Tail-GNN: Tail-Node Graph Neural Networks

Zemin Liu, Trung-Kien Nguyen, Yuan Fang

In Proceedings of the 27th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD-21)
14th -18th August, 2021

Outline

- **Problem & related work**
- Challenges & insights
- Proposed model: Tail-GNN
- Experiments
- Conclusions

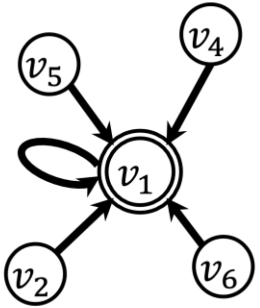
Graph Representation Learning

- Graph embedding approaches
  - DeepWalk [1], node2vec [2], …

- Graph neural networks (GNNs) [3,4,5]

$$\mathbf{h}_v^l = \mathcal{M}(\mathbf{h}_v^{l-1}, \{\mathbf{h}_i^{l-1} : i \in \mathcal{N}_v\}; \theta^l)$$

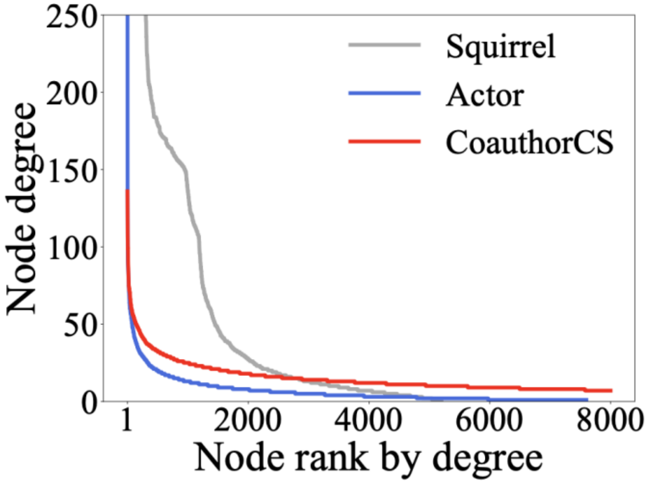
Message passing function



[1] Perozzi B., et al. 2014. Deepwalk: Online learning of social representations. KDD.
[2] Grover A., et al. 2014. node2vec: Scalable feature learning for networks. KDD.
[3] Kipf, T. N., et al. 2017. Semi-supervised classification with graph convolutional networks. ICLR.
[4] Veličković, P., et al. 2018. Graph attention networks. ICLR.
[5] Hamilton W L., et al. 2017. Inductive representation learning on large graphs. NeurIPS.

Problem: long-tailed node distribution

- Long-tailed distribution
  - Node degree
- GNNs
  - Depend on the abundance of structural information (head nodes *vs.* tail nodes)
  - Do not pay special attention to tail nodes
- Problem
  - Robust tail node embedding with GNNs
  - Definition for tail and head nodes



Long-tailed node distribution

$$\mathcal{V}_{\text{tail}} = \{v : |\mathcal{N}_v| \leqslant K\} \qquad \mathcal{V}_{\text{head}} = \{v : |\mathcal{N}_v| > K\}$$

Related Work

- Degree-specific models [1,2]
  - Distinguish nodes based on their degrees
  - Not specifically designed to enhance the embeddings of the tail nodes

- meta-tail2vec [3]
  - For tail node embedding
  - Main disadvantage: decoupled two-stage, not end-to-end

[1] Wu J, et al. 2019. Demo-Net: Degree-specific graph neural networks for node and graph classification. KDD.
[2] Tang X, et al. 2020. Investigating and Mitigating Degree-Related Biases in Graph Convoltuional Networks. CIKM.
[3] Liu Z, et al. 2020. Towards locality-aware meta-learning of tail node embeddings on networks. CIKM.

