Spatio-Temporal Directed Graph Learning for Account Takeover Fraud Detection

Mohsen Nayebi Kerdabadi William Andrew Byron Xin Sun Amirfarrokh Iranitalab

NeurIPS 2025 — New Perspectives in Advancing Graph Machine Learning (NGML)

scan me!

Abstract

Account Takeover (ATO) fraud poses a significant challenge in consumer banking, demanding high recall under tight latency while keeping friction low for legitimate users. Production systems often use tabular gradient-boosted trees (e.g., XGBoost) that score sessions independently and ignore the relational, temporal structure of coordinated attacks. We propose **ATLAS** (**A**ccount **T**akeover **L**earning **A**cross **S**patio-Temporal Directed Graph), which casts ATO detection as node classification on a time-respecting directed session graph linking entities via shared identifiers (account, device, IP) under time-window and recency constraints. ATLAS uses causal, leakage-free message passing and latency-aware label propagation, instantiated with inductive GraphSAGE variants trained via neighbor sampling on a 100M+ node, ~1B-edge graph. On a high-risk Capital One digital product, ATLAS achieves +6.38% AUC and >50% reduction in customer friction.

Our Contribution

1. We reformulate ATO as spatio-temporal graph learning with node classification over sessions on a time-respecting directed acyclic graph, enabling causal, time-respecting message passing and lag-aware label propagation.
2. We construct a directed temporal graph with strict causal ordering (past-to-future); link entities via shared identifiers (account, device, IP); and regulate connectivity with designed time-window and recency constraints. We operationalize this with an inductive GraphSAGE approach.
3. We conduct extensive experiments on a high-risk digital product dataset at Capital One financial institution (100M+ nodes, ~1B edges), demonstrating the effectiveness of our approach: +6.38% AUC and >50% reduction in customer friction, improving fraud capture while lowering friction.

Method Overview

We present an overview of our proposed method, ATLAS, a spatio-temporal directed-graph approach to ATO Fraud Detection.

1. **Graph Formulation Strategy.** Model each high-risk transaction session as a node with tabular features; add directed edges from past to future sessions that share an identifier (account, device, IP), restricted by a time window T and a recency cap K to preserve causality and bound node degree.
2. **Lag-aware Label Propagation.** For each node, aggregate labels only from in-neighbors whose outcomes are known by serve time, and convert them into simple leakage-free features (e.g., counts, fraud rate) concatenated to the original node features.
3. **GNN Architecture.** Use an inductive GraphSAGE-style encoder over this temporal graph, taking as input the tabular + propagated-label features and outputting node embeddings passed through a logistic head for fraud risk prediction.

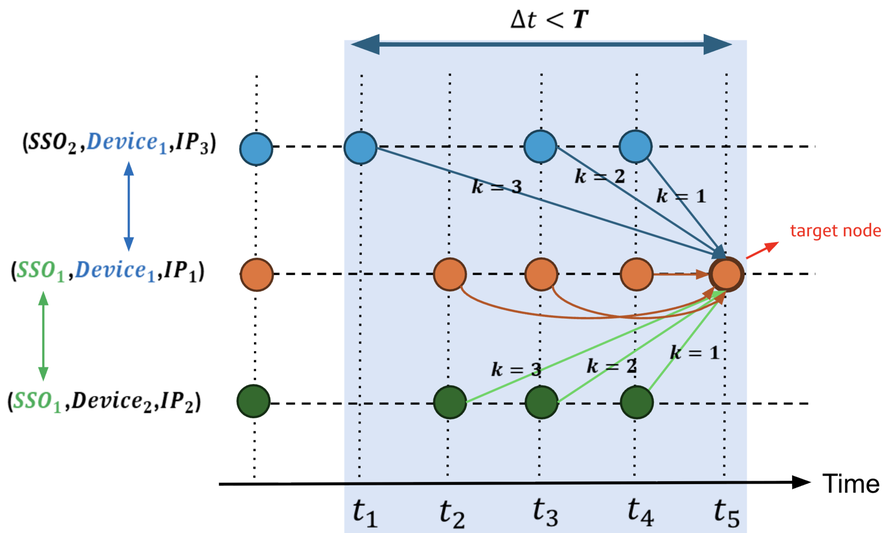

Figure 1: ATLAS formulation. Nodes are HRT sessions keyed by (`account_id`, `device_id`, `ip_address`, timestamp). Directed edges connect past→future sessions that share an identifier, constrained by time window T and recency cap K.

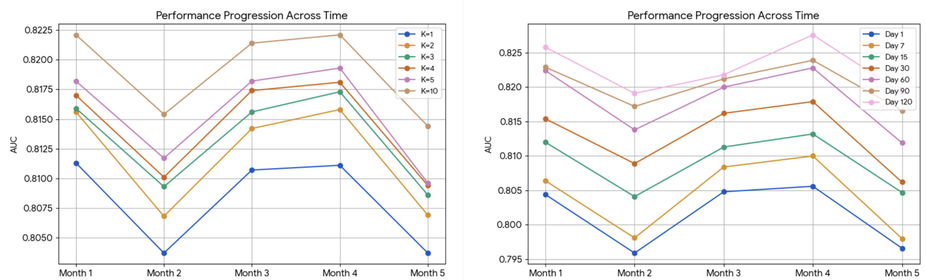

Figure 2: Effect of K (left) and T (right) on ROC AUC for Segment 1 and Segment 2.

