ReadGraph: Relational Evolution Enhanced Anomaly Detection in Dynamic Heterogeneous Graph

Xiaolin Han
Northwestern Polytechnical University
Xi'an, China
xiaolinh@nwpu.edu.cn

Xiurui Hu
Northwestern Polytechnical University
Xi'an, China
huxr@mail.nwpu.edu.cn

Chenhao Ma
The Chinese University of Hong Kong, Shenzhen
Shenzhen, China
machenhao@cuhk.edu.cn

Xuequn Shang*
Northwestern Polytechnical University
Xi'an, China
shang@nwpu.edu.cn

Abstract

Abnormal behavior detection is crucial in many fields, such as social networks, financial transactions, and cybersecurity. However, it poses significant challenges due to the intricate structural evolution of heterogeneous graphs. To address this issue, we propose a novel method called Relational Evolution enhanced Anomaly Detection in dynamic heterogeneous Graph (**ReadGraph**). ReadGraph focuses on tracing relation-based dynamic structural evolution to comprehensively capture features related to abnormal behaviors (edges) across different types of nodes. We conduct extensive experiments to evaluate ReadGraph against advanced competitors. It demonstrates that ReadGraph is 13.69% more effective than other methods on average.

CCS Concepts

• **Information systems** → **Spatial-temporal systems**.

Keywords

Anomaly detection, Dynamic heterogeneous graph

ACM Reference Format:
Xiaolin Han, Xiurui Hu, Chenhao Ma, and Xuequn Shang. 2025. ReadGraph: Relational Evolution Enhanced Anomaly Detection in Dynamic Heterogeneous Graph. In *Companion Proceedings of the ACM Web Conference 2025 (WWW Companion '25), April 28-May 2, 2025, Sydney, NSW, Australia.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3701716.3715509

1 Introduction

Anomaly detection is crucial in real-world applications such as identifying suspicious activities in social networks, detecting fraud in financial transactions, and uncovering network intrusions in cybersecurity [1]. In particular, detecting anomalous behaviors (edges) is essential because it can uncover potentially harmful or fraudulent activities that might otherwise go unnoticed [17]. For example, in social networks, identifying anomalies in user interactions can expose malicious activities, such as behaviors aimed at manipulating public opinion or spreading misinformation [26].

As anomalies may change over time, many methods propose to detect anomalous behaviors in dynamic graphs [9, 20]. They employ both temporal and structural patterns to identify anomalous edges in evolving graphs. However, they do not account for the heterogeneity of dynamic graphs, as the types of nodes and edges are not the same for most scenarios [7, 15, 17, 19]. The state-of-the-art (SOTA) method THGNN [17] further integrates the heterogeneous information of the edges, as the types of nodes and edges are not the same for most scenarios [19, 27]. Nevertheless, THGNN [17] does not capture the dynamic structural evolution of relations in the heterogeneous graph. It primarily focuses on single relation representations and may overlook the temporal structural changes in the relationships between nodes.

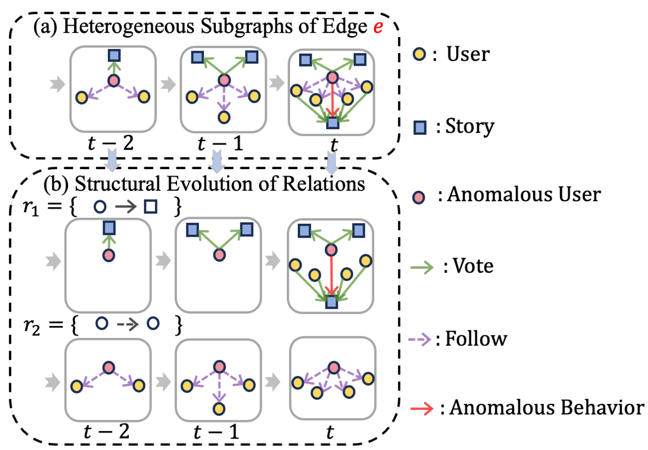

Figure 1: (a) Heterogeneous subgraphs for the target edge e **(in red); (b) Structural evolution of two relations** r_1 **and** r_2**.**

In contrast, capturing the dynamic evolution of structural relationships offers a more comprehensive understanding of how relationships between nodes change over time, which is crucial for accurate anomaly detection. Specifically, Figure 1 illustrates two

*Xuequn Shang is the corresponding author.

