Towards Locality-Aware Meta-Learning of Tail Node Embeddings on Networks

Zemin Liu[*,1], Wentao Zhang[*,2], Yuan Fang[$,1], Xinming Zhang[2], Steven C.H. Hoi[1,3]

[1] Singapore Management University

[2] University of Science and Technology of China

[3] Salesforce Research Asia

[*] Co-first authors with equal contribution

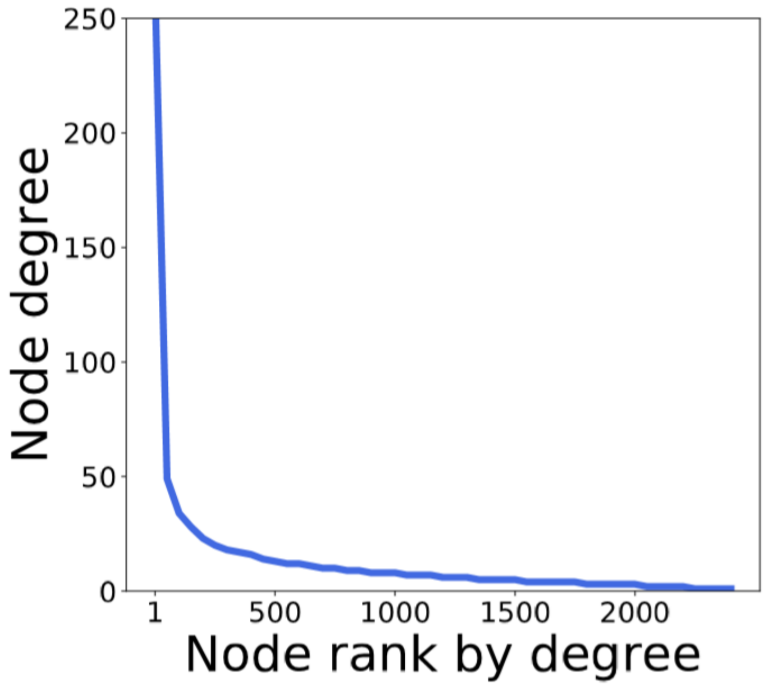
[$] Presenter of this video

Outline

Background

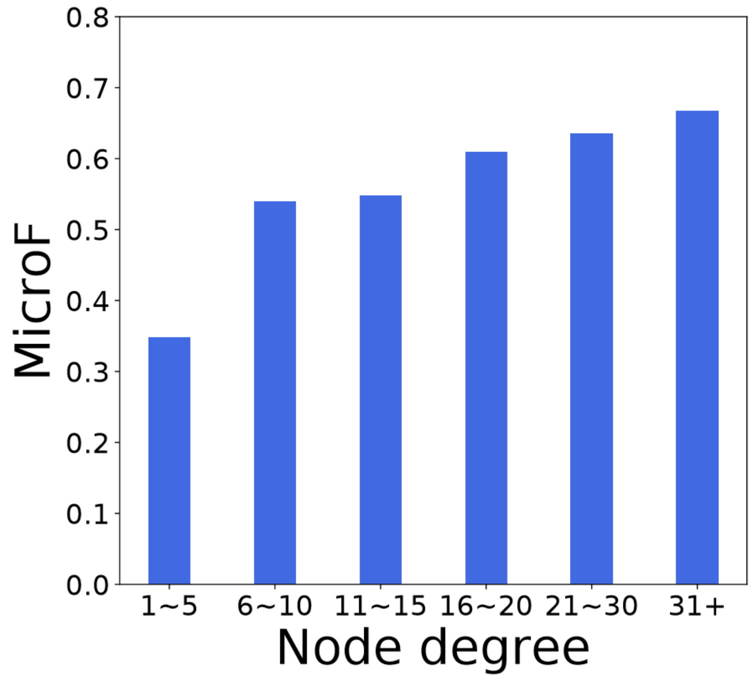
- ❑ Node degrees vary considerably across the network
  - Power-law distribution

- ❑ Low-degree or *tail* nodes are ubiquitous
  - Newcomers
  - Existing users who are less active



