LESS IS MORE: ONE-SHOT-SUBGRAPH LINK PREDICTION ON LARGE-SCALE KNOWLEDGE GRAPH

Anonymous authors
Paper under double-blind review

Introduction –

- Semantic vs Structural models,
- Motivation -> limit info. needed for prediction,
- One-Shot subgraph Link Prediction,
- Challenges with the approach,
- Their approach.

Contributions –

- Formalize the notion of **one-shot link prediction**,
- Solve a non-trivial, **bi-level optimization problem**,
- Extensive experiments to demonstrate strong performance.

Related Work –

- Semantic models,
- Efficient Semantic models,
- Structural models,
- Sampling based structural models.

One-shot-subgraph Link prediction –

$$\mathcal{G} \xrightarrow{g_{\boldsymbol{\phi}},(u,q)} \mathcal{G}_s \xrightarrow{f_{\boldsymbol{\theta}}} \hat{\boldsymbol{Y}}$$

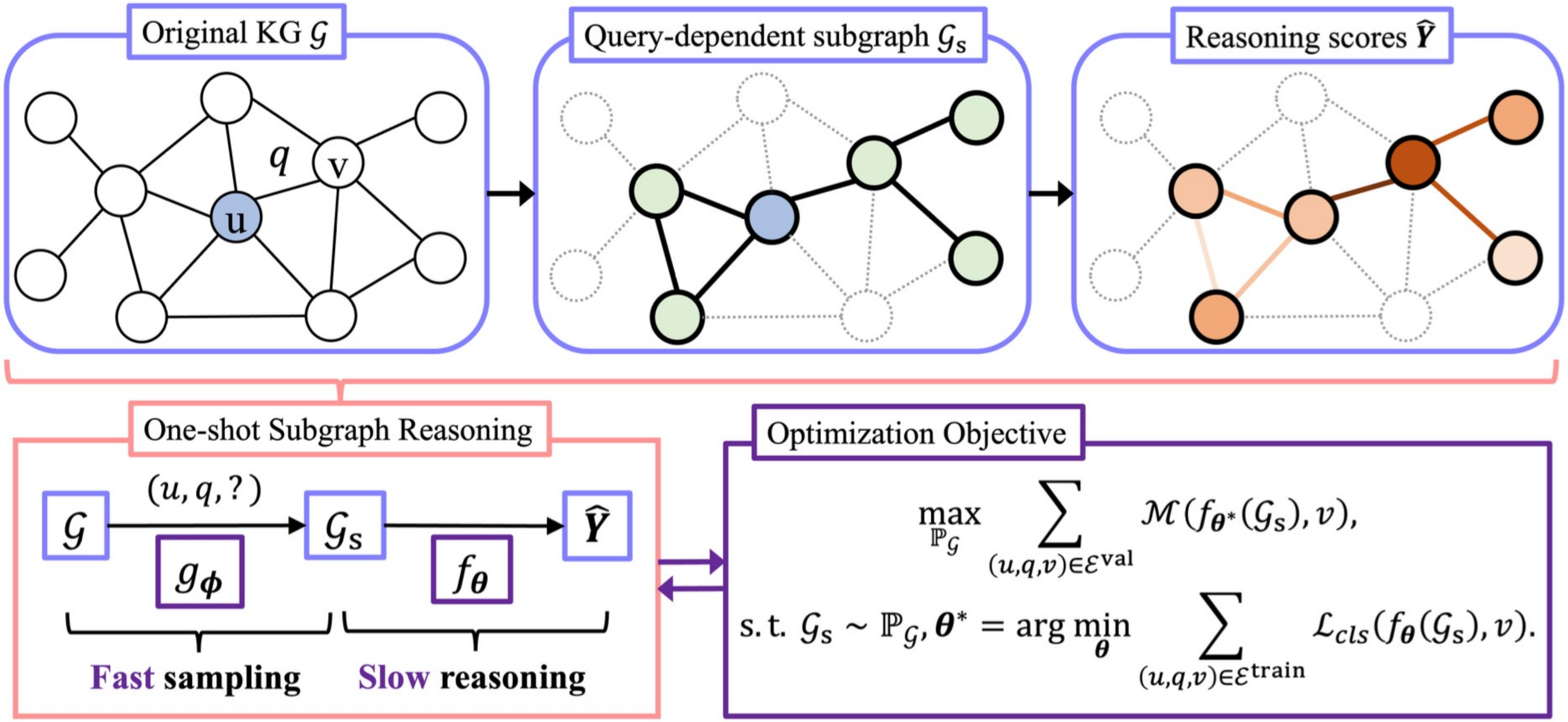
$$\boldsymbol{H} \xrightarrow{f_{\boldsymbol{\theta}},(u,q)} \hat{\boldsymbol{Y}}, \text{ s.t. } \mathcal{G} \xrightarrow{f_{\boldsymbol{\theta}}} \boldsymbol{H}$$