Outline

- Problem & related work
- **Challenges & insights**
- Proposed model: Tail-GNN
- Experiments
- Conclusions

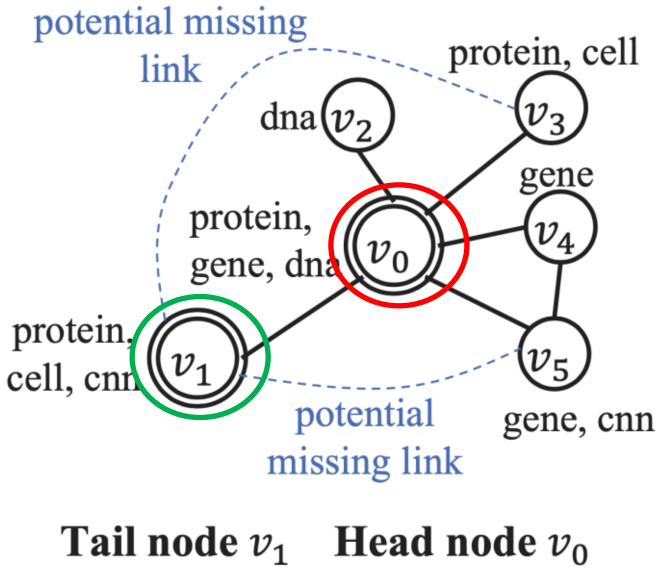
Challenges

- Tail nodes
  - Small neighborhood
  - Potentially suffer from missing information

- Challenges

  C1: How to uncover the missing neighborhood information for tail nodes?

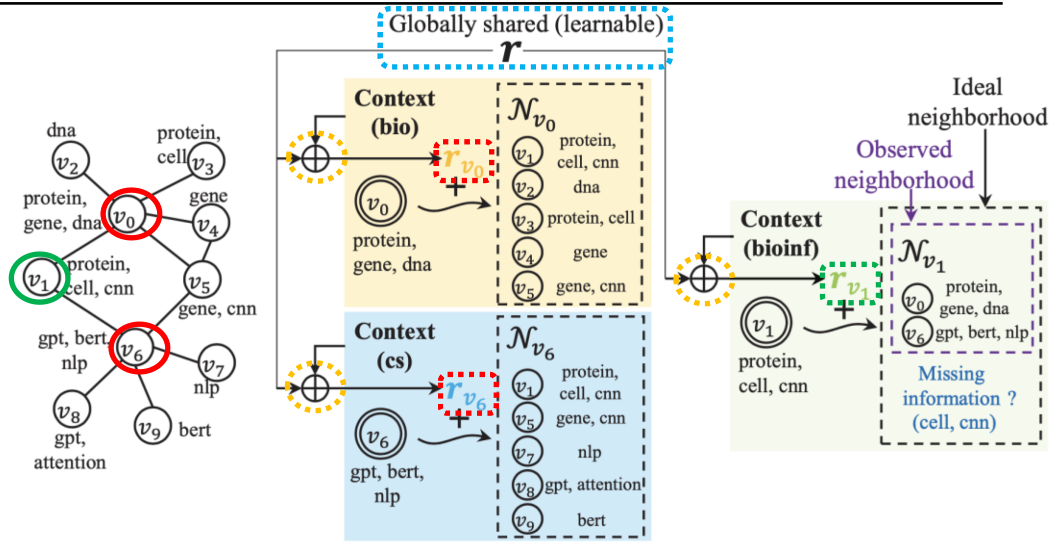
  C2: How to localize the missing information for each tail node while maintaining the generality across nodes?



Toy citation network

Insights: Tail-GNN

- Key idea
  - Neighborhood translation
- First challenge
  - predict the missing neighborhood information for tail nodes by exploiting a transferable neighborhood translation
- Second challenge
  - tailor the shared neighborhood translation to each target node w.r.t. its local context.