WWW Companion '25, Sydney, NSW, Australia

ACM ISBN 979-8-4007-1331-6/2025/04
https://doi.org/10.1145/3701716.3715509

types of relations in the heterogeneous subgraphs of the target edge e (in red color at the snapshot t): voting (r_1) and following (r_2). For the voting (r_1) relation in "user votes story", the anomalous user (in red) initially votes normally at the snapshot $t-2$, continues with normal voting at $t-1$, finally, votes abnormally to one story at the snapshot t, influencing other normal users to vote the same way. For the following (r_2) relation in "user follows user", the anomalous user follows normal users from $t-2$ to t to disguise himself gradually. We can see that the user camouflages himself through normal voting and following before conducting abnormal behavior at t. To fix this issue, we propose ReadGraph, a novel method that captures the evolution of relational structures to reveal underlying behaviors and detect anomalies effectively.

Our contributions. Our goal is to improve anomaly detection for edges by leveraging relation-aware structural evolution for dynamic heterogeneous graphs. We start by sampling heterogeneous subgraphs around each target edge using strategies that account for types of nodes and edges. We then construct relation subgraphs from these heterogeneous samples and perform relation-level message passing to encode them. To capture dynamic structural changes across different snapshots, we use Gated Recurrent Units (GRUs) for sequential modeling. This approach enables us to detect changes in relation-aware structures, thereby enhancing anomaly detection in dynamic heterogeneous graphs. Finally, we perform extensive evaluation using three real datasets against existing methods. ReadGraph demonstrates a significant average increase of 13.69% in effectiveness compared to the SOTA.

2 Related Work

We discuss related work on three categories: static heterogeneous graphs [8, 16, 33, 35], dynamic homogeneous graphs [5, 11, 12, 18, 28], and dynamic heterogeneous graphs [6, 10, 21, 32, 34, 35]. Besides, there is also some related work on static homogeneous graphs [13, 14, 22–25, 30, 31], which will not be discussed further.

Anomaly Detection on Static Heterogeneous Graphs. RGCN [27] introduces graph convolutional networks to capture the features behind the relational data. HAN [19] uses node and semantic level attentions for heterogeneous graph representation.

Anomaly Detection on Dynamic Homogeneous Graphs. The TADDY [20] utilizes Transformer to capture spatial-temporal patterns in dynamic graphs for anomaly detection. MIDAS [2] detects suddenly arriving groups of anomalous edges in edge streams for real-time detection. AnoGraph [3] employs sketch data structure for higher-order modeling of anomalies in temporal graphs. RustGraph [9] learns structural-temporal dependency by a variational graph auto-encoder for robust anomaly detection in temporal graphs.

Anomaly Detection on Dynamic Heterogeneous Graphs. The state-of-the-art (SOTA) method THGNN [17] utilizes heterogeneous encoders and dual-level distributive attention mechanisms for anomalous behavior (edge) detection in dynamic heterogeneous graphs. However, the state-of-the-art method does not fully capture the relation-aware structural evolution during anomaly detection.

3 Preliminary

Definition 1 (Dynamic Heterogeneous Graphs [17]). *It is defined as $G(\mathcal{V},\mathcal{E})$, where $\tau:\mathcal{V}\rightarrow\mathcal{A}$ maps $\mathcal{V}$ to node types and $\phi:\mathcal{E}\rightarrow\mathcal{R}$ maps $\mathcal{E}$ to edge types (relations), with $|\mathcal{A}|>1$ or $|\mathcal{R}|>1$. The edge set $\mathcal{E}$ is an evolving stream of edges $\mathcal{E}=\langle e_1,e_2,\cdots,e_{|\mathcal{E}|}\rangle$, where each edge $e\in\mathcal{E}$ is represented by a 5-tuple (v^s,r,v^g,a_e,t). Here, $v^s\in\mathcal{V}$ and $v^g\in\mathcal{V}$ denote source node and target node, r is the edge type, a_e is the edge attribute (optional), and t is the time of the edge e. Each edge type $r\in\mathcal{R}$ has a reverse edge type $r^{-1}\in\mathcal{R}$, with $r=r^{-1}$ for symmetric edges.*

Table 1: Summary of Notations

Notations	Descriptions
$G(\mathcal{V},\mathcal{E})$	A dynamic heterogeneous graph.
$\mathcal{A}$	The set of node types in the graph.
$\mathcal{R}$	The set of edge types in the graph.
$\mathbf{x}(v_i^t)\in\mathbb{R}^{d_v}$	Embedding representation of node v_i at time t.
$\mathbf{x}'_{\phi(e)}(v_i^t)\in\mathbb{R}^{d_v}$	Embedding representation of node v_i at time t under the relation $\phi(e)$.
$\mathbf{m}_{r^t}\in\mathbb{R}^{d_v}$	Embedding of the relational subgraph at time t for relation r.
$\mathbf{h}_t\in\mathbb{R}^{d_u}$	Embedding of the change trends for the given relation subgraph.
$\mathbf{H}'\in\mathbb{R}^{\lvert\mathcal{R}\rvert\times d_u}$	Embedding of the change trends for all relation subgraphs.
$\mathbf{m}_p\in\mathbb{R}^{d_p}$	Relation-aware edge representation.
$\mathbf{h}_{re}\in\mathbb{R}^{d_p}$	Representation of the dynamic structural evolution of edge e under relation r.
$\hat{y}_e$	The abnormality score for edge e.