Tail nodes are under-modeled

- Tail nodes with very few links are under-modeled
  - Limited structural information
  - Existing methods regard all nodes uniformly using the same model



***Problem:** Given the embedding vectors of nodes learned from a base embedding model, can we refine/improve the embeddings of the tail nodes?*

Challenges and insights

❑ Challenge 1: Tail nodes have scarce structural information

Oracle reconstruction

- Leverage higher quality embeddings of head nodes $\mathcal{V}_{\text{head}}$

$$\arg\min_{\Theta} \sum_{v \in \mathcal{V}_{\text{head}}} \|F(v;\Theta) - \mathbf{h}_v\|^2$$

Embedding vector of head node v from a base embedding model

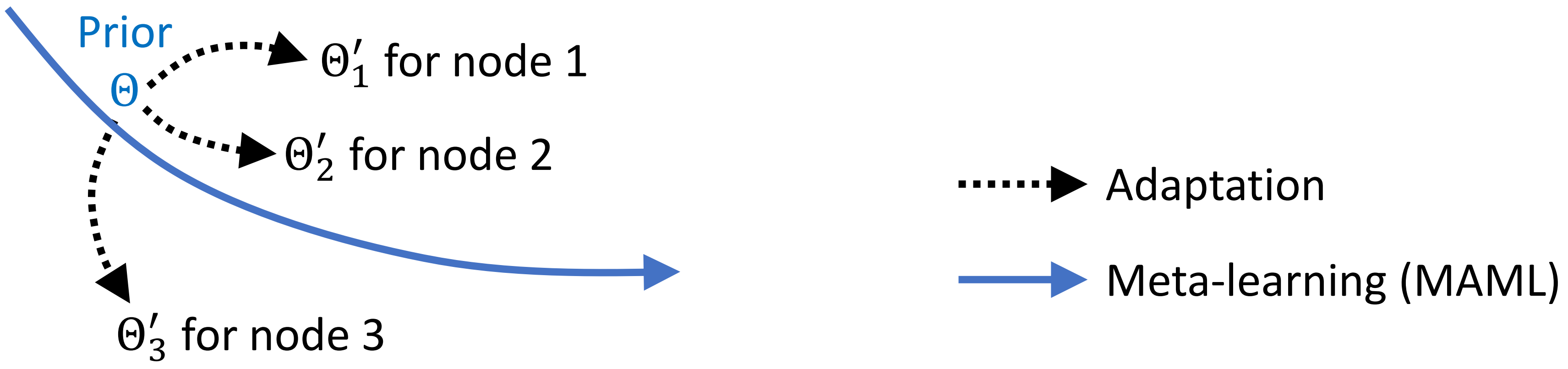
Regression model trained on head nodes, and predict on tail nodes

Link dropouts

- Head nodes have more links than tail nodes
- Drop some of the links on head nodes to simulate tail nodes

Challenges and insights

- **Challenge 2: Each node has a unique local context**
  - The regression model needs adapted to each node
  - Learn how to adapt: Meta-learning (MAML)

