SAD: Semi-Supervised Anomaly Detection on Dynamic Graphs

Sheng Tian[1*], **Jihai Dong**[1*], **Jintang Li**[2], **Wenlong Zhao**[1], **Xiaolong Xu**[1], **Baokun Wang**[1], **Bowen Song**[1], **Changhua Meng**[1], **Tianyi Zhang**[1] and **Liang Chen**[2]

[1]Ant Group

[2]Sun Yat-sen University

{tiansheng.ts, dongjihai.djh}@antgroup.com, lijt55@mail2.sysu.edu.cn, {chicheng.zwl, yiyin.xxl, yike.wbk, bowen.sbw, changhua.mch, zty113091}@antgroup.com, chenliang6@mail.sysu.edu.cn

IJCAI 2023

Code&data： https://github.com/D10Andy/SAD.

Reported by Nengqiang Xiang

Introduction

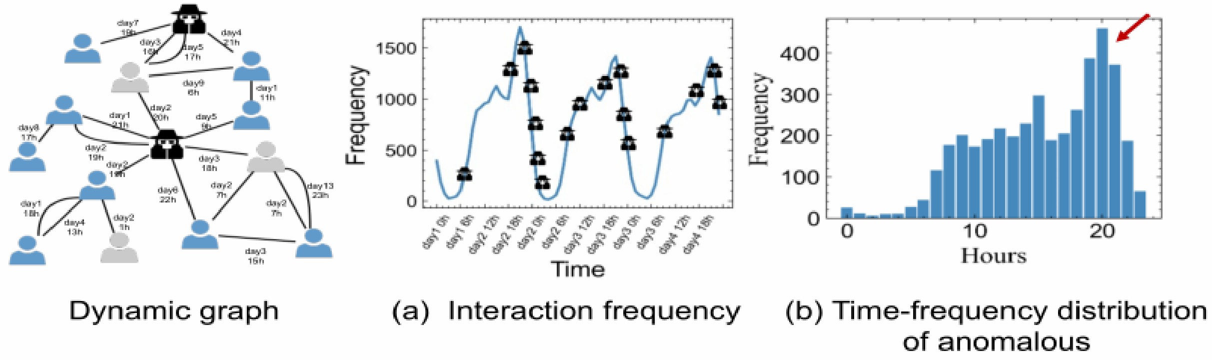


Figure 1: Statistical distribution of evolving graph streams on the MOOC dataset. **Observations**: (a) The frequency of interactions shows a clear cyclical pattern on the time axis. (b) Anomalous samples typically occurred between 8 am and 10 pm, with peaks at around 8 pm.

Existing methods are two limitations：Underutilization of unlabeled data; Lack of dynamic information mining.

In this paper, the author presents semi-supervised anomaly detection (SAD), an end-to-end framework for anomaly detection on dynamic graphs.