Problem Definition: Anomalous Behavior Detection in Dynamic Heterogeneous Graphs [17]. We frame the task of detecting anomalous behavior as identifying anomalous edges within dynamic heterogeneous graphs, as defined by [17]. Specifically, the goal is to identify the subset of anomalous edges $\mathcal{E}'\subseteq\mathcal{E}$ in $G(\mathcal{V},\mathcal{E})$. We summarise the notations in Table 1.

4 The ReadGraph Model

Framework. Figure 2 illustrates the framework. It extracts edge-centric heterogeneous subgraphs (module (a)) and encodes relation subgraphs using relation-specific message passing (module (b)). Sequential learning captures relation-aware structural evolution (module (c)).

Edge-Centric Heterogeneous Subgraph Sampling. The idea is that subgraphs around a particular edge tend to have a more direct and localized effect on that edge than the overall structure of the entire graph. This localized view is essential for capturing the specific interactions and dependencies that influence the edge, rather than just considering the broader network. To better capture this influence, for each edge, we use three sampling strategies to extract its heterogeneous subgraphs for each snapshot of the dynamic graph: meta-path, k-hop neighborhoods, and node-importance-based sampling. For a target edge $e=(v_{s_1},v_{s_2})$, meta-path-based sampling explores nodes connected to both v_{s_1} and v_{s_2} through specific meta-paths, while node-importance-based sampling uses importance scores calculated by Personalized PageRank [20] as:

$$S=\alpha(\mathbf{I}-(1-\alpha)\mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2})^{-1}, \tag{1}$$

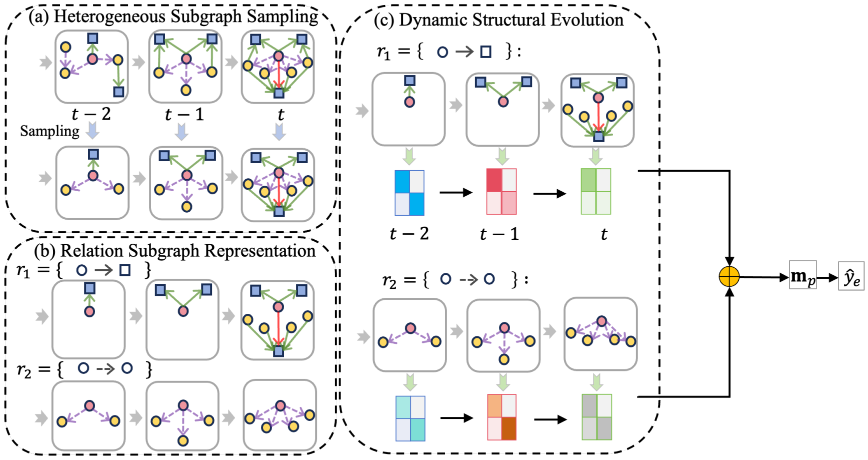

Figure 2: The overall framework of our proposed ReadGraph model.

where $\mathbf{A} \in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|}$ denotes the adjacency matrix which represents the edge count over time, $\mathbf{D} \in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|}$ is the diagonal degree matrix, $\mathbf{S} \in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|}$, $\alpha \in (0,1)$ denotes the teleport probability.

In the heterogeneous subgraph induced by the node set sampled using those three strategies, the node embedding is initially by the attention function of its importance rank, spatial distance to the target edge $e = (v_{s_1}, v_{s_2})$, temporal distance, and node type information as:

$$\begin{aligned} \mathbf{x}(v_i^t) &= \mathrm{att}(\mathbf{x}_{\mathrm{imp}}(v_i^t)||\mathbf{x}_{\mathrm{spt}}(v_i^t)||\mathbf{x}_{\mathrm{tmp}}(v_i^t)||\mathbf{x}_{\mathrm{typ}}(v_i^t)) \\ \mathbf{x}(v_i^t) &= \mathrm{pooling}(\mathbf{x}(v_i^t)), \end{aligned} \tag{2}$$