$$\mathcal{G} \xrightarrow{f_{\boldsymbol{\theta}},(u,q)} \hat{\boldsymbol{Y}}$$

$$\mathcal{G} \xrightarrow{f_{\boldsymbol{\theta}}^{(1)},(u,q)} \mathcal{G}_s^{(1)} \xrightarrow{f_{\boldsymbol{\theta}}^{(2)},(u,q)} \mathcal{G}_s^{(2)} \rightarrow \cdots \rightarrow \mathcal{G}_s^{(L-1)} \xrightarrow{f_{\boldsymbol{\theta}}^{(L)},(u,q)} \hat{\boldsymbol{Y}}$$

$$\left\{\hat{\boldsymbol{Y}}_v : \mathcal{G} \xrightarrow{(u,v)} \mathcal{G}_s^{(u,v)} \xrightarrow{f_{\boldsymbol{\theta}},(u,q,v)} \hat{\boldsymbol{Y}}_v\right\}_{v\in\mathcal{V}} \rightarrow \hat{\boldsymbol{Y}}$$

Instantiating *One-shot-subgraph* Link prediction –



Instantiating *One-shot-subgraph* Link prediction –

1. Generate sampling distribution, use PPR!

$$\texttt{Non-parametric indicator}: \boldsymbol{p}^{(k+1)} \leftarrow \alpha \cdot \boldsymbol{s} + (1-\alpha) \cdot \boldsymbol{D}^{-1}\boldsymbol{A} \cdot \boldsymbol{p}^{(k)}$$

2. Extract Subgraph,

$$\texttt{Entity Sampling}: \mathcal{V}_s \leftarrow \texttt{TopK}\Big(\mathcal{V},\ \boldsymbol{p},\ K = r^q_{\mathcal{V}} \times |\mathcal{V}|\Big),$$

$$\texttt{Edge Sampling}: \mathcal{E}_s \leftarrow \texttt{TopK}\Big(\mathcal{E},\ \{\boldsymbol{p}_x \cdot \boldsymbol{p}_o : x, o \in \mathcal{V}_s, (x, r, o) \in \mathcal{E}\},\ K = r^q_{\mathcal{E}} \times |\mathcal{E}|\Big)$$

3. Propagate messages,

$$\texttt{Indicating}: \boldsymbol{h}_o^{(0)} \leftarrow \mathbb{1}(o = u),$$

$$\texttt{Propagation}: \boldsymbol{h}_o^{(\ell+1)} \leftarrow \texttt{DROPOUT}\Big(\texttt{ACT}\Big(\texttt{AGG}\Big\{\texttt{MESS}(\boldsymbol{h}_x^{(\ell)}, \boldsymbol{h}_r^{(\ell)}, \boldsymbol{h}_o^{(\ell)}) : (x, r, o) \in \mathcal{E}_s\Big\}\Big)\Big)$$

Instantiating *One-shot-subgraph* Link prediction –

Algorithm 1 One-shot-subgraph Link Prediction on Knowledge Graph

Require: KG $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{R})$, degree matrix $\boldsymbol{D} \in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|}$, adjacency matrix $\boldsymbol{A} \in \{0,1\}^{|\mathcal{V}|\times|\mathcal{V}|}$, damping coefficient α, maximun PPR iterations K , query $(u, q, ?)$, sampler $g_{\boldsymbol{\phi}}$, predictor $f_{\boldsymbol{\theta}}$.

`# Step1. Generate sampling distribution`

initialize $\boldsymbol{s} \leftarrow \mathbb{1}(u),\ \boldsymbol{p}^{(0)} \leftarrow \mathbb{1}(u)$.

for $k = 1 \ldots K$ **do**

  $\boldsymbol{p}^{(k+1)} \leftarrow \alpha \cdot \boldsymbol{s} + (1-\alpha) \cdot \boldsymbol{D}^{-1}\boldsymbol{A} \cdot \boldsymbol{p}^{(k)}$

end for

`# Step2. Extract a subgraph` $\mathcal{G}_s$

$\mathcal{V}_s \leftarrow \texttt{TopK}(\mathcal{V},\ \boldsymbol{p},\ K = r^q_{\mathcal{V}} \times |\mathcal{V}|)$.

$\mathcal{E}_s \leftarrow \texttt{TopK}(\mathcal{E},\ \{\boldsymbol{p}_u \cdot \boldsymbol{p}_v : u, v \in \mathcal{V}_s, (u, r, v) \in \mathcal{E}\},\ K = r^q_{\mathcal{E}} \times |\mathcal{E}|)$.