Method

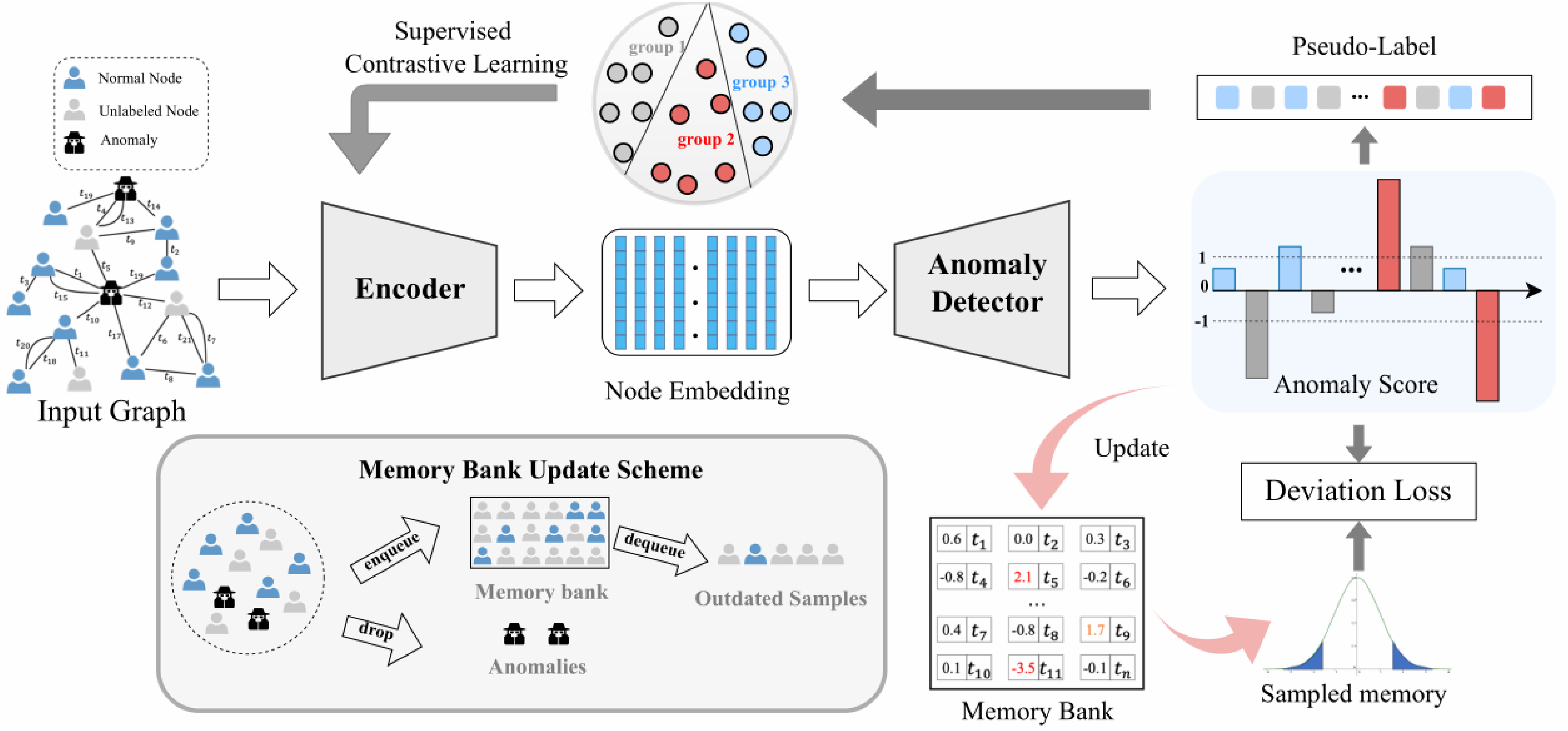
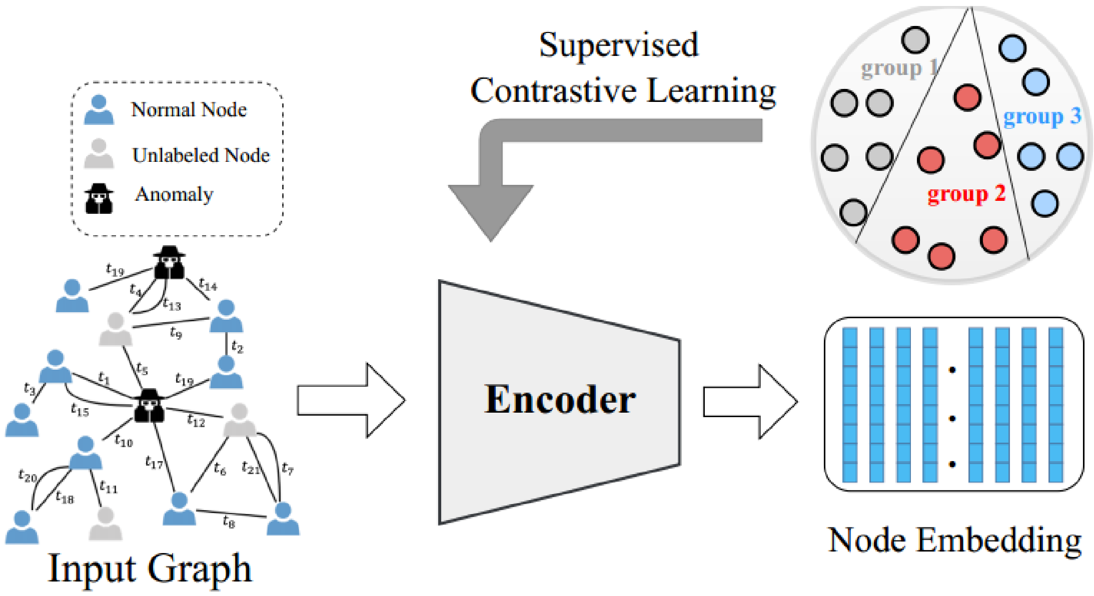


Figure 2: The proposed semi-supervised anomaly detection framework. SAD consists of four main components: the temporal graph encoder, the anomaly detector, the time-equipped memory bank, and the supervised contrastive learning module.