where the importance rank is $\mathbf{x}_{\mathrm{imp}}(v_i^t) = \mathrm{linear}(\mathrm{rank}(s_{i,s_1}^t + s_{i,s_2}^t))$, spatial distance is $\mathbf{x}_{\mathrm{spt}}(v_i^t) = \mathrm{linear}(\min(\mathrm{dist}(v_i^t, v_{s_1}^t), \mathrm{dist}(v_i^t, v_{s_2}^t)))$, temporal distance between the occurring time t and the current time t_o is $\mathbf{x}_{\mathrm{tmp}}(v_i^t) = \mathrm{linear}(||t_o - t||)$, and the node type is $\mathbf{x}_{\mathrm{typ}}(v_i^t) = \mathrm{linear}(a_{v_i^t})$, $a_{v_i^t}$ is the type of node v_i^t; $\mathrm{att}(\cdot) = \mathrm{softmax}(\frac{\mathbf{Q_v}\mathbf{K_v}^\top}{\sqrt{d_v}})\mathbf{V_v}$, where $\mathbf{Q_v} \in \mathbb{R}^{4\times d_v}$, $\mathbf{K_v} \in \mathbb{R}^{4\times d_v}$, and $\mathbf{V_v} \in \mathbb{R}^{4\times d_v}$ are calculated by multiplying learnable parameters with the concatenated vector, and d_v is the hidden dimension. Then we perform a pooling operation to obtain the final node representation $\mathbf{x}(v_i^t) \in \mathbb{R}^{d_v}$.

Relation Subgraph Representation Learning. To better capture the varying representations of the same heterogeneous subgraph under different relational contexts, we introduce a Relation Subgraph Representation Learning module. This module is designed to explicitly account for the different types of relations that can influence the subgraph's structure and meaning. For each edge-centric heterogeneous subgraph, we first construct its corresponding relation subgraphs, such as those in Figure 2 (b) with relations r_1 (solid arrow) and r_2 (dashed arrow). Nodes connected by each relation type are used to create relation subgraphs.

We propagate messages within each relation subgraph as follows:

$$\begin{aligned} \mathbf{x}'_{\phi(e)}(v_i^t) &= \sigma\Big(\sum_{j\in\mathcal{N}_{\phi(e)}(v_i^t)} g_{ij}^{\phi(e)} \cdot \mathbf{x}_{\phi(e)}(v_j^t)\Big) \\ g_{ij}^{\phi(e)} &= \mathrm{softmax}\big(w_{\phi(e)}^\top \cdot [\mathbf{x}_{\phi(e)}(v_i^t)||\mathbf{x}_{\phi(e)}(v_j^t)]\big), \end{aligned} \tag{3}$$

where $\mathbf{x}_{\phi(e)} \in \mathbb{R}^{d_v}$, $w_{\phi(e)} \in \mathbb{R}^{2d_v}$, and $\mathcal{N}_{\phi(e)}(v_i^t)$ denotes the neighboring nodes of the node v_i^t for relation $\phi(e)$.

Next, relation-level messages are fused across subgraphs:

$$\mathbf{x}'(v_i^t) = \sigma\Big(\sum_{r\in\mathcal{R}} \mathrm{softmax}\big(\mathbf{q}_{\phi(e)}^\top \mathbf{x}'_r(v_i^t)\big) \cdot \mathbf{x}'_r(v_i^t)\Big), \tag{4}$$

where $\mathbf{q}_{\phi(e)} \in \mathbb{R}^{d_v}$ denotes trainable parameter.

Finally, the representation $\mathbf{m}_{r^t} \in \mathbb{R}^{d_v}$ of the relation subgraph is calculated by average pooling among embeddings $\mathbf{x}'(v_i^t) \in \mathbb{R}^{d_v}$ of all nodes in the relation subgraph.

Relation-Aware Structural Evolution. In order to capture the relation-aware dynamic structural evolution over time, here we use a sequential learning algorithm, whose fewer parameters allow for more efficient training, especially when dealing with large-scale dynamic graphs. For different relation subgraphs across snapshots, we use Gated Recurrent Units (GRUs) [4] as:

$$\begin{aligned} \mathbf{h}_t &= \mathbf{z}_t \circ \mathbf{h}_{t-1} + (1-\mathbf{z}_t) \circ \bar{\mathbf{h}}_t \\ \mathbf{z}_t &= \sigma(\mathbf{W}_z \mathbf{m}_{r^t} + \mathbf{U}_z \mathbf{h}_{t-1}) \\ \bar{\mathbf{h}}_t &= \tanh(\mathbf{W}_h \mathbf{m}_{r^t} + \mathbf{U}_h(\mathbf{c}_t \circ \mathbf{h}_{t-1})) \\ \mathbf{c}_t &= \sigma(\mathbf{W}_r \mathbf{m}_{r^t} + \mathbf{U}_r \mathbf{h}_{t-1}), \end{aligned} \tag{5}$$

where $\circ$ denotes element-wise product operation, $\mathbf{W}_z, \mathbf{W}_h, \mathbf{W}_r \in \mathbb{R}^{d_u\times d_v}$, $\mathbf{U}_z, \mathbf{U}_h, \mathbf{U}_r \in \mathbb{R}^{d_u\times d_u}$ are learnable parameters, and d_u denotes the hidden unit dimension. The GRU length is related to the number of snapshots $|\mathcal{T}|$, which is typically short.