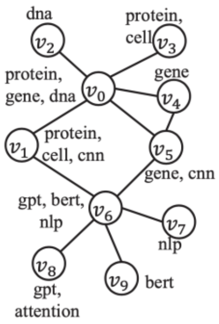


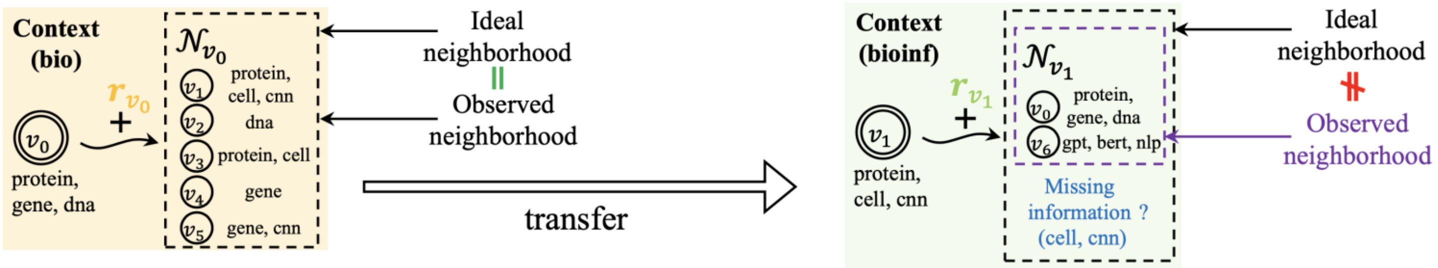
(a) Toy network (b) Neighborhood translation for head nodes (c) Neighborhood translation for tail nodes

Outline

- Problem & related work
- Challenges & insights
- **Proposed model: Tail-GNN**
- Experiments
- Conclusions

Concept: transferable neighborhood translation





- Neighborhood of **head** nodes
  - Observed neighborhood: complete and representative
  - no missing information

- Neighborhood of **tail** nodes
  - Observed neighborhood: not representative enough
  - Imperative: uncover the missing information

Neighborhood translation

Translation vector

$$\boxed{\mathbf{h}_v + \mathbf{r}_v \approx \mathbf{h}_{\mathcal{N}_v}}$$

Embedding vector

Embedding of observed neighborhood

$$\mathbf{m}_v = \mathbf{h}_{\mathcal{N}_v^*} - \mathbf{h}_{\mathcal{N}_v} = \mathbf{0}$$

Missing information

$$\mathbf{m}_v = \mathbf{h}_{\mathcal{N}_v^*} - \mathbf{h}_{\mathcal{N}_v} \neq \mathbf{0}$$

Embedding of ideal neighborhood

Embedding of observed neighborhood

Embedding of ideal neighborhood

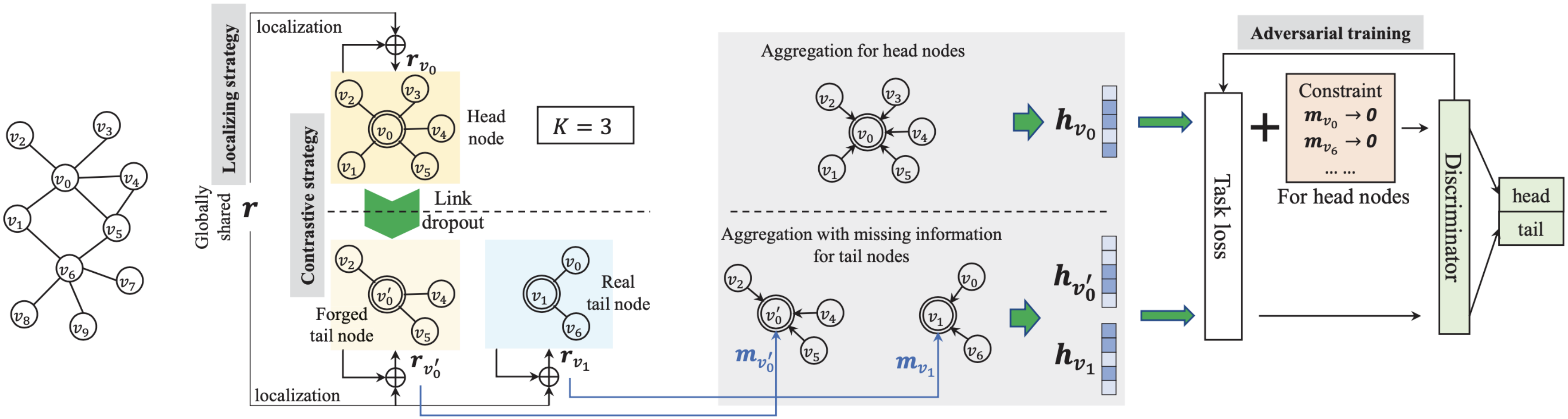
Embedding of observed neighborhood

- Predicting missing information for tail node *v*

Predict embedding of ideal neighborhood for tail node *v* ⇒ $\mathbf{h}_{\mathcal{N}_v^*} = \mathbf{h}_v + \mathbf{r}_v$

