Relative and Absolute Location Embedding for Few-Shot Node Classification on Graph

Zemin Liu[1], Yuan Fang[1], Chenghao Liu[1,2], Steven C.H. Hoi[1,2]

[1] Singapore Management University, Singapore;
[2] Salesforce Research Asia, Singapore

In Proceeding of 35th AAAI Conference on Artificial Intelligence, February 2-9, 2021

Outline

- **Problem definition & related work**
- Challenges & insights
- Proposed model: RALE
- Experiments
- Conclusions

Problem definition: few-shot node classification

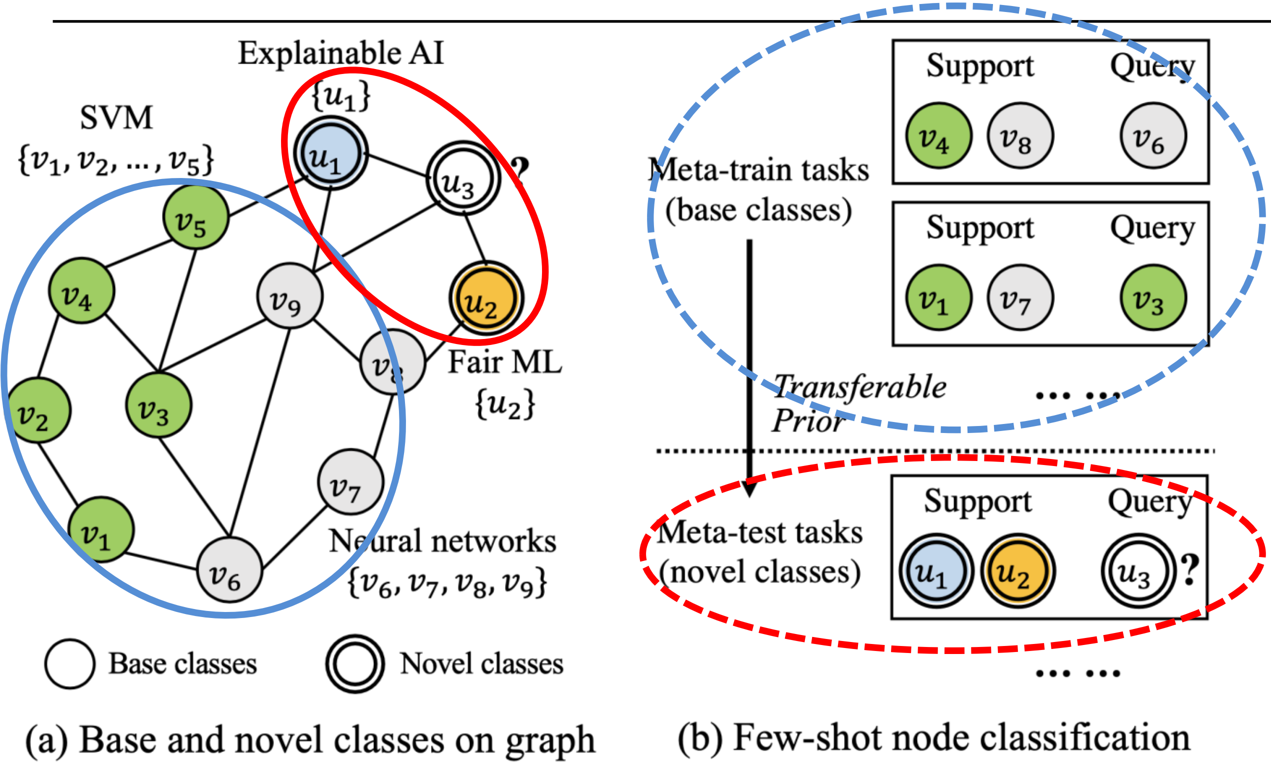


Figure 1: Illustration of few-shot node classification.