The output of the last snapshot $\mathbf{h}_t \in \mathbb{R}^{d_u}$ is utilized to encode the change trends of the relation subgraph over time. For all relation subgraphs, we form their sequential outputs as $\mathbf{H}' \in \mathbb{R}^{|\mathcal{R}|\times d_u}$.Then, We can obtain the relation-aware edge representation after a linear transformation of $\mathbf{H}'$, e.g., $\mathbf{m}_p = \mathrm{linear}(\mathbf{H}')$.

Anomaly Prediction. The predicted anomaly score $\hat{y}_e$ of the target edge e is calculated as $\hat{y}_e = \sigma(\mathrm{linear}(\mathbf{m}_p))$, where $0 \le \hat{y}_e \le 1$. A higher $\hat{y}_e$ indicates a greater abnormality and σ is the sigmoid activation function.

Optimization. We optimize this problem by minimizing the loss function as follows:

$$\mathcal{L} = \mathcal{L}_{\mathrm{reg}} + \mathcal{L}_{\mathrm{ent}} \tag{6}$$

where $\mathcal{L}_{\mathrm{reg}} = ||\hat{\mathbf{A}}_{s_1 s_2}^t - (1-\hat{y}_e^t)||$, $\mathcal{L}_{\mathrm{ent}} = y_e \log(\hat{y}_e) + (1-y_e)\log(1-\hat{y}_e)$. The regularization loss $\mathcal{L}_{\mathrm{reg}}$ aligns the node connectivity in

Table 2: AUC values across three datasets, with bold indicating the highest values and underline marking the second highest.

Datasets	Anomaly Ratio	HAN	RGCN	TADDY	MIDAS	RustGraph	THGNN	ReadGraph
Digg	5%	0.5996	0.7406	0.7018	0.2614	<u>0.7734</u>	0.7073	**0.7952**
	10%	0.5339	0.7587	0.7222	0.2802	<u>0.7842</u>	0.7084	**0.8012**
Yelp	5%	0.5033	**0.6162**	0.5178	0.5475	0.5179	0.5447	<u>0.5683</u>
	10.27%	0.5048	<u>0.5669</u>	0.5163	0.5445	0.5146	0.5628	**0.5909**
Amazon	5%	0.6069	0.5775	0.5004	0.4949	<u>0.6333</u>	0.3908	**0.7596**
	10%	0.6361	0.5945	0.5005	0.4879	<u>0.6448</u>	0.5655	**0.7501**

the reconstructed adjacency matrix $\hat{\mathbf{A}} = \sigma(\mathbf{X}^t \mathbf{X}^{t\top})$ with the value $1 - \hat{y}_e$. For normal edges, where $\hat{y}_e$ is 0, this value equals 1.

THEOREM 4.1. *The complexity of ReadGraph is dominated by* $O\big(|\mathcal{E}|(|\mathcal{V}_h| + |\mathcal{E}_h|)\big)$*, where* $|\mathcal{V}_h|$ *and* $|\mathcal{E}_h|$ *denote the number of nodes and edges in the sampled subgraph for target edge* e.

5 Experiments

<u>**Experimental Setup.**</u> ***Datasets:*** We evaluate the models on datasets from three real-world platforms as in Table 3.

Table 3: The statistics of three datasets.

	Digg	Yelp	Amazon
# Nodes	3,532,340	161,148	9,084,722
# Edges	4,748,185	359,052	34,686,770
% Anomalous Edges	10%	10.27%	10%
# Node Types	2	2	2
# Edge Types	4	2	2

• Digg Dataset [17]: It includes users and stories, with nodes denoting both users and stories. Edges between users indicate the 'following' relations, while edges between users and stories represent 'voting' relations. The meta-paths are "USU", and "UU", where 'U' denotes a user and 'S' denotes a story. The relations include 'following' and 'voting', along with their respective inverse relations.

• Yelp Dataset [1]: It captures user reviews for various hotels. Nodes represent users and hotels, and directed edges reflect user reviews of hotels. Each edge is labeled as either normal or abnormal. The meta-path selected is "UHU", with 'U' indicating a user and 'H' indicating a hotel. The relations include 'review' and its inverse. This dataset contains 10.27% of edges that are labeled as anomalous.