Predict missing information for tail node *v* ⇒ $\mathbf{m}_v = \mathbf{h}_v + \mathbf{r}_v - \mathbf{h}_{\mathcal{N}_v}$

Tail-GNN: overall framework



(a) Toy network

(b) Neighborhood translation for missing information prediction

(c) Neighbors aggregation with missing information

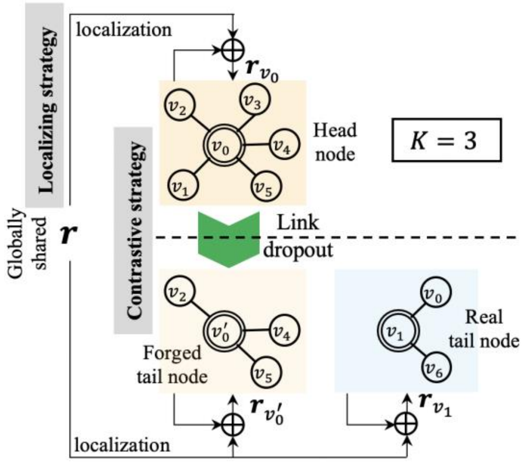
(d) Loss function and constraints

Tail-GNN: realizing neighborhood translation (1)

- Contrastive strategy
  - Head nodes

$$\mathbf{m}_v^l = \mathbf{h}_v^l + \mathbf{r}_v^l - \mathbf{h}_{\mathcal{N}_v}^l \rightarrow \mathbf{0}$$

Embedding of observed neighborhood



  - Tail nodes
    - Forged tail nodes: randomly dropping some links from the head nodes, for contrast
    - Robust tail node embedding: uncover the missing neighborhood information

missing neighborhood information

$$\mathbf{m}_v^l = \mathbf{h}_{\mathcal{N}_v^*}^l - \mathbf{h}_{\mathcal{N}_v}^l = \mathbf{h}_v^l + \mathbf{r}_v^l - \mathbf{h}_{\mathcal{N}_v}^l$$

Tail-GNN: realizing neighborhood translation (2)

- Localizing strategy
  - Local context of each node
  - Generality across the graph

Local context

Parameter set

$$\mathbf{r}_v^l = \phi(\mathbf{h}_v^l, \mathbf{h}_{\mathcal{N}_v}^l, \mathbf{r}^l; \theta_\phi^l)$$

  - Scaling and shifting factors [1]

$$\mathbf{r}_v^l = \phi(\mathbf{h}_v^l, \mathbf{h}_{\mathcal{N}_v}^l, \mathbf{r}^l; \theta_\phi^l) = (\gamma_v^l + \mathbf{1}) \odot \mathbf{r}^l + \beta_v^l$$

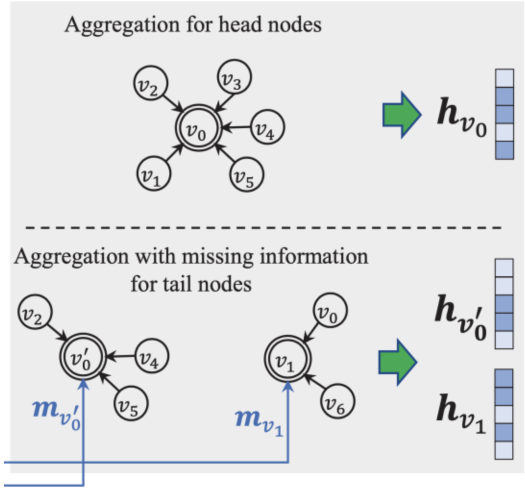
Scaling vector

Shifting vector

[1] Perez E, et al. 2018. FiLM: Visual reasoning with a general conditioning layer. AAAI.