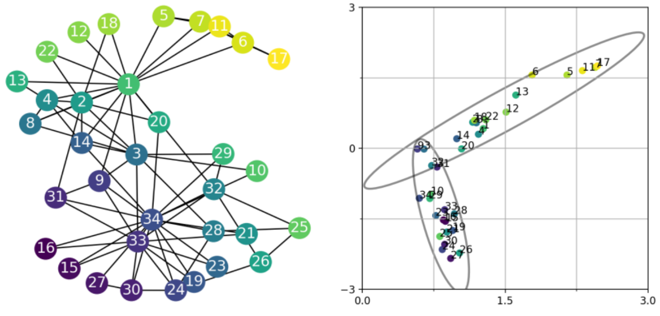
- Base classes
  - SVM
  - Neural networks
- Novel classes
  - Explainable AI
  - Fair ML

Task :

few-shot node classification for novel classes.

Related work: graph representation learning

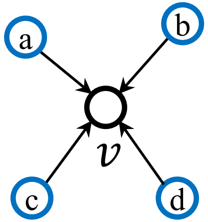
- Graph embedding
  - Capture local structure
  - *e.g.*, Deepwalk [1], node2vec [2]



(a) Karate club graph (b) Visualization based on our model

(Figure credit: [3])

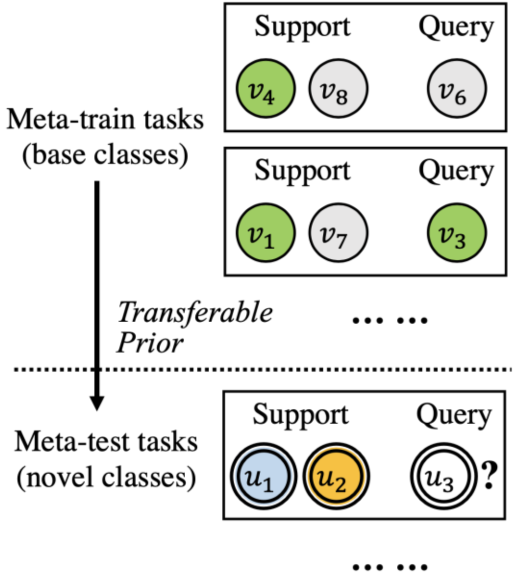
- Graph neural networks
  - Recursive neighborhood aggregation
  - *e.g.*, GCN [4], GAT [5]



[1] Perozzi, B., et al. 2014. Deepwalk: Online learning of social representations. In KDD, 701–710.
[2] Grover, A., et al. 2016. node2vec: Scalable feature learning for networks. In KDD, 855-864.
[3] Cavallari, S., et al. 2017. Learning community embedding with community detection and node embedding on graphs. In CIKM, 377-386
[4] Kipf, T. N., et al. 2017. Semi-supervised classification with graph convolutional networks. ICLR.
[5] Veličković, P., et al. 2018. Graph attention networks. ICLR.

Related work: few-shot node classification

- Few-shot node classification
  - *e.g.*, Meta-GNN [1], GFL [2]
  - Follow meta-learning paradigm
  - Formulate few-shot node classification as a series of classification tasks
  - Learn a transferable prior from meta-train tasks
  - Adapt the prior on meta-test tasks



- What is missing?
  - Graph nodes related to each other and the inter-dependency is crucial to learning node representations
  - These models overlook such inter-dependency in a task, which should be incorporated into the transferable prior

[1] Zhou, F., et al. 2019. Meta-GNN: On Few-shot Node Classification in Graph Meta-learning. In CIKM, 2357-2360
[2] Yao, H., et al. 2020. Graph few-shot learning via knowledge transfer. In AAAI, 6656-6663.