• Amazon Dataset [2]: It contains user reviews of items on Amazon. Nodes represent users and items, while directed edges denote user reviews of items. The meta-path is "UIU", where 'U' denotes a user and 'I' denotes an item. The relations are 'review' and its inverse.

Preprocessing: Since the Digg and Amazon datasets lack labels for anomalous edges, we follow [17] and randomly assign anomalous labels to 10% of the edges in the training set. The remaining edges are treated as normal. The Yelp dataset already contains labeled anomalous edges, so no injection was needed. For evaluation, we retain the original anomalies in the training set and set the anomaly percentage to 5% in the testing set by randomly removing some anomalous edges in the Yelp dataset as [17].

Competitors: We compare ReadGraph with **HAN** [29] and **RGCN** [27] designed for **static heterogeneous graphs**, **TADDY** [20], **MIDAS** [2] and **RustGraph** [9] designed for **dynamic homogeneous graphs**, and **THGNN** [17] (**SOTA**) designed for **dynamic heterogeneous graphs**. More details can be found in Section 2.

Hyperparameter Settings: Hyperparameter tuning is performed via Bayesian optimization with the following ranges: snapshot size $|\mathcal{E}_s|$ {500, 1000, 1500, 2000, 2500}, which refers to the number of edges used to construct a snapshot as [9, 20], number of nodes $|\mathcal{V}_h|$ in the sampled heterogeneous graphs {5, 7, 9, 11, 13, 15}, and the number of snapshots $|\mathcal{T}|$ {2, 3, 4, 5, 6}.

Performance Metrics: We use AUC (area under the ROC curve) as the performance metric, following [17]. AUC ranges from 0 to 1, with higher values indicating better performance. This metric is effective for anomaly detection, as it measures accuracy even when the number of anomalies is unknown.

<u>**Effectiveness Evaluation.**</u> Table 2 shows the effectiveness results of all models. Our observations are as follows:

(1) The ReadGraph model consistently achieves the highest AUC values on the Digg, Yelp, and Amazon datasets when the percentage of anomalous edges is 10% or 10.27%. In particular, on the real-world Yelp dataset with labeled anomalies, the ReadGraph model outperforms all other models by 4.15%, highlighting its superior anomaly detection capability. It can be seen that capturing the dynamic structural evolution of relationships in heterogeneous graphs plays an important role.

(2) When the anomaly ratio is reduced to 5%, the ReadGraph model maintains the best and most stable performance on the Digg and Amazon datasets. On the Yelp dataset, it achieves the second-best AUC value.

(3) Among the baseline models in the first category, RGCN demonstrates the best performance on both Digg and Yelp datasets. It even achieves the highest AUC among all models for the Yelp dataset when the anomalous edge percentage is 5%. This underscores the effectiveness of relational learning in detecting anomalous edges and its critical role in edge anomaly detection.

<u>**Ablation Study.**</u> We conduct an ablation study to assess the impact of key components in ReadGraph. The first variant ignores the meta-path enhanced modeling, denoted as $\text{ReadGraph}_{\text{nm}}$. The second variant retains only the cross-entropy loss, denoted as $\text{ReadGraph}_{\text{nl}}$. The last one ignores the dynamic structural evolution, denoted as $\text{ReadGraph}_{\text{ne}}$. From Figure 3, we observe:

(1) ReadGraph is significantly more effective than $\text{ReadGraph}_{\text{nm}}$ on both datasets. It shows that the modeling of the meta-paths is necessary in the model.

(2) ReadGraph shows superior performance compared to $\text{ReadGraph}_{\text{nl}}$ on the Yelp dataset and performs comparably to $\text{ReadGraph}_{\text{nl}}$ on the Amazon dataset. This indicates that including additional losses enhances anomaly detection.

[1]https://www.kaggle.com/datasets/abidmeeraj/yelp-labelled-dataset/code
[2]https://snap.stanford.edu/data/web-Amazon-links.html

(3) ReadGraph has better performance than ReadGraph$_{\text{ne}}$ and has a significant effect on the Yelp dataset. It shows that the dynamic structural evolution plays an important role.

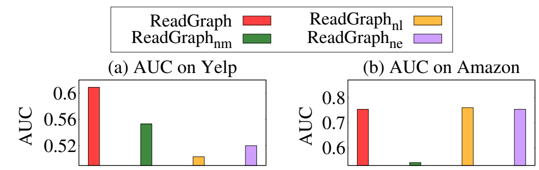

Figure 3: The ablation experiments of the ReadGraph model.