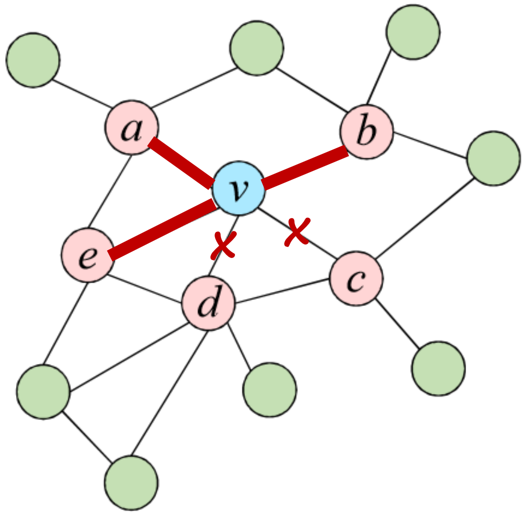


Outline

Regression model

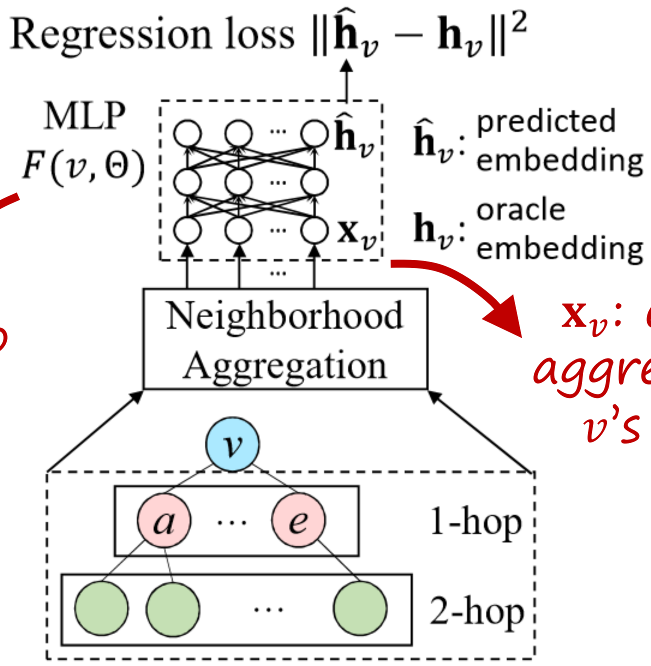
- Reconstructing the embedding of a head node from its neighbors
- Only utilize some of the neighbors

Link dropout



Oracle reconstruction

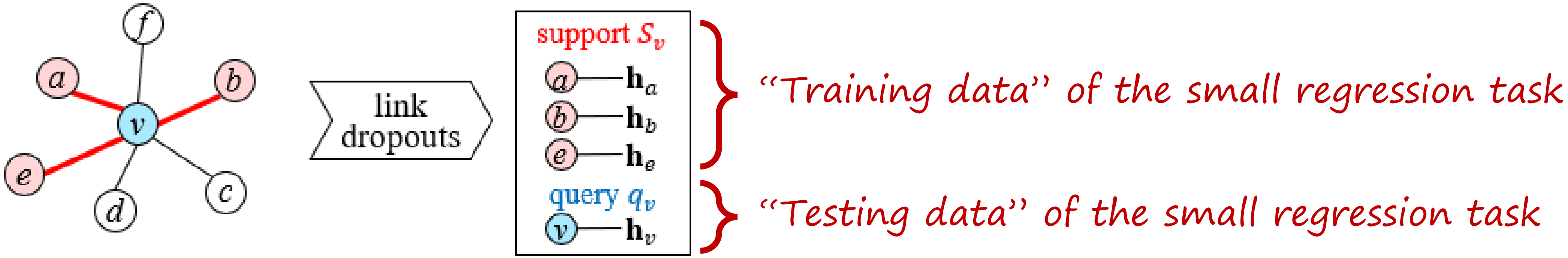
Regression model to predict $\hat{\mathbf{h}}_v$, using $\mathbf{x}_v$ as input features