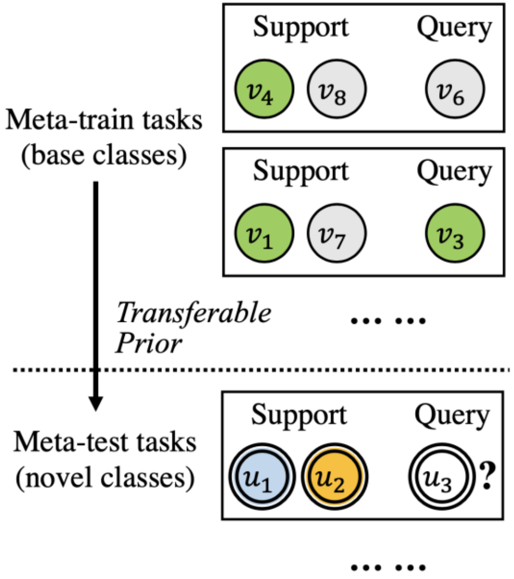
Outline

- Problem definition & related work
- **Challenges & insights**
- Proposed model: RALE
- Experiments
- Conclusions

Challenges

- Goal
  - Exploit additional dependencies based on graph structures to enhance the limited support set.

- Challenges
  - How to capture the potentially long-ranged dependencies between nodes within a task?
  - How to align the dependencies across tasks to converge on a common transferable prior?



Insights

- Hub nodes
  - Structurally important nodes
    - *e.g.*, as measured by network centrality scores such as degree or PageRank [1]
    - Backbone of the graph

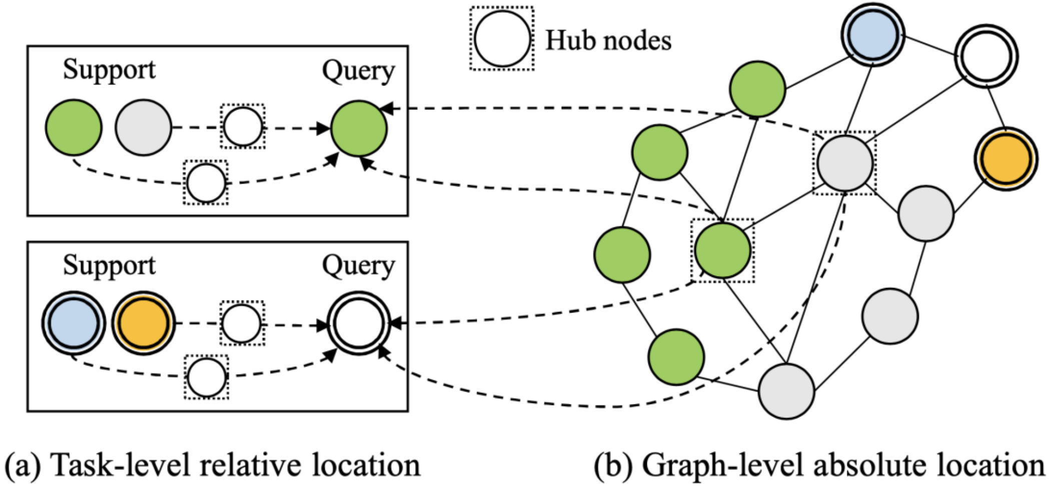


Figure 2: Hub-based relative and absolute locations.

  - Two roles:
    - Utilize hubs for efficient sampling of important paths; **relative location**
    - Utilize hubs as a set of global references, to align the dependencies across tasks; **absolute location**

[1] Page, L., et al. 1999. The PageRank citation ranking: Bringing order to the web. Stanford InfoLab.

Outline

- Problem definition & related work
- Challenges & insights
- **Proposed model: RALE**
- Experiments
- Conclusions

