GraphSHA: Synthesizing Harder Samples for Class-Imbalanced Node Classification

Wen-Zhi Li [1,2] Chang-Dong Wang [1] Hui Xiong [2,3] Jian-Huang Lai [1]

[1]CSE, Sun Yat-sen University [2]AI Thrust, HKUST (GZ) [3]CSE, HKUST

Research Background

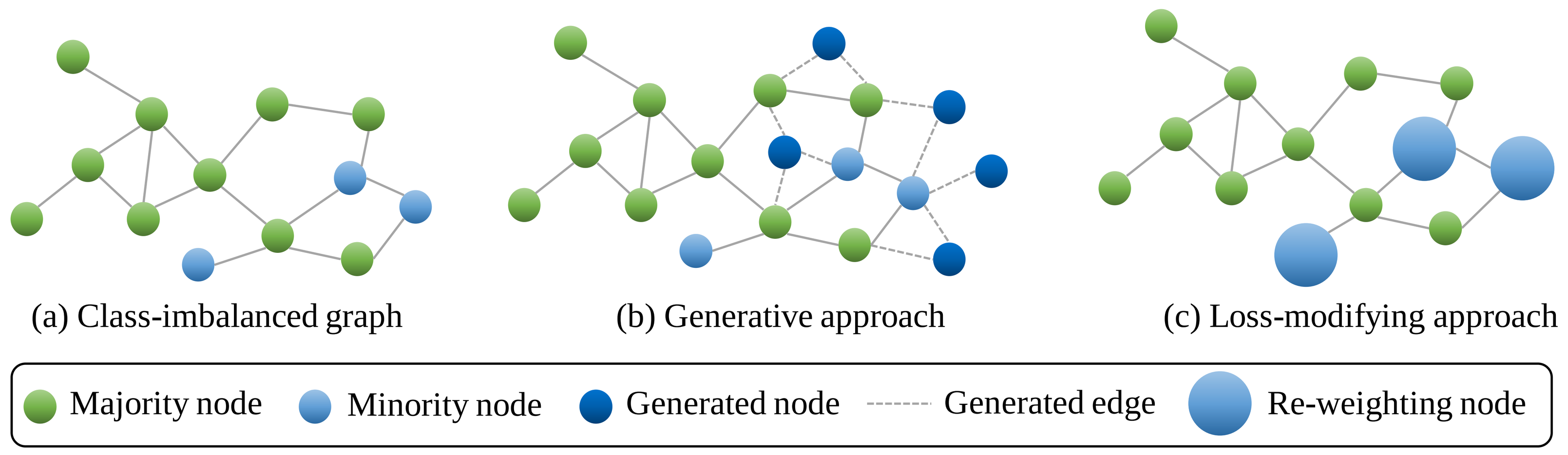


Figure 1. Schematic diagram for class-imbalanced graph and existing methods.

- Graph data in-the-wild tend to be **class-imbalanced** intrinsically.
- Existing methods adapting GNNs to class-imbalanced graphs:
  - **Generative approaches**: augmenting the original class-imbalanced graph by synthesizing plausible minor nodes;
  - **Loss-modifying approaches**: adjusting the objective function to pay more attention to minor class samples.