Method



PRELIMINARIES:

$$\mathcal{G} = (\mathcal{V}, \mathcal{E})$$

$$\mathcal{E} = \{\delta(t_1), \delta(t_2), \dots, \delta(t_m)\}$$

$$\delta(t) = (v_i, v_j, t, x_{ij})$$

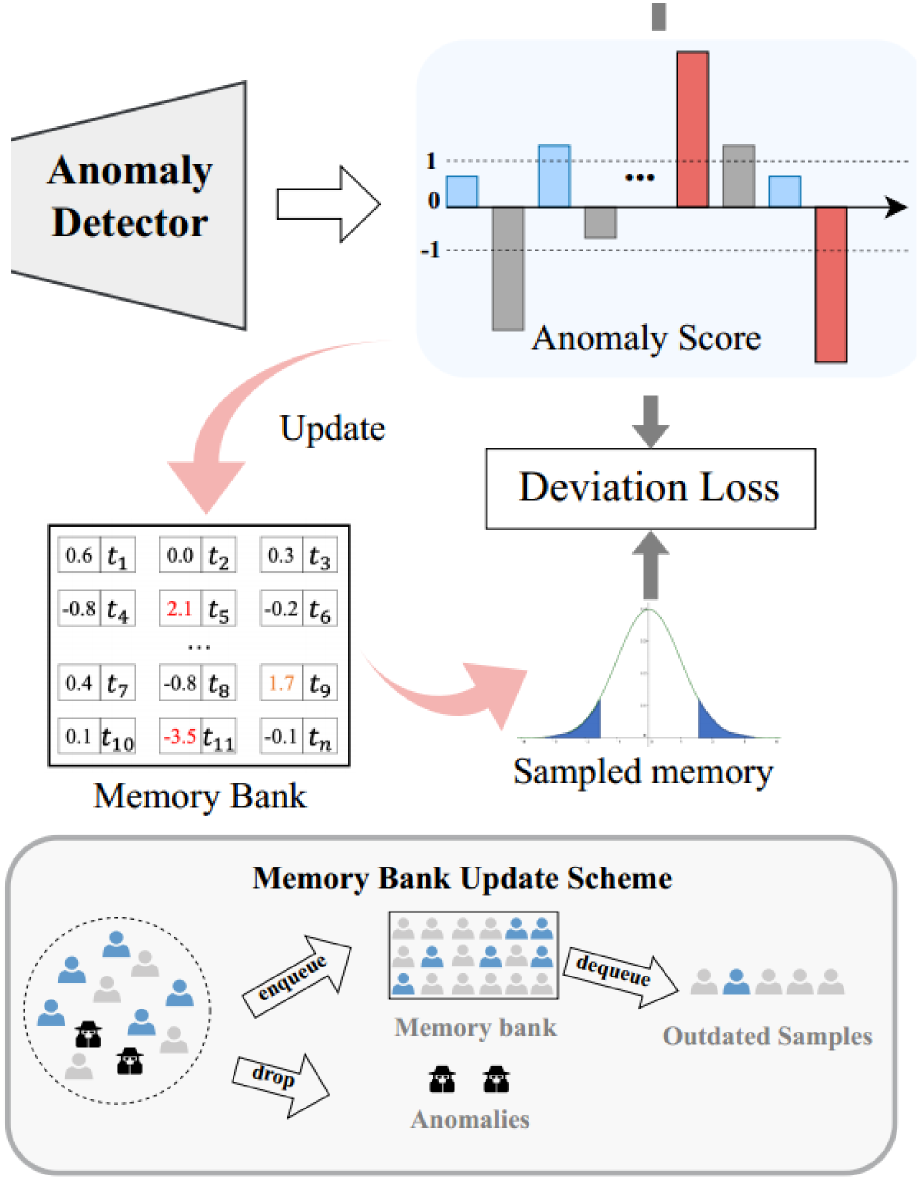
Deviation Networks with Memory Bank:

$$\mathbf{h}_{N_i}^{(k)}(t) = \mathrm{AGG}^{(k)}\left(\left\{\left(\mathbf{h}_j^{(k-1)}(t), x_{ij}, \phi(\Delta t)\right) : v_j \in \mathcal{N}(v_i, t)\right\}\right),$$

$$\mathbf{h}_i^{(k)}(t) = \mathrm{COMBINE}^{(k)}\left(\mathbf{h}_i^{(k-1)}(t), \mathbf{h}_{N_i}^{(k)}(t)\right),$$

$$\Delta t = t - t_{ij} \qquad (1)$$

$$z_i(t) = \mathbf{h}_i^{(K)}(t)$$



Method

Anomaly Detector:

$$s_i(t) = f_{\theta_a}(z_i(t)) = \mathbf{W_2}\cdot\mathrm{ReLU}(\mathbf{W_1}\cdot z_i(t)+\mathbf{b_1})+\mathbf{b_2}, \quad (2)$$

Memory bank:

$$m = s_i(t), \quad \text{if } \ y_i(t) = 0 \text{ or } -1 \qquad (3)$$

$$m = (s_i(t), t), \quad \text{if } \ y_i(t) = 0 \text{ or } -1 \qquad (4)$$

Deviation loss:

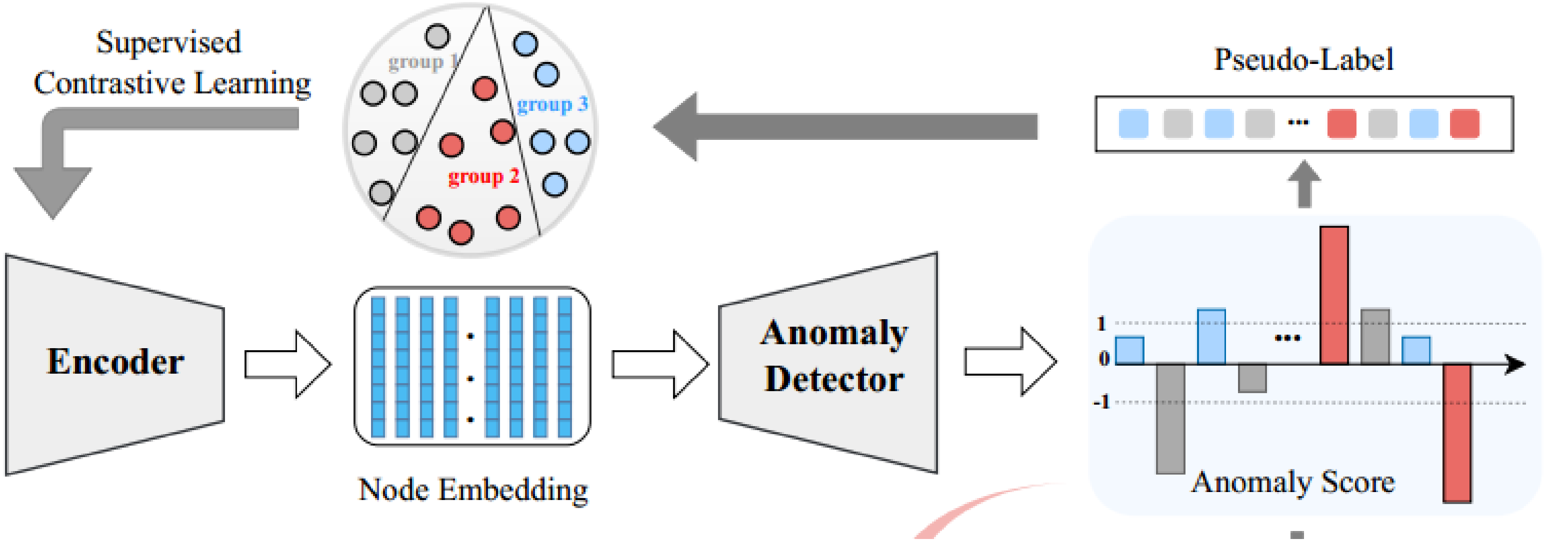
$$\mu_r(t) = \frac{1}{k}\sum_{i=1}^{k} w_i(t)\cdot r_i$$