6 Conclusions

In this paper, we address the challenge in anomalous behavior (edge) detection for dynamic heterogeneous graphs. We introduce ReadGraph, which is a novel model that tracks relation-aware structural evolution to thoroughly capture the features associated with behaviors. Our evaluation results indicate that our proposed model outperforms SOTA w.r.t. effectiveness.

Acknowledgement

Xiaolin Han and Xiurui Hu were supported by NSFC under Grant 62302397, the fund of Laboratory for Advanced Computing and Intelligence Engineering (No. 2023-LYJJ-01-021) and Fundamental Research Funds for the Central Universities, China under Grant No. D5000230191. Chenhao Ma was partially supported by NSFC under Grant 62302421, Basic and Applied Basic Research Fund in Guangdong Province under Grant 2023A1515011280, Ant Group through CCF-Ant Research Fund, Shenzhen Research Institute of Big Data under grant SIF20240004, and the Guangdong Provincial Key Laboratory of Big Data Computing, The Chinese University of Hong Kong, Shenzhen.

References

[1] Leman Akoglu, Hanghang Tong, and Danai Koutra. 2015. Graph based anomaly detection and description: a survey. *DMKD* 29 (2015), 626–688.

[2] Siddharth Bhatia, Rui Liu, Bryan Hooi, et al. 2022. Real-time anomaly detection in edge streams. *TKDD* 16, 4 (2022), 1–22.

[3] Siddharth Bhatia, Mohit Wadhwa, Kenji Kawaguchi, Neil Shah, Philip S Yu, et al. 2023. Sketch-based anomaly detection in streaming graphs. In *SIGKDD*.

[4] Junyoung Chung et al. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. In *NIPS Workshop on Deep Learning*.

[5] Dhivya Eswaran, Srijan Kumar, and Christos Faloutsos. 2020. Higher-order label homogeneity and spreading in graphs. In *Proceedings of The Web Conference 2020*. 2493–2499.

[6] Yang Fang, Xiang Zhao, Peixin Huang, Weidong Xiao, and Maarten de Rijke. 2022. Scalable representation learning for dynamic heterogeneous information networks via metagraphs. *ACM Transactions on Information Systems (TOIS)* 40, 4 (2022), 1–27.

[7] Chaofan Fu, Guanjie Zheng, Chao Huang, Yanwei Yu, and Junyu Dong. 2023. Multiplex heterogeneous graph neural network with behavior pattern modeling. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 482–494.

[8] Xinyu Fu, Jiani Zhang, Ziqiao Meng, and Irwin King. 2020. Magnn: Metapath aggregated graph neural network for heterogeneous graph embedding. In *Proceedings of the web conference 2020*. 2331–2341.

[9] Jianhao Guo et al. 2023. RustGraph: Robust Anomaly Detection in Dynamic Graphs by Jointly Learning Structural-Temporal Dependency. *TKDE* (2023).

[10] Shengnan Guo, Youfang Lin, Huaiyu Wan, Xiucheng Li, and Gao Cong. 2021. Learning dynamics and heterogeneity of spatial-temporal graph data for traffic forecasting. *IEEE Transactions on Knowledge and Data Engineering* 34, 11 (2021), 5415–5428.

[11] Xiaolin Han, Reynold Cheng, Tobias Grubenmann, Silviu Maniu, Chenhao Ma, and Xiaodong Li. 2022. Leveraging contextual graphs for stochastic weight completion in sparse road networks. In *Proceedings of the 2022 SIAM International Conference on Data Mining (SDM)*. SIAM, 64–72.

[12] Xiaolin Han, Reynold Cheng, Chenhao Ma, and Tobias Grubenmann. 2022. DeepTEA: Effective and efficient online time-dependent trajectory outlier detection. *Proceedings of the VLDB Endowment* 15, 7 (2022), 1493–1505.

[13] Xiaolin Han, Tobias Grubenmann, Reynold Cheng, Sze Chun Wong, Xiaodong Li, and Wenya Sun. 2020. Traffic incident detection: A trajectory-based approach. In *2020 IEEE 36th international conference on data engineering (ICDE)*. IEEE, 1866–1869.

[14] Xiaolin Han, Tobias Grubenmann, Chenhao Ma, Xiaodong Li, Wenya Sun, Sze Chun Wong, Xuequn Shang, and Reynold Cheng. 2024. FDM: Effective and efficient incident detection on sparse trajectory data. *Information Systems* (2024), 102418.

[15] Ziniu Hu, Yuxiao Dong, Kuansan Wang, and Yizhou Sun. 2020. Heterogeneous graph transformer. In *Proceedings of the web conference 2020*. 2704–2710.