Tail-GNN: neighborhood aggregation

- Neighborhood aggregation



- Head nodes
- Tail nodes

Message passing function

$$\mathbf{h}_v^{l+1} = \mathcal{M}(\mathbf{h}_v^l, \{\mathbf{h}_i^l : i \in \mathcal{N}_v\}; \theta^{l+1})$$

$$\mathbf{h}_v^{l+1} = \mathcal{M}(\mathbf{h}_v^l, \{\mathbf{m}_v^l\} \cup \{\mathbf{h}_i^l : i \in \mathcal{N}_v\}; \theta^{l+1})$$

missing neighborhood information

observed neighborhood

Tail-GNN: overall loss

- Task loss

Cross entropy

$$\mathcal{L}_t = \sum_{v \in \mathcal{V}_{\text{tr}}} \textsc{CrossEnt}(\mathbf{h}_v^\ell, \mathbf{y}_v) + \lambda_t \|\Theta\|_2^2$$

- Loss for missing information constraint

$$\mathcal{L}_m = \sum_{v \in \mathcal{V}_{\text{tr}}} I_v \sum_{l=1}^{\ell} \|\mathbf{m}_v^{l-1}\|_2^2$$

Missing information

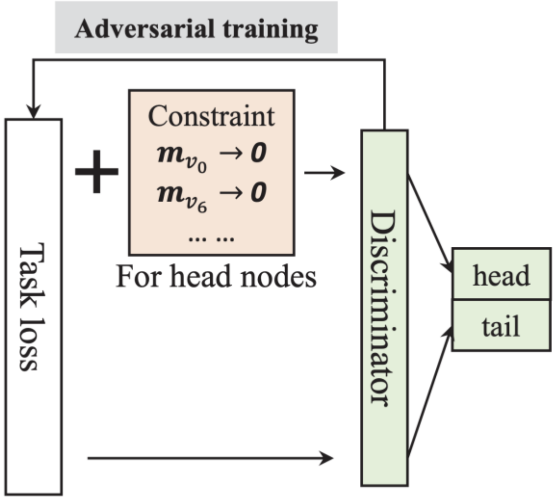
- Loss for adversarial constraint [1]

$$\mathcal{L}_d = \sum_{v \in \mathcal{V}_{\text{tr}}} \textsc{CrossEnt}(I_v, D(\mathbf{h}_v^\ell; \theta_d)) + \lambda_d \|\theta_d\|_2^2$$

Discriminator

- Overall loss

$$\min_\Theta \max_{\theta_d} \; \mathcal{L}_t + \mu \mathcal{L}_m - \eta \mathcal{L}_d$$



[1] Goodfellow I J, et al. 2014. Generative Adversarial Nets. NeurIPS.

Outline

- Problem & related work
- Challenges & insights
- Proposed model: Tail-GNN
- **Experiments**
- Conclusions

Experimental setup

Datasets

	# Nodes	# Edges	# Features	# Classes	# Tail (K = 5)
Email	1,005	25,571	128	42	235
Squirrel	5,201	217,073	2,089	5	942
Actor	7,600	33,391	931	5	4,823
CoauthorCS	18,333	327,576	6,805	15	8,037
Amazon	937,349	12,455,925	100	44	248,125

Base GNN models

- GCN [1]
- GAT [2]
- GraphSAGE [3]

Baselines

- Conventional:
  - DeepWalk [4], GCN [1]
- Refinement:
  - Additive [5], a la carte [6], meta-tail2vec [7]
- Robust models:
  - SDNE [8], ARGA [9], DDGCN
- Degree-aware models:
  - Demo-Net [11], role2vec