Table 1: Performance comparison of XGBoost (XGB) vs. GNN based on AUC percentage.

Model	Overall	Segment 1	Segment 2
XGB	79.83	78.88	82.45
GNN	82.27	81.59	83.82
Improvement (vs. XGB)	*+3.06%*	*+3.43%*	*+1.66%*
GNN + Label Propagation	84.46	83.92	85.45
Improvement (vs. XGB)	*+5.8%*	*+6.38%*	*+3.63%*

1: Graph Formulation Strategy

We cast ATO detection as spatio-temporal node classification on a directed session graph $G = (V, E)$, where nodes are high-risk transaction (HRT) sessions and edges encode temporal, identifier-based links from past to future sessions.

Nodes. Each node $v \in V$ is an HRT session keyed by (account id, device id, ip address), with feature vector $\mathbf{x}_v \in \mathbb{R}^d$ and binary fraud label $y_v \in \{0, 1\}$. The model outputs a serve-time risk score

$$s_v = \Pr(y_v = 1 \mid G_{\prec v}, X_{\preceq v}),$$

where $G_{\prec v}$ contains only sessions with $t_u < t_v$ and $X_{\preceq v}$ are features observable by serve time, ensuring non-anticipative, leakage-free scoring.

Edges. For sessions u, v with $t_u < t_v$, we add a directed edge $(u \to v)$ of type $m \in \mathcal{M} = \{\text{account}, \text{device}, \text{IP}\}$ if they share identifier m, yielding

$$E = \bigcup_{m \in \mathcal{M}} E_m, \quad E_m = \{(u \to v) : t_u < t_v,\ u \text{ and } v \text{ share identifier } m\}.$$

All edges respect $t_u < t_v$, so G is a time-respecting DAG.

Connectivity constraints. To preserve causality and control degree/latency, we restrict edges via:

- *Time window* T: include $(u \to v)$ only if $0 < t_v - t_u \le T$;
- *Recency cap* K: for each v and type m, keep at most the K most recent predecessors in E_m.

2: Lag-aware Label Propagation

We add *serve–time–correct* label features to each session v, using only past neighbors whose outcomes are known by time t_v.

Serve-time neighborhood. From the directed graph $G = (V, E)$, define recent in-neighbors

$$\mathcal{N}_T^-(v) = \{\, u : (u \to v) \in E,\ 0 < t_v - t_u \le T \,\},$$

and keep at most the K most recent:

$$\mathcal{R}(v) = \operatorname*{TopK}_{u \in \mathcal{N}_T^-(v)} (t_u), \quad |\mathcal{R}(v)| \le K.$$

Each u has label $y_u \in \{0, 1\}$ revealed at adjudication time τ_u; we only use labels known by serve time:

$$\mathcal{A}(v) = \{\, u \in \mathcal{R}(v) : \tau_u \le t_v \,\}.$$

Aggregated label features. From $\mathcal{A}(v)$ we compute

$$n_v^{\text{lab}} = |\mathcal{A}(v)|, \quad n_v^{\text{fraud}} = \sum_{u \in \mathcal{A}(v)} y_u, \quad r_v = \frac{n_v^{\text{fraud}}}{\max(1, n_v^{\text{lab}})}, \quad a_v = \mathbb{1}[n_v^{\text{fraud}} \ge 1].$$

with $r_v = a_v = 0$ if $n_v^{\text{lab}} = 0$. We form

$$\ell_v = [\, n_v^{\text{lab}},\ n_v^{\text{fraud}},\ r_v,\ a_v \,], \qquad \mathbf{h}_v^{(0)} = [\, \mathbf{x}_v \,;\, \ell_v \,],$$

and apply the same $(T, K, \tau_u \le t_v)$ rules at train and serve time, ensuring causal, leakage-free label propagation.

3: GraphSAGE Architecture

On the time-respecting session graph (with the same (T, K) constraints), each layer k updates node v by mean-aggregating sampled past neighbors $\mathcal{S}^{(k)}(v)$:

$$\mathbf{m}_v^{(k)} = \frac{1}{|\mathcal{S}^{(k)}(v)|} \sum_{u \in \mathcal{S}^{(k)}(v)} \mathbf{h}_u^{(k-1)}, \qquad \mathbf{h}_v^{(k)} = \sigma\Big(W^{(k)} [\mathbf{h}_v^{(k-1)} \,;\, \mathbf{m}_v^{(k)}] + \mathbf{b}^{(k)}\Big),$$

and $\mathbf{h}_v^{(L)}$ is fed to a logistic head for fraud risk scoring.