[16] Yongcheng Jing, Yiding Yang, Xinchao Wang, Mingli Song, and Dacheng Tao. 2021. Amalgamating knowledge from heterogeneous graph neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 15709–15718.

[17] Yilin Li, Jiaqi Zhu, et al. 2023. THGNN: An Embedding-based Model for Anomaly Detection in Dynamic Heterogeneous Social Networks. In *CIKM*.

[18] Erez Lieberman, Christoph Hauert, and Martin A Nowak. 2005. Evolutionary dynamics on graphs. *Nature* 433, 7023 (2005), 312–316.

[19] Hu Linmei, Tianchi Yang, Chuan Shi, et al. 2019. Heterogeneous graph attention networks for semi-supervised short text classification. In *EMNLP-IJCNLP*.

[20] Yixin Liu, Shirui Pan, Yu Guang Wang, Fei Xiong, Liang Wang, et al. 2021. Anomaly detection in dynamic graphs via transformer. *TKDE* (2021).

[21] Wenjuan Luo, Han Zhang, Xiaodi Yang, Lin Bo, Xiaoqing Yang, Zang Li, Xiaohu Qie, and Jieping Ye. 2020. Dynamic heterogeneous graph neural network for real-time event prediction. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*. 3213–3223.

[22] Chenhao Ma, Reynold Cheng, Laks VS Lakshmanan, and Xiaolin Han. 2022. Finding Locally Densest Subgraphs: A Convex Programming Approach. *Proceedings of the VLDB Endowment* 15, 11.

[23] Chenhao Ma, Yixiang Fang, Reynold Cheng, Laks VS Lakshmanan, and Xiaolin Han. 2022. A Convex-Programming Approach for Efficient Directed Densest Subgraph Discovery. In *Proceedings of the 2022 International Conference on Management of Data*. 845–859.

[24] Chenhao Ma, Yixiang Fang, Reynold Cheng, Laks VS Lakshmanan, Xiaolin Han, and Xiaodong Li. 2024. Accelerating directed densest subgraph queries with software and hardware approaches. *The VLDB Journal* 33, 1 (2024), 207–230.

[25] Chenhao Ma, Yixiang Fang, Reynold Cheng, Laks VS Lakshmanan, Wenjie Zhang, and Xuemin Lin. 2020. Efficient algorithms for densest subgraph discovery on large directed graphs. In *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*. 1051–1066.

[26] Mariam Orabi, Djedjiga Mouheb, Zaher Al Aghbari, and Ibrahim Kamel. 2020. Detection of bots in social media: a systematic review. *IPM* (2020).

[27] Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, et al. 2018. Modeling relational data with graph convolutional networks. In *ESWC*.

[28] Hao Tang, Zhiao Huang, Jiayuan Gu, Bao-Liang Lu, and Hao Su. 2020. Towards scale-invariant graph-related problem solving by iterative homogeneous gnns. *Advances in Neural Information Processing Systems* 33 (2020), 15811–15822.

[29] Xiao Wang, Houye Ji, Chuan Shi, Bai Wang, Yanfang Ye, Peng Cui, and Philip S Yu. 2019. Heterogeneous graph attention network. In *WWW*. 2022–2032.

[30] Yongtian Wang, Xinmeng Liu, Yewei Shen, Xuerui Song, Tao Wang, Xuequn Shang, and Jiajie Peng. 2023. Collaborative deep learning improves disease-related circRNA prediction based on multi-source functional information. *Briefings in bioinformatics* 24, 2 (2023), bbad069.

[31] Yongtian Wang, Wenkai Shen, Yewei Shen, Shang Feng, Tao Wang, Xuequn Shang, and Jiajie Peng. 2024. Integrative graph-based framework for predicting circRNA drug resistance using disease contextualization and deep learning. *IEEE Journal of Biomedical and Health Informatics* (2024).

[32] Qianlong Wen, Zhongyu Ouyang, Jianfei Zhang, Yiyue Qian, Yanfang Ye, and Chuxu Zhang. 2022. Disentangled dynamic heterogeneous graph learning for opioid overdose prediction. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2009–2019.

[33] Xiaocheng Yang, Mingyu Yan, Shirui Pan, Xiaochun Ye, and Dongrui Fan. 2023. Simple and efficient heterogeneous graph neural network. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 37. 10816–10824.

[34] Zeyang Zhang, Ziwei Zhang, Wang, et al. 2023. Dynamic heterogeneous graph attention neural architecture search. In *AAAI*, Vol. 37. 11307–11315.

[35] Jianan Zhao, Xiao Wang, Chuan Shi, Binbin Hu, Guojie Song, and Yanfang Ye. 2021. Heterogeneous graph structure learning for graph neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 35. 4697–4705.