$\mathbf{x}_v$: embedding aggregated from v's neighbors

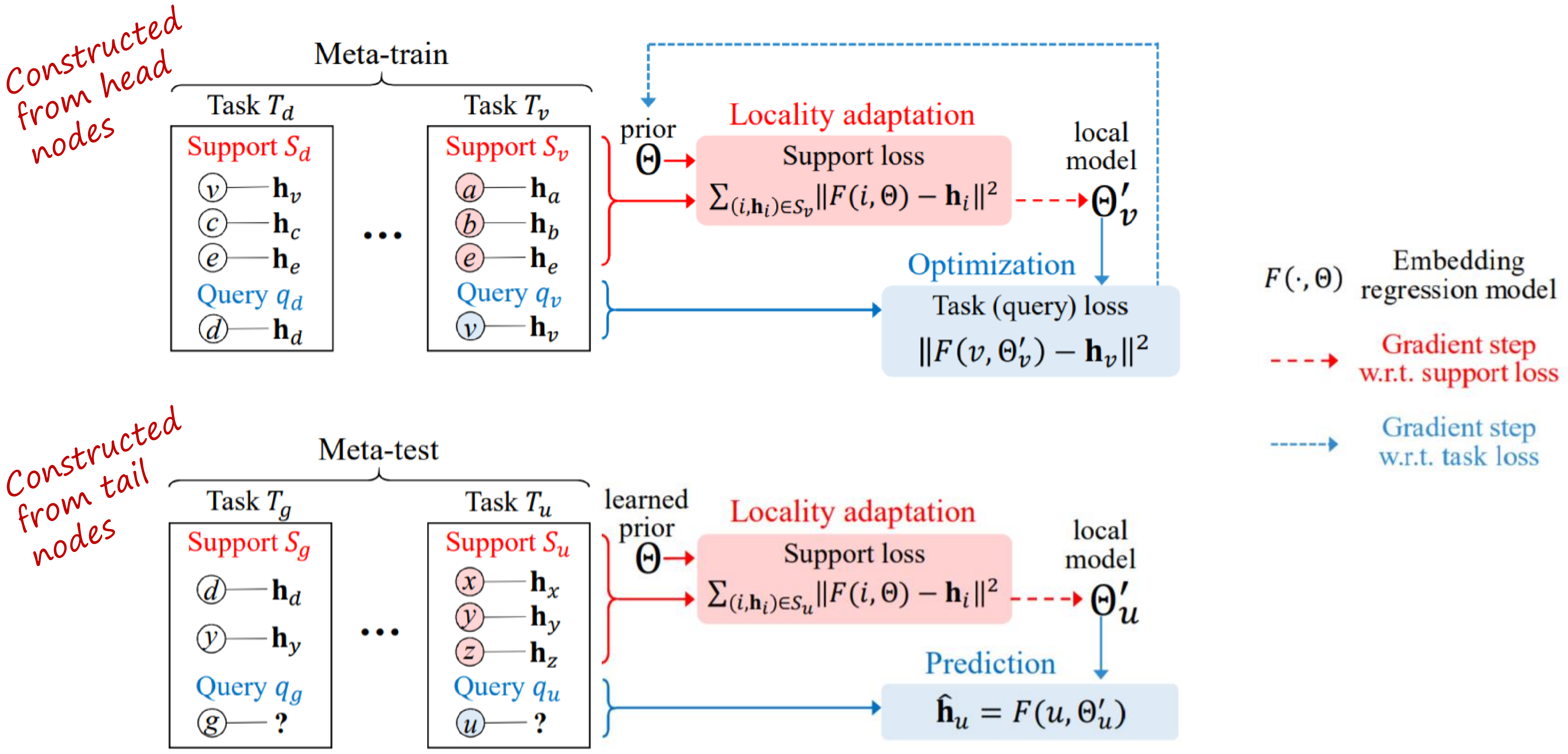
Locality-aware tasks

- Each node forms a "small" regression task
  - For head nodes, perform link dropouts



- Support set only consists of a few neighboring nodes
  - Known as **few-shot regression**
  - **Locality-aware** tasks, assuming neighboring nodes has similar local contexts

meta-tail2vec: Locality-aware meta-learning



Meta-training process of meta-tail2vec

1. Given a task $T_v = (S_v, q_v)$, the loss of the prior Θ on support is

$$L_{S_v}(\Theta) = \sum_{(i,\mathbf{h}_i)\in S_v} \|F(i;\Theta) - \mathbf{h}_i\|^2$$

2. Θ will be adapted on support S_v by one (or a few) gradient updates

$$\Theta'_v = \Theta - \alpha \frac{\partial L_{S_v}(\Theta)}{\partial \Theta}$$

3. The local model Θ'_v will be applied on query v to calculate the task loss

$$L_{q_v}(\Theta'_v) = \|F(v;\Theta'_v) - \mathbf{h}_v\|^2$$

4. Task losses are backpropagated to update the prior Θ

$$\arg\min_{\Theta} \sum_{T_v=(S_v,q_v)\in\mathcal{T}_{\text{train}}} L_{q_v}\left(\Theta - \alpha \frac{\partial L_{S_v}(\Theta)}{\partial \Theta}\right)$$