Empirical Study & Motivation

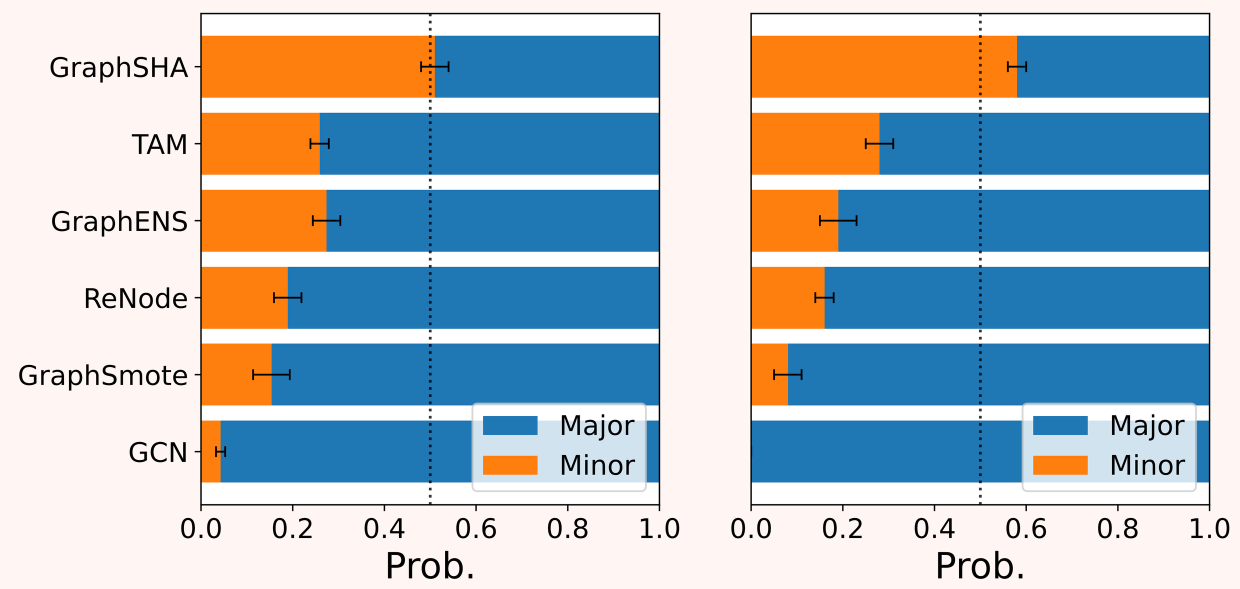


Figure 2. Probability distribution of misclassified samples on Cora-LT and CiteSeer-LT.

- **Squeezed minority problem**: minor subspaces are squeezed by major ones in the latent space.
- **Motivation**: enalrging the minor decision boundary in the latent sapce! 😲
  → Synthesizing harder minor samples beyond the hard minor ones.

Challenges

- The decision boundary is shared by a minor class and its neighbor class. Synthesizing harder minor samples would unavoidably violate the neighbor class subspace.
- A proper augmentation method is required to **enlarge the subspaces of minor classes while avoiding deteriorating those of the neighbor ones.** 😲

