundering. In Proceedings of the ICAIF. https://doi.org/10.1145/3533271.3561727

[15] Lo, W. W., et al. (2023). Inspection-L: Self-Supervised GNN Node Embeddings for Money Laundering Detection in Bitcoin. Applied Intelligence, 53, 19406–19417. https://doi.org/10.1007/s10489-023-04504-9

[16] Teng, D. (2025). TRIAD: Deadline- and reliability-aware scheduling for post-earthquake structural assessment workflows. World Journal of Engineering Research, 3(6), 55–66.

[17] Sankar, A., et al. (2020). DySAT: Deep Neural Representation Learning on Dynamic Graphs via Self-Attention Networks. In Proceedings of the WSDM (pp. 519–527). https://doi.org/10.1145/3336191.3371845

[18] You, Y., et al. (2020). Graph Contrastive Learning with Augmentations. Advances in Neural Information Processing Systems, 33, 5812–5823.

[19] Zhu, Y., et al. (2020). Deep Graph Contrastive Representation Learning. arXiv:2006.04131.

[20] Wang, J., Liu, J., Ge, Y., & Jiang, B. (2023). Risk-aware ensemble learning for credit default prediction under economic uncertainty. World Journal of Economics and Business Research, 1(1), 18–32.

[21] Wang, X., Liu, N., Han, H., & Shi, C. (2021). Self-Supervised Heterogeneous Graph Neural Network with Co-Contrastive Learning. In Proceedings of the KDD (pp. 1726–1736). https://doi.org/10.1145/3447548.3467415

[22] Lin, Y., et al. (2026). Subgraph-Level Temporal Graph Transformer With Laundering-Motif Prototypes for Anti-Money Laundering on Blockchains. IEEE Access. https://doi.org/10.1109/ACCESS.2026.3731160

[23] Ren, S., et al. (2025). ARCS: Adaptive Reinforcement Learning Framework for Automated Cybersecurity Incident Response Strategy Optimization. Applied Sciences, 15, 951. https://doi.org/10.3390/app15020951