Outline

Experimental settings

- Datasets

	# nodes	# edges	# node classes	multi-label	# tail nodes
Wiki	2,405	17,981	19	No	1,069
Flickr	80,513	5,899,882	195	Yes	9,367
Email	1,005	25,571	42	No	235

Defined as degree ≤ 5

- Base embedding model
  - DeepWalk, GraphSAGE
  - SDNE, ARGA, DDGCN (robust models for sparse networks)

- Baselines for refining tail node embeddings w.r.t. each base model
  - Biased walk, Additive, A la carte, Nonce2vec, Dropout

Evaluation on node classification

		Base	Biased walk	Additive	Additive-2	a la carte	a la carte-2	Nonce2vec	Dropout	meta-tail2vec	Improv. over Base	Improv. over 2^{nd} best
DeepWalk as the base embedding model												
Wiki	MicroF	44.27 ± 0.25	44.69 ± 0.31	45.32 ± 0.52	42.11 ± 0.76	23.65 ± 0.44	23.34 ± 0.47	44.97 ± 0.29	36.88 ± 0.65	**49.10** ± 0.23	+10.9%	+8.3%
	Accuracy	46.68 ± 0.31	47.05 ± 0.17	47.18 ± 0.29	44.73 ± 0.53	24.17 ± 0.49	24.48 ± 0.42	47.11 ± 0.22	38.13 ± 0.57	**50.70** ± 0.45	+8.6%	+7.5%
Flickr	MicroF	33.48 ± 0.26	33.61 ± 0.39	34.43 ± 0.41	32.59 ± 0.17	31.89 ± 0.47	32.25 ± 0.35	33.83 ± 0.28	33.91 ± 0.22	**36.31** ± 0.19	+8.5%	+5.5%
	Accuracy	32.44 ± 0.13	32.57 ± 0.19	33.29 ± 0.17	31.31 ± 0.24	32.13 ± 0.26	32.62 ± 0.31	33.01 ± 0.15	32.86 ± 0.09	**35.28** ± 0.25	+8.8%	+6.0%
Email	MicroF	51.32 ± 0.29	50.95 ± 0.24	52.50 ± 0.17	51.17 ± 0.23	17.88 ± 0.48	18.21 ± 0.52	51.84 ± 0.33	32.72 ± 0.45	**55.26** ± 0.18	+7.7%	+5.3%
	Accuracy	54.41 ± 0.34	54.13 ± 0.22	55.38 ± 0.43	53.82 ± 0.36	21.06 ± 0.45	21.13 ± 0.37	54.79 ± 0.19	33.85 ± 0.51	**57.78** ± 0.29	+6.2%	+4.3%
GraphSAGE as the base embedding model												
Wiki	MicroF	39.68 ± 0.24	40.07 ± 0.15	37.84 ± 0.31	35.96 ± 0.43	23.88 ± 0.47	22.52 ± 0.39	40.75 ± 0.33	19.78 ± 0.59	**44.29** ± 0.31	+11.6%	+8.7%
	Accuracy	41.22 ± 0.19	41.39 ± 0.06	39.31 ± 0.26	36.59 ± 0.25	25.71 ± 0.36	24.94 ± 0.62	41.65 ± 0.28	24.73 ± 0.42	**44.90** ± 0.12	+8.9%	+7.8%
Flickr	MicroF	29.38 ± 0.32	28.75 ± 0.31	27.86 ± 0.14	23.69 ± 0.44	30.02 ± 0.17	29.67 ± 0.20	29.85 ± 0.12	28.75 ± 0.11	**32.11** ± 0.41	+9.3%	+7.0%
	Accuracy	28.46 ± 0.08	27.52 ± 0.19	27.69 ± 0.31	22.82 ± 0.45	29.83 ± 0.22	28.18 ± 0.46	29.26 ± 0.31	28.78 ± 0.14	**31.96** ± 0.35	+12.3%	+7.1%
Email	MicroF	41.25 ± 0.17	41.07 ± 0.33	35.83 ± 0.31	34.19 ± 0.13	27.81 ± 0.44	26.97 ± 0.39	41.97 ± 0.24	23.47 ± 0.25	**46.73** ± 0.37	+13.3%	+11.3%
	Accuracy	42.61 ± 0.31	42.20 ± 0.31	37.25 ± 0.16	35.13 ± 0.35	29.41 ± 0.46	27.16 ± 0.34	43.23 ± 0.30	25.84 ± 0.18	**47.70** ± 0.46	+11.9%	+10.3%