$$\sigma_r(t) = \sqrt{\frac{\sum_{i=1}^{k} w_i(t)\cdot(r_i-\mu_r(t))^2}{k-1}} \qquad (5)$$

$$dev(v_i, t) = \frac{s_i(t)-\mu_r(t)}{\sigma_r(t)}. \qquad (6)$$

$$\mathcal{L}^{dev} = (1-y_i(t))\cdot|dev(v_i,t)|+y_i(t)\cdot\max(0, m-|dev(v_i,t)|), \qquad (7)$$

Method



Contrastive Learning for Unlabeled Samples:

$$\mathcal{L}_i^{scl} = \frac{-1}{N-1} \sum_{j,j\neq i}^{N} 1_{\Delta d_{ij}<1} \cdot \frac{1}{1+\Delta d_{ij}} \cdot \ln \frac{exp(z_i(t_i)\cdot z_j(t_j)/\tau)}{\sum_{k,i\neq k}^{N} exp(z_i(t_i)\cdot z_k(t_k)/\tau)} \tag{8}$$

$$\Delta d_{ij} = |dev(v_i,t_i) - dev(v_j,t_j)|$$

Learning procedure:

$$\arg\min_{\theta_{enc},\theta_{ano}} \mathcal{L}^{dev}(\theta_{enc},\theta_{ano}) + \alpha\mathcal{L}^{scl}(\theta_{enc}), \tag{9}$$

$$\arg\min_{\theta_{enc},\theta_{ano},\theta_{pro}} \mathcal{L}^{sup}(\theta_{pro}) + \alpha\mathcal{L}^{dev}(\theta_{enc},\theta_{ano}) + \beta\mathcal{L}^{scl}(\theta_{enc}), \tag{10}$$

Experiments

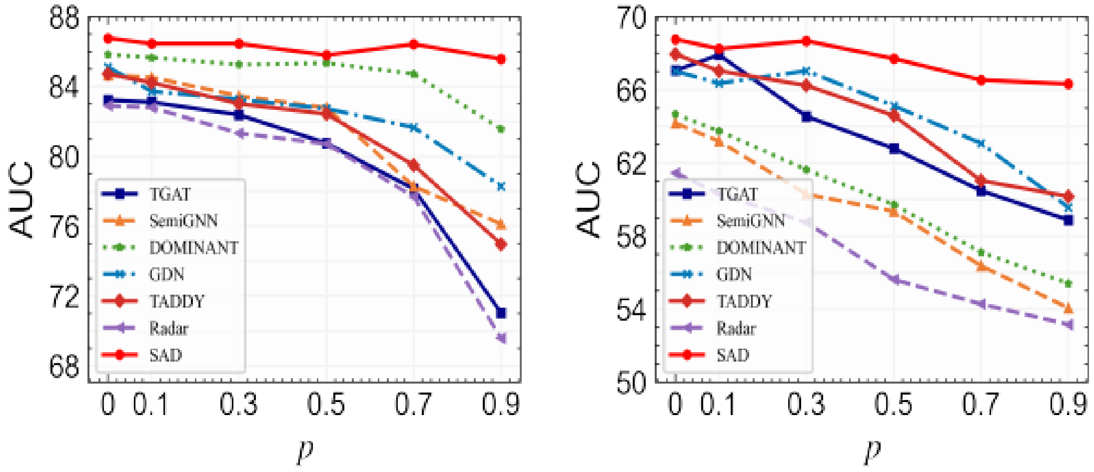
	Wikipedia	Reddit	Mooc	Alipay
#Nodes	9,227	10,984	7,074	3,575,301
#Edges	157,474	672,447	411,749	53,789,768
#Edge features	172	172	4	100
#Anomalies	217	366	4,066	24,979
Timespan	30 days	30 days	30 days	90 days
Pos. label meaning	posting banned	editing banned	dropping out	fraudster
Chronological Split	70%-15%-15%	70%-15%-15%	70%-15%-15%	70%-15%-15%

Table 1: Statistics of datasets.