Our solution: **GraphSHA** for Synthesizing HArder minor samples. 😃

GraphSHA Overview

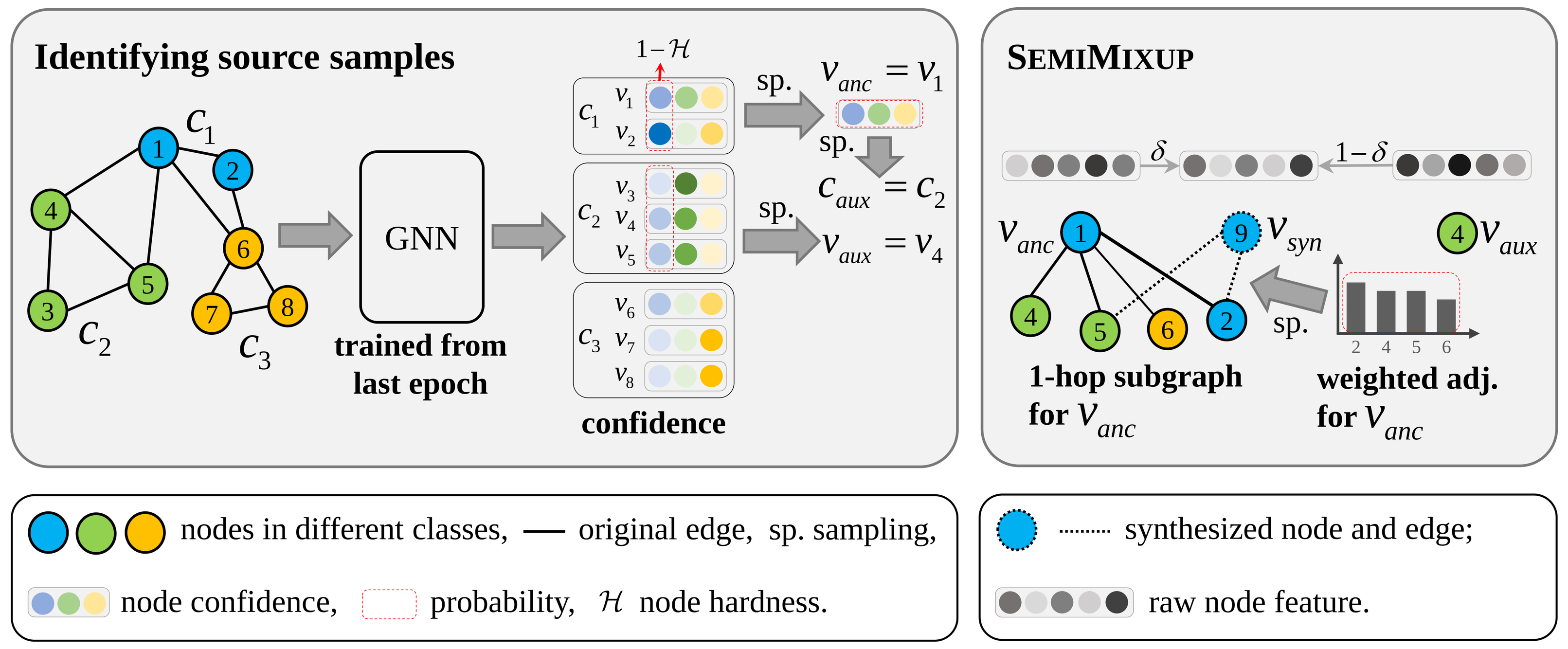


Figure 3. GraphSHA overview where c_1 is minor class and c_2, c_3 are major classes.

- **(Left)**: Identifying two source nodes v_{anc} and v_{aux} via three samplings.
- **(Right)**: Mixuping v_{anc}'s 1-hop subgraph and v_{aux} solely to get v_{syn}.

#1: Identifying Source Samples

Def. (node hardness). $\mathcal{H}_i = 1 - \mathrm{softmax}\left(\boldsymbol{Z}_{i,\boldsymbol{Y}(v_i)}\right)$, where $\boldsymbol{Z}_i = f_\theta(v_i) \in \mathbb{R}^C$.

- Identifying anchor node v_{anc}:
  - Sampling from minor nodes in c_1 according to their hardness $\mathcal{H}$ to get v_{anc}.
- Identifying auxiliary node v_{aux}:
  - Sampling from major classes c_2, c_3 according to v_{anc}'s confidence on them to get neighbor class c_{aux};
  - Sampling from nodes in neighbor class c_{aux} according to their confidences on minor class c_1 to get v_{aux}.

#2: SemiMixup for Harder Sample Synthesis

- Synthesizing node features: a simple mixup between node embeddings of v_{anc} and v_{aux} in the raw feature space

$$\boldsymbol{X}_{syn} = \delta \boldsymbol{X}_{anc} + (1-\delta)\boldsymbol{X}_{aux}, \quad \delta \in [0,1].$$

- Synthesizing edges: enabling propagating information beyond the minor boundary to the interior of the minor class & blocking propagation from the minor class to the neighbor class.
  - Leveraging Graph Diffusion Convolution (GDC) to build the weighted adjacency matrix $\tilde{\boldsymbol{S}}$ based on topology information.
  - Sampling the neighbor set of v_{syn} according to $\tilde{\boldsymbol{S}}_{anc}$. The number of neighbors is sampled from another degree distribution based on the entire graph to keep degree statistics.

Remark. For v_{anc} and v_{aux}, the synthesized harder minor sample is defined as

$$\begin{cases} \boldsymbol{X}_{syn} = \delta \boldsymbol{X}_{anc} + (1-\delta)\boldsymbol{X}_{aux}, \\ \mathcal{N}_{syn} \sim P_{1hop}^{diff}(v_{anc}), \\ \boldsymbol{Y}(v_{syn}) = \boldsymbol{Y}(v_{anc}), \end{cases}$$

where δ is a random variable in $[0,1]$, and $P_{1hop}^{diff}(v_{anc})$ is the 1-hop neighbor distribution of v_{anc} with probability $\tilde{\boldsymbol{S}}_{anc}$ generated via GDC.

Experiments

Table 1. Node classification results in long-tailed class-imbalanced setting.