Proposed model: Overview of RALE

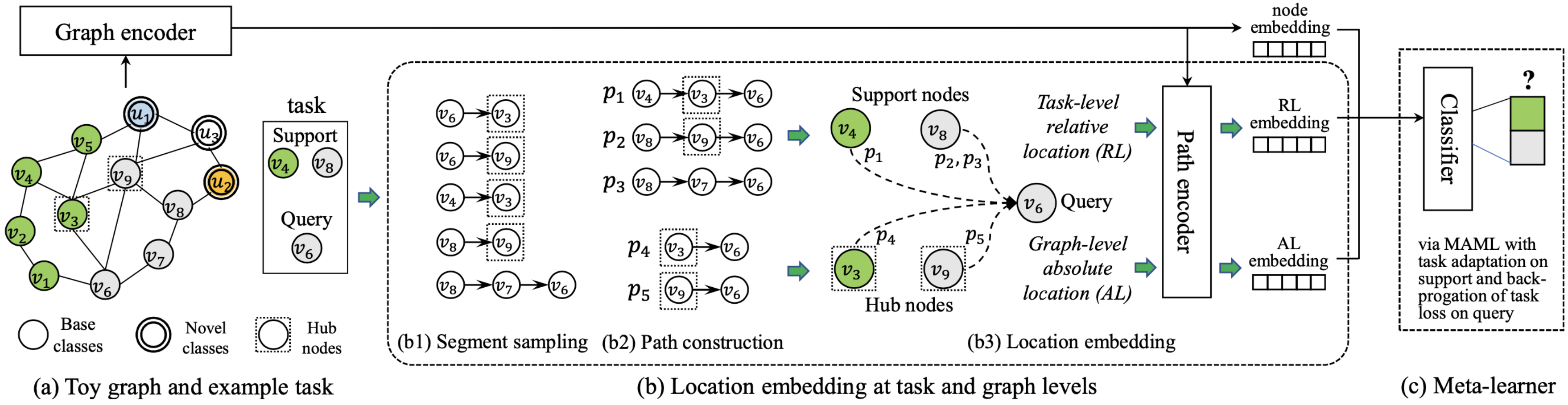


Figure 3: Overview of the proposed model RALE.

Abstraction of Location Embedding

- **Graph encoder** ϕ_g: embed nodes into representations
- **Path encoder** ϕ_p: embed paths into representations
- **Location embedding** for node v *w.r.t.* reference nodes $\mathcal{R}$

$$\mathbf{e}_v^{\{u\}} = \mathrm{AGGR}(\{\phi_p(\phi_g(p;\theta_g);\theta_p) : \forall p \in \mathcal{P}_{u,v}\})$$

$$\begin{aligned}\mathbf{e}_v^{\mathcal{R}} &= \phi(\{\mathcal{P}_{u,v} : u \in \mathcal{R}\};\theta_g,\theta_p) \\ &= \mathrm{AGGR}(\{\mathbf{e}_v^{\{u\}} : \forall u \in \mathcal{R}\}).\end{aligned}$$

Task- and graph-level location embedding

- **Task-level** relative location
  (task $t = (S_t, Q_t)$)

$$\mathbf{e}_v^{S_t} = \phi(\{\mathcal{P}_{s,v} : s \in S_t\}; \theta_g, \theta_p)$$

- **Graph-level** absolute location
  (hubs set $\mathcal{H}$)

$$\mathbf{e}_v^{\mathcal{H}} = \phi(\{\mathcal{P}_{h,v} : h \in \mathcal{H}\}; \theta_g, \theta_p)$$

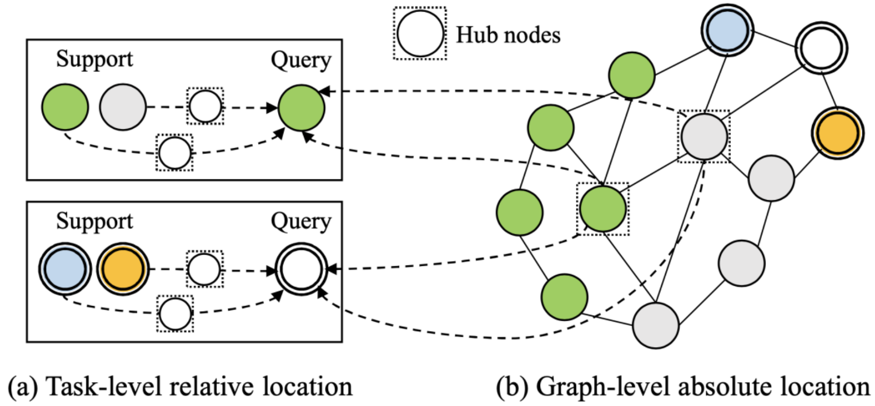


Figure 2: Hub-based relative and absolute locations.