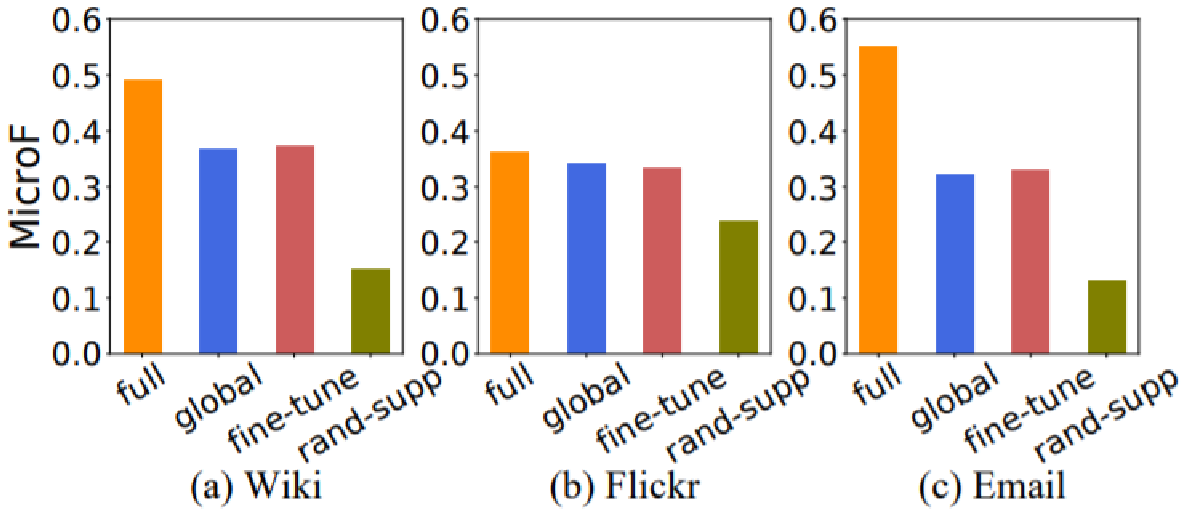
Evaluation on link prediction

		Base	Biased walk	Additive	Additive-2	a la carte	a la carte-2	Nonce2vec	Dropout	meta-tail2vec	Improv. over Base	Improv. over 2^{nd} best
DeepWalk as the base embedding model												
Wiki	MRR	75.28 ± 0.37	75.13 ± 0.41	75.81 ± 0.62	74.89 ± 0.78	76.31 ± 0.25	76.14 ± 0.33	67.42 ± 0.87	77.06 ± 0.71	**79.18** ± 0.52	+5.2%	+2.8%
	Hit@1	51.83 ± 0.42	52.04 ± 0.57	52.51 ± 0.67	51.48 ± 0.39	53.70 ± 0.61	53.59 ± 0.32	53.34 ± 0.49	54.19 ± 0.30	**57.22** ± 0.46	+10.4%	+5.6%
Flickr	MRR	50.05 ± 0.30	49.57 ± 0.19	49.80 ± 0.45	49.72 ± 0.41	50.36 ± 0.55	50.71 ± 0.65	50.83 ± 0.48	50.25 ± 0.59	**52.18** ± 0.61	+4.3%	+2.7%
	Hit@1	25.32 ± 0.24	25.63 ± 0.55	26.10 ± 0.41	26.55 ± 0.62	26.07 ± 0.30	26.39 ± 0.58	26.67 ± 0.33	26.19 ± 0.44	**28.11** ± 0.40	+11.0%	+5.4%
Email	MRR	44.17 ± 0.35	44.58 ± 0.26	44.52 ± 0.68	44.96 ± 0.28	44.49 ± 0.50	45.11 ± 0.34	44.80 ± 0.15	45.33 ± 0.08	**48.42** ± 0.55	+9.6%	+6.8%
	Hit@1	19.47 ± 0.38	19.96 ± 0.27	21.38 ± 0.15	21.66 ± 0.40	22.45 ± 0.58	22.63 ± 0.31	20.90 ± 0.44	23.02 ± 0.33	**24.31** ± 0.46	+24.9%	+5.6%
GraphSAGE as the base embedding model												
Wiki	MRR	81.36 ± 0.14	82.01 ± 0.10	80.56 ± 0.45	80.39 ± 0.21	81.82 ± 0.53	80.94 ± 0.62	82.18 ± 0.64	82.52 ± 0.40	**84.38** ± 0.61	+3.7%	+2.3%
	Hit@1	58.87 ± 0.52	58.39 ± 0.15	58.43 ± 0.61	58.92 ± 0.30	59.56 ± 0.29	59.34 ± 0.44	59.70 ± 0.37	59.93 ± 0.56	**62.04** ± 0.68	+5.4%	+3.5%
Flickr	MRR	55.83 ± 0.29	56.17 ± 0.36	55.04 ± 0.25	55.40 ± 0.58	56.28 ± 0.49	56.76 ± 0.40	56.31 ± 0.32	56.85 ± 0.71	**58.15** ± 0.43	+4.2%	+2.3%