Methods	Wikipedia	Reddit	Mooc	Alipay
TGAT	83.23 ± 0.84	67.06 ± 0.69	66.88 ± 0.68	92.53 ± 0.93
TGN	84.67 ± 0.36	62.66 ± 0.85	67.07 ± 0.73	92.84 ± 0.81
Radar	82.91 ± 0.97	61.46 ± 1.27	62.14 ± 0.89	88.18 ± 1.05
DOMINANT	85.84 ± 0.63	64.66 ± 1.29	65.41 ± 0.72	91.57 ± 0.93
SemiGNN	84.65 ± 0.82	64.18 ± 0.78	64.98 ± 0.63	92.29 ± 0.85
GDN	85.12 ± 0.69	67.02 ± 0.51	66.21 ± 0.74	93.64 ± 0.79
TADDY	84.72 ± 1.01	67.95 ± 0.94	68.47 ± 0.76	93.15 ± 0.88
SAD	$\mathbf{86.77 \pm 0.24}$	$\mathbf{68.77 \pm 0.75}$	$\mathbf{69.44 \pm 0.87}$	$\mathbf{94.48 \pm 0.65}$

Table 2: Overall performance of all methods in terms of AUC on dynamic node classification tasks. Means and standard deviations were computed over 10 runs.

Experiments



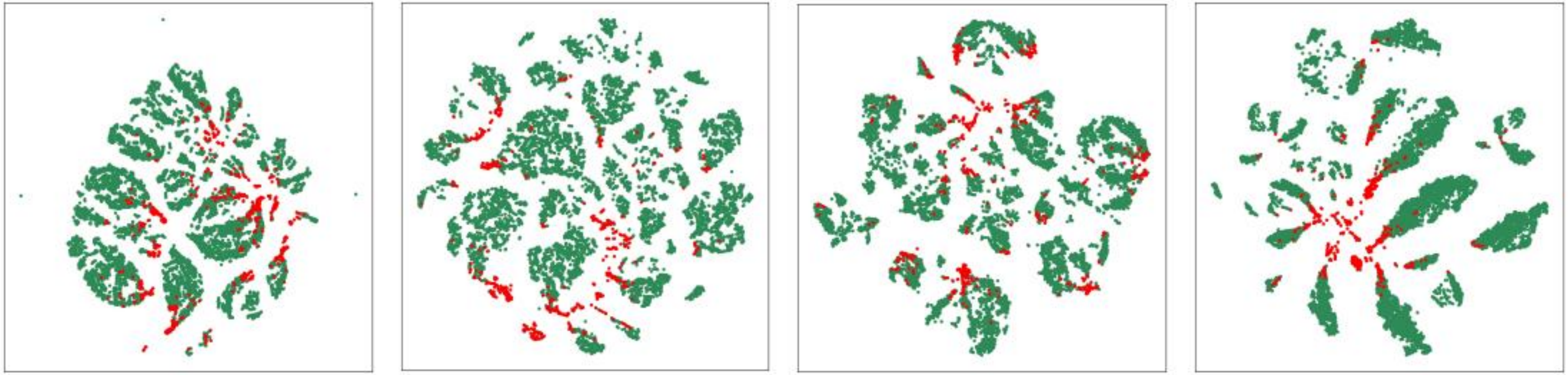
(a) Wikipedia (b) Reddit

Figure 3: Dynamic node classification task results under different drop ratios p on Wikipedia and Reddit, respectively.

	Wikipedia	Reddit	Mooc
TGAT	80.76 ± 2.30	62.79 ± 3.42	64.04 ± 1.02
w/dev	82.45 ± 0.64	64.15 ± 2.93	65.33 ± 1.67
w/mem	85.20 ± 1.30	66.96 ± 1.51	67.25 ± 0.75
w/time	85.44 ± 0.75	66.78 ± 1.98	67.53 ± 0.93
w/scl	$\mathbf{85.80 \pm 1.32}$	$\mathbf{67.71 \pm 0.75}$	$\mathbf{67.57 \pm 0.54}$

Table 3: Results of the ablation study on the dynamic node classification task under the label-dropping ratio of 0.5.

Experiments



(a) TGAT (b) GDN (c) TADDY (d) SAD

Figure 4: Visualization of the learned node embeddings w.r.t. different methods on Alipay. The red and green points represent the abnormal and normal samples, respectively.