Dependency-aware meta-optimization

- Dependency-aware classification layer

$$\psi(v;\Theta) = \textsc{Softmax}\left(\sigma(\mathbf{W}\left[\mathbf{h}_v \| \mathbf{e}_v^{S_t} \| \mathbf{e}_v^{\mathcal{H}}\right])\right)$$

node emb. relative location emb. absolute location emb.

- Meta-objective (MAML) [1]

$$L(S_t;\Theta) = -\sum_{v\in S_t}\sum_{i=1}^{m}\mathbb{I}_{\ell(v)=i}\ln(\psi(v;\Theta)[i])$$ (Loss on the support)

$$\Theta' = \Theta - \alpha\frac{\partial L(S_t;\Theta)}{\partial\Theta}$$ (Adapting prior Θ to Θ′ by one gradient update)

$$\Theta^* = \arg\min_{\Theta}\sum_{t\in\mathcal{T}_{tr}} L(Q_t;\Theta - \alpha\frac{\partial L(S_t;\Theta)}{\partial\Theta})$$ (Optimization loss on the query)

[1] Finn C, et al. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. ICML.

Outline

- Problem definition & related work
- Challenges & insights
- Proposed model: RALE
- **Experiments**
- Conclusions

Datasets, evaluation and baselines

- Datasets

	Nodes	Edges	Features	Classes (Train/Val/Test)
Amazon	13,381	245,778	767	10 (5/2/3)
Email	909	13,733	128	28 (15/6/7)
Reddit	231,371	11,606,876	602	41 (25/6/10)

- Baselines

GNNs: GCN, GraphSAGE, GAT
  - Training: optimize GNN on all base classes
  - Test: freeze parameters of aggregation layer, update parameters of classification layer

GNN+'s: GCN+, GraphSAGE+, GAT+
  - Training: optimize GNN on all base classes
  - Test: update all parameters

Meta-learning models: Meta-GNN, Proto-GNN
  - Meta-GNN: MAML [1] based model
  - Proto-GNN: prototypical network [2] based model

[1] Finn, C., et al. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. ICML.
[2] Snell, J., et al. 2017. Prototypical networks for few-shot learning. NIPS, 4077-4087.

Empirical results

Table 2: Accuracy (percent) of RALE and baselines. In each row, the best result is bolded and the second best is underlined. RALE's improvement is calculated relative to the best baseline, with the corresponding p-value under two-tail paired t-test.

		GCN	GraphSAGE	GAT	GCN+	GraphSAGE+	GAT+	Meta-GNN	Proto-GNN	RALE	impr.	p-value
Amazon (2-way)	1-shot	71.97	63.19	72.61	70.86	68.36	66.99	<u>73.20</u>	68.91	**78.07**	+6.65%	0.007
	3-shot	<u>78.67</u>	63.15	78.18	72.66	69.75	76.07	74.45	76.41	**84.17**	+6.99%	0.021
	5-shot	79.46	64.71	<u>85.18</u>	79.84	68.44	**85.45**	76.58	79.67	84.50	-1.11%	0.437
Email (5-way)	1-shot	43.15	40.25	43.11	47.15	43.48	40.82	<u>47.21</u>	35.47	**51.82**	+9.76%	< 0.001
	3-shot	53.29	45.56	41.06	50.51	44.28	35.83	<u>55.64</u>	40.93	**59.47**	+6.88%	0.002
	5-shot	58.19	48.38	37.24	58.37	47.01	33.52	<u>58.76</u>	42.38	**65.46**	+11.40%	< 0.001
Reddit (5-way)	1-shot	20.02	41.61	19.95	20.13	35.89	19.99	<u>46.42</u>	44.95	**48.63**	+4.76%	< 0.001
	3-shot	20.16	47.00	20.15	20.21	49.11	20.02	51.28	<u>51.60</u>	**56.85**	+10.17%	< 0.001
	5-shot	20.40	50.53	20.12	20.41	51.30	20.14	<u>53.33</u>	52.57	**57.45**	+7.73%	0.029