Dataset	Cora-LT			CiteSeer-LT			PubMed-LT		
ρ=100	Acc.	bAcc.	F1	Acc.	bAcc.	F1	Acc.	bAcc.	F1
Vanilla	72.02±0.50	59.42±0.74	59.23±1.02	51.40±0.44	44.64±0.42	37.82±0.67	51.58±0.60	42.11±0.48	34.73±0.71
Reweight	78.42±0.10	72.66±0.17	73.75±0.15	63.61±0.22	56.80±0.20	55.18±0.18	77.02±0.14	72.45±0.17	72.12±0.15
PC Softmax	77.30±0.13	72.08±0.30	71.65±0.34	62.15±0.45	**59.08**±0.28	58.13±0.31	74.36±0.62	72.59±0.34	71.79±0.50
CB Loss	77.97±0.19	72.70±0.28	73.17±0.22	61.47±0.51	55.18±0.52	53.47±0.65	76.57±0.19	72.16±0.18	72.84±0.19
Focal Loss	78.43±0.19	73.17±0.23	73.76±0.20	59.66±0.38	53.39±0.33	51.80±0.39	75.67±0.20	71.34±0.24	72.03±0.21
ReNode	78.93±0.13	73.13±0.17	74.46±0.16	62.39±0.31	55.62±0.27	54.05±0.24	76.00±0.16	70.68±0.15	71.41±0.15
Upsample	75.52±0.11	66.68±0.14	68.35±0.15	55.05±0.11	48.41±0.11	45.22±0.14	71.58±0.06	63.79±0.06	64.62±0.07
GraphSmote	75.44±0.43	68.99±0.51	70.41±0.52	56.58±0.29	50.39±0.28	47.96±0.33	74.62±0.08	69.53±0.10	71.18±0.09
GraphENS	76.15±0.24	71.16±0.40	70.85±0.49	63.14±0.35	56.92±0.37	55.54±0.41	77.11±0.11	71.89±0.15	72.71±0.14
TAM (G-ENS)	77.30±0.23	72.10±0.29	72.25±0.29	63.40±0.34	57.15±0.35	55.68±0.40	78.07±0.15	72.63±0.23	72.96±0.22
GraphSHA	**79.90**±0.29	**74.62**±0.35	**75.74**±0.32	**64.50**±0.41	59.04±0.34	**59.16**±0.21	**79.20**±0.13	**74.46**±0.17	**75.24**±0.27

(GCN backbone)

Table 2. Ablation study on Cora-LT with GCN. "+" stands for synthesizing.

Method	Acc.	bAcc.	F1	C0 (0.5%)	C1 (1.1%)	C2 (2.4%)	C3 (5.4%)	C4 (11.6%)	C5 (25.0%)	C6 (54.0%)
GCN	72.02±0.50	59.42±0.74	59.23±1.02	0.0	28.6	67.0	60.0	81.2	93.8	93.1
+easy samples	76.90±0.19	69.55±0.21	71.28±0.25	21.1	69.4	67.9	63.1	73.1	95.1	94.7
+harder samples w/o SemiMixup	75.84±0.38	71.38±0.58	71.44±0.59	54.7	71.7	63.1	58.4	74.2	92.5	82.6
+harder samples w/ SemiMixup	79.16±0.25	72.89±0.32	74.62±0.27	42.2	74.3	71.8	62.3	72.5	94.4	93.4
+harder samples w/ SemiMixup (HK)	79.60±0.17	74.37±0.18	75.17±0.15	48.4	75.8	68.3	63.2	77.8	93.5	92.8
+harder samples w/ SemiMixup (PPR)	79.90±0.29	74.62±0.35	75.74±0.32	51.6	76.9	66.0	65.4	76.5	93.8	92.1

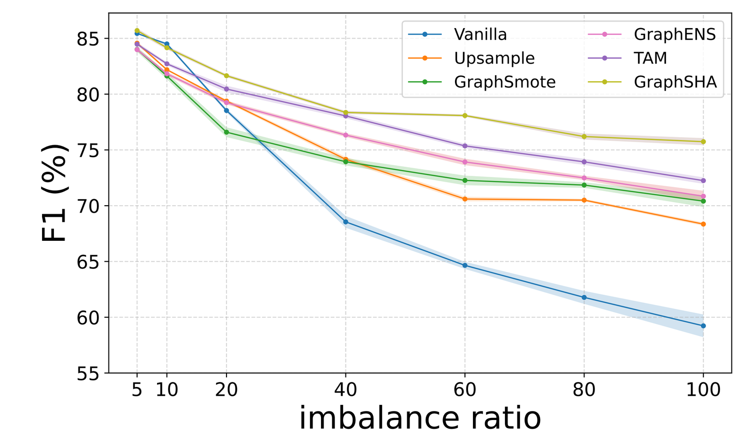


Figure 4. Changing trend of F1-score with the increase of imbalance ratio on Cora-LT with GCN.