[1] Kipf, T. N., et al. 2017. Semi-supervised classification with graph convolutional networks. ICLR.
[2] Veličković, P., et al. 2018. Graph attention networks. ICLR.
[3] Hamilton W L., et al. 2017. Inductive representation learning on large graphs. NeurIPS.
[4] Perozzi B., et al. 2014. Deepwalk: Online learning of social representations. KDD.
[5] Lazaridou A, et al. 2017. Multimodal word meaning induction from minimal exposure to natural text. Cognitive science.
[6] Khodak M, et al. 2018. A la carte embedding: Cheap but effective induction of semantic feature vectors. ACL.
[7] Liu Z, et al. 2020. Towards locality-aware meta-learning of tail node embeddings on networks. CIKM.
[8] Wang D, et al. 2016. Structural deep network embedding. KDD.
[9] Pan S, et al. 2018. Adversarially regularized graph autoencoder for graph embedding. IJCAI.
[10] Cai R, et al. 2020. Dual-dropout graph convolutional network for predicting synthetic lethality in human cancers. Bioinformatics.
[11] Wu J, et al. 2019. Demo-Net: Degree-specific graph neural networks for node and graph classification. KDD.
[12] Ahmed N, et al. 2020. Role-based graph embeddings. TKDE.

Node classification for tail nodes

- **GCN as base model**

Table 2: Evaluation on tail node classification using GCN as the base model.

Henceforth, tabular results are in percent; the best result is **bolded** and the runner-up is underlined; a dash (-) denotes no result reported for failing to work on a large dataset.

Methods	Email		Squirrel		Actor		CoauthorCS		Amazon	
	Accuracy	Micro-F	Accuracy	Micro-F	Accuracy	Micro-F	Accuracy	Micro-F	Accuracy	Micro-F
DeepWalk	54.4 ± 0.3	51.3 ± 0.3	28.8 ± 1.6	28.0 ± 2.3	21.8 ± 0.6	18.2 ± 0.9	84.1 ± 0.7	81.5 ± 0.7	83.7 ± 0.1	74.3 ± 0.6
GCN	57.9 ± 1.2	57.7 ± 1.3	24.8 ± 1.3	23.2 ± 1.8	29.7 ± 0.2	15.0 ± 0.9	88.4 ± 0.1	86.1 ± 0.1	82.3 ± 0.2	70.6 ± 0.1
Additive	55.4 ± 0.4	52.5 ± 0.2	27.0 ± 1.7	22.9 ± 1.6	28.1 ± 0.3	15.1 ± 1.3	89.5 ± 0.1	87.8 ± 0.1	84.2 ± 0.2	73.2 ± 0.6
a la carte	21.1 ± 0.4	17.9 ± 0.5	22.5 ± 1.1	22.5 ± 0.7	28.0 ± 0.5	14.8 ± 1.4	88.7 ± 0.2	86.7 ± 0.3	81.1 ± 0.1	69.7 ± 0.7
meta-tail2vec	57.1 ± 0.1	55.3 ± 0.2	25.1 ± 0.5	21.5 ± 0.3	29.7 ± 0.4	20.1 ± 0.7	89.3 ± 0.1	87.4 ± 0.1	81.9 ± 0.1	71.4 ± 0.4
SDNE	32.9 ± 0.6	29.8 ± 0.5	23.8 ± 3.2	16.6 ± 6.2	24.4 ± 0.8	12.6 ± 5.6	70.6 ± 0.9	64.5 ± 1.1	-	-
ARGA	45.1 ± 0.9	41.2 ± 1.0	22.4 ± 1.0	22.8 ± 1.9	25.9 ± 0.3	8.2 ± 0.6	74.6 ± 1.8	67.9 ± 2.5	-	-
DDGCN	39.8 ± 0.6	38.9 ± 0.7	26.3 ± 2.1	26.4 ± 3.3	24.0 ± 0.4	11.7 ± 0.7	73.6 ± 0.9	68.8 ± 1.0	-	-
DEMO-Net	56.9 ± 0.6	56.5 ± 0.7	28.3 ± 0.5	22.5 ± 2.2	28.4 ± 0.8	22.0 ± 1.3	90.8 ± 0.5	88.9 ± 0.6	83.1 ± 0.1	72.0 ± 0.4
role2vec	44.9 ± 1.6	43.8 ± 2.4	26.3 ± 0.8	27.5 ± 1.7	23.1 ± 0.1	18.3 ± 0.6	62.7 ± 0.3	56.3 ± 0.3	77.1 ± 0.2	61.5 ± 0.5
Tail-GCN	**59.2** ± 0.8	**58.5** ± 1.3	**30.2** ± 1.1	**31.1** ± 1.1	**34.9** ± 0.5	**25.2** ± 0.6	**93.6** ± 0.1	**92.7** ± 0.1	**87.0** ± 0.1	**78.2** ± 0.2

- **Other GNNs as the base model**

Table 3: Evaluation on tail node classification using other GNNs as the base model.