- **Observations**
  - GNNs *vs* GNN+'s: no consistent winner emerge
  - Meta-GNN and Proto-GNN achieve comparable or slightly better performance than GNNs and GNN+'s
  - RALE generally outperforms all baselines
  - More shots for support set
    - GNNs and GNN+'s: fluctuate
    - Meta-learning methods: steadily perform better

Ablation study and parameter analysis

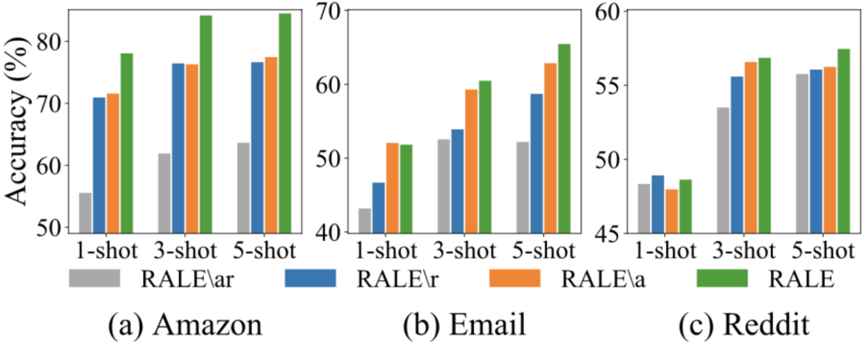


Figure 4: Ablation study.

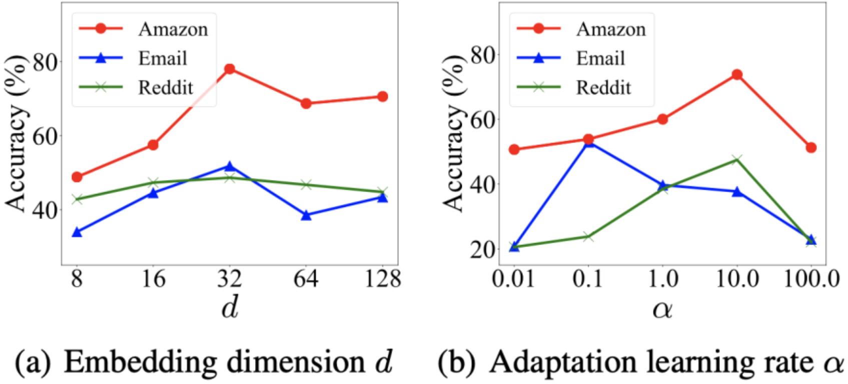


Figure 6: Analysis of parameter sensitivity.

- **Ablation study**
  - RALE\ar has the lowest accuracy
  - RALE\r and RALE\a perform better than RALE\ar; RALE\a is slightly better than RALE\r.
  - Full model RALE achieves the best performance

- **Parameters sensitivity**
  - $d = 32$ appears to be robust across datasets
  - $\alpha \in [0.1, 10]$ is better, though they may have different optimal values due to the various relationships between tasks

Outline

- Problem definition & related work
- Challenges & insights
- Proposed model: RALE
- Experiments
- **Conclusions**

Conclusions

- Problem
  - **Few-shot node classification** on graph
- Motivation
  - Nodes on a graph are **non-i.i.d**
  - Need to capture the node dependencies within tasks
- Proposed model: RALE
  - **hub-based** relative and absolute location embedding
    - **Relative locations** capture the task-level dependency
    - **Absolute locations** capture the graph-level dependency to align the tasks

Thanks!

Zemin Liu, Yuan Fang, Chenghao Liu, Steven C.H. Hoi.

Relative and Absolute Location Embedding for Few-Shot Node Classification on Graph.

In Proceeding of 35th AAAI Conference on Artificial Intelligence, February 2-9, 2021