	Hit@1	34.59 ± 0.52	35.15 ± 0.47	33.79 ± 0.38	33.36 ± 0.40	35.22 ± 0.68	35.29 ± 0.64	34.97 ± 0.50	35.74 ± 0.31	**36.92** ± 0.39	+6.7%	+3.3%
Email	MRR	46.71 ± 0.45	46.24 ± 0.29	46.05 ± 0.25	46.68 ± 0.44	47.03 ± 0.53	46.92 ± 0.30	47.18 ± 0.19	46.37 ± 0.60	**48.15** ± 0.44	+3.1%	+2.1%
	Hit@1	23.02 ± 0.23	22.73 ± 0.41	22.91 ± 0.44	22.65 ± 0.52	23.19 ± 0.39	23.14 ± 0.61	23.28 ± 0.43	23.07 ± 0.56	**24.55** ± 0.70	+6.6%	+5.4%

Ablation study

Ablation study of the meta-learning strategy on node classification w.r.t. DeepWalk as the base model.

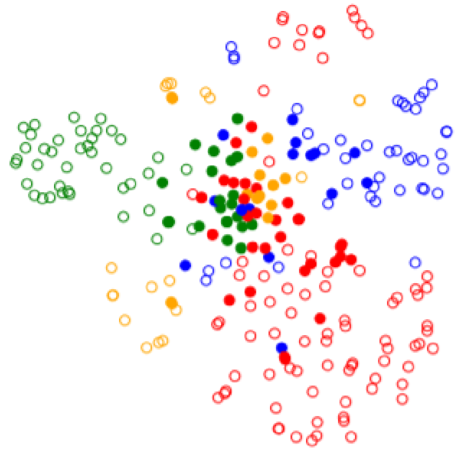
- *Full: The full meta-tail2vec model*
- *Global: Only train one global regression model on head nodes*
- *Fine-tune: Fine-tune the pre-trained global model on the support sets*
- *Rand-supp: Same as Full but samples random nodes as support sets*



Visualization

Visualization of base embeddings by SDNE, and their respective refinement by meta-tail2vec on the Email dataset.

Solid points denote tail nodes and hollow points denote head nodes. Each color represents one class.



(a) Base embeddings

(b) Refined embeddings by meta-tail2vec

Outline

1	Background & Problem
2	Proposed approach
3	Experiment
4	**Conclusion**

Outline

- We formulated the novel problem of learning tail node embeddings, and cast it as a regression problem based on oracle reconstruction and link dropout.
- We designed the locality-aware tasks on networks in a meta-learning framework, which allows for easy local adaption of the base model.
- We conducted extensive experiments on three public datasets, in which meta-tail2vec attains significant performance gains.

Q & A

Thank you!

More information: http://www.yfang.site/