Methods	Email		Squirrel		Actor		CoauthorCS		Amazon	
	Accuracy	Micro-F	Accuracy	Micro-F	Accuracy	Micro-F	Accuracy	Micro-F	Accuracy	Micro-F
GAT	57.9 ± 0.4	57.3 ± 0.2	24.1 ± 2.4	23.1 ± 2.6	29.8 ± 0.6	13.2 ± 2.7	88.6 ± 0.2	86.2 ± 0.2	-	-
Tail-GAT	**59.4** ± 0.9	**58.2** ± 1.2	**28.8** ± 2.1	**30.4** ± 2.6	**34.5** ± 1.3	**24.7** ± 2.0	**92.5** ± 0.1	**90.8** ± 0.1	-	-
GraphSAGE	52.0 ± 1.6	51.3 ± 1.7	27.1 ± 2.7	26.4 ± 4.9	33.1 ± 1.1	23.2 ± 2.4	89.8 ± 2.4	87.7 ± 1.1	79.1 ± 0.4	62.8 ± 0.6
Tail-GraphSAGE	**55.7** ± 0.6	**54.9** ± 0.7	**28.5** ± 1.6	**28.2** ± 2.4	**34.1** ± 1.7	**26.8** ± 1.8	**93.8** ± 0.7	**92.4** ± 1.4	**85.1** ± 0.2	**75.5** ± 0.3

Ablation study and scalability study

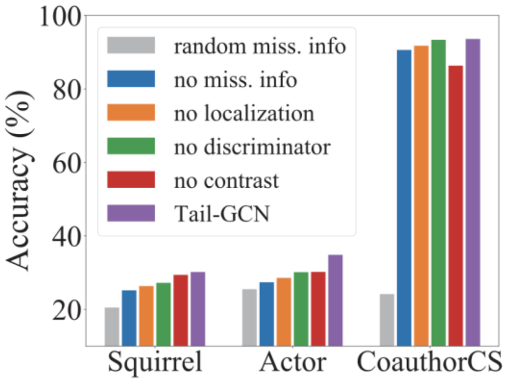


Figure 4: Ablation study.

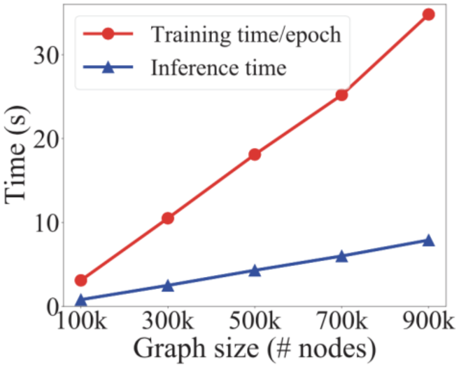


Figure 5: Scalability study.

- **Ablation study**
  - Random/no missing info impairs the performance
  - Without localization: hurts the performance
  - Discriminator contributes to the performance
  - Without contrastive strategy: performance becomes worse
- **Scalability**
  - Increase linearly w.r.t. graph size

Outline

- Problem & related work
- Challenges & insights
- Proposed model: Tail-GNN
- Experiments
- **Conclusions**

Conclusions

- **Problem**
  - Tail node embedding in graph neural networks
- **Proposed model**
  - A new concept of **transferable neighborhood translation**
    - to capture the relational tie between a node and its neighboring nodes
  - A novel model **Tail-GNN**
    - to narrow the gap between head and tail nodes for robust tail node embedding
- **Experiments**

Thanks!

Paper, code, data…
www.yfang.site

Tail-GNN: Tail-Node Graph Neural Networks

Zemin Liu, Trung-Kien Nguyen, Yuan Fang

In Proceedings of the 27th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD-21)
14th -18th August, 2021