Class Distribution	C_0 (0.5%)	C_1 (1.1%)	C_2 (2.4%)	C_3 (5.4%)	C_4 (11.6%)	C_5 (25.0%)	C_6 (54.0%)
Vanilla (GCN)	0.0	28.6	67.0	60.0	81.2	93.8	93.1
Reweight	32.9	70.9	75.0	67.5	78.6	93.5	87.9
PC Softmax	29.7	78.0	70.9	66.2	81.2	93.8	82.5
CB Loss	31.3	74.7	72.8	69.2	81.9	94.0	83.4
Focal Loss	29.9	75.9	72.2	71.9	82.6	94.6	83.2
ReNode	35.9	73.6	72.8	66.9	83.9	95.1	88.1
Upsample	12.5	58.2	65.1	67.5	76.5	92.4	89.7
GraphSmote	22.2	66.4	68.9	62.1	79.4	93.4	89.4
GraphENS	37.7	72.2	73.2	63.5	74.6	94.7	82.3
TAM (G-ENS)	32.6	76.3	68.9	71.5	74.2	95.1	86.9
GraphSHA	**51.6**	76.9	66.0	65.4	76.5	93.8	**92.1**

Table 3. Classification accuracy for each class on Cora-LT.

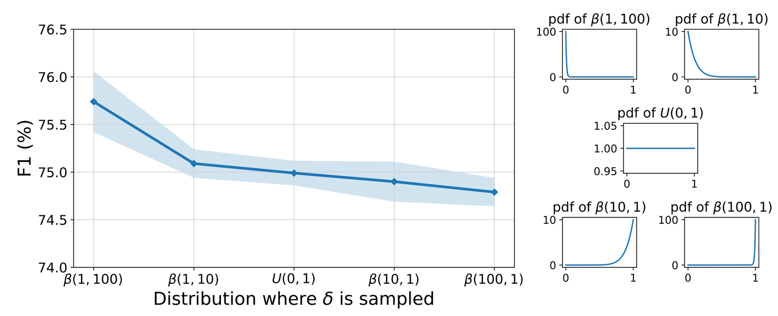


Figure 5. Performance of GraphSHA w.r.t. different distributions where δ is sampled.

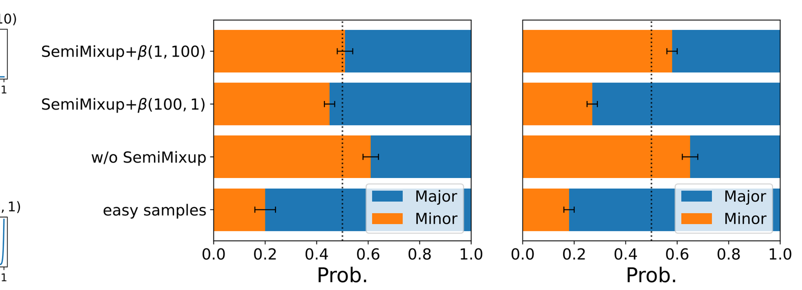
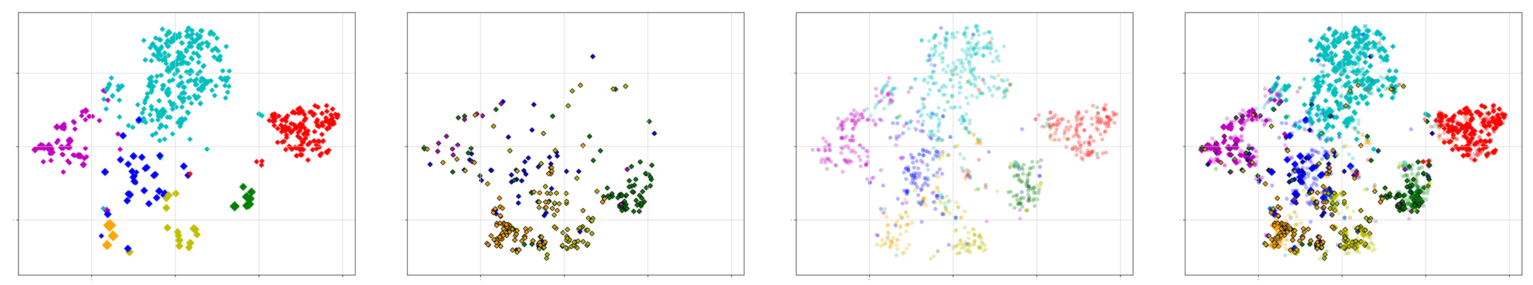


Figure 6. probability distribution of misclassified samples with GCN backbone.



(a) Training set samples (b) Synthesized samples (c) Test set samples (d) All samples from (a), (b), and (c)

Figure 7. Visualization of GraphSHA on Cora-LT with GCN, where each node is colored by its label. In (a), the hardness of each training node is marked via the node size.

Code: https://github.com/wenzhilics/GraphSHA Paper: https://arxiv.org/abs/2306.09612 Contact: liwzh63@mail2.sysu.edu.cn