`# Step3. Reason on the subgraph`

initialize representations $\boldsymbol{h}_o^{(0)} \leftarrow \mathbb{1}(o = u)$

for $\ell = 1 \ldots L$ **do**

  $\boldsymbol{h}_o^{(\ell)} \leftarrow \texttt{DROPOUT}(\texttt{ACT}(\texttt{AGG}\{\texttt{MESS}(\boldsymbol{h}_x^{(\ell-1)}, \boldsymbol{h}_r^{(\ell-1)}, \boldsymbol{h}_o^{(\ell-1)}) : (x, r, o) \in \mathcal{E}_s\}))$.

end for

return Prediction $\hat{\boldsymbol{y}}_o = \texttt{Readout}(\boldsymbol{h}_o^{(L)}, \boldsymbol{h}_u^{(L)})$ for each entity $o \in \mathcal{V}_s$.

Instantiating *One-shot-subgraph* Link prediction –

- Optimization,

$$\phi^*_{\text{hyper}} = \arg\max_{\phi_{\text{hyper}}} \mathcal{M}(f_{(\theta^*_{\text{hyper}}, \theta^*_{\text{learn}})}, g_{\phi_{\text{hyper}}}, \mathcal{E}^{\text{val}}),$$

$$\text{s.t. } \theta^*_{\text{hyper}} = \arg\max_{\theta_{\text{hyper}}} \mathcal{M}(f_{(\theta_{\text{hyper}}, \theta^*_{\text{learn}})}, g_{\bar{\phi}_{\text{hyper}}}, \mathcal{E}^{\text{val}}),$$

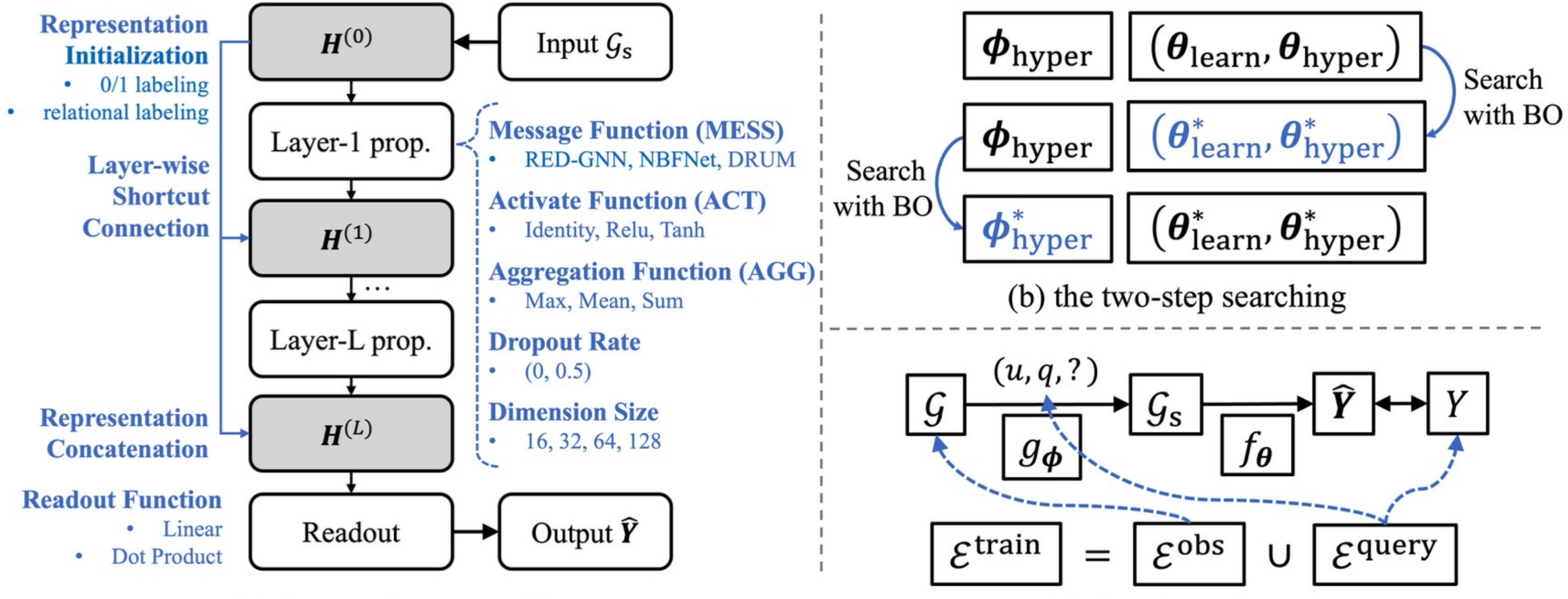
$$\theta^*_{\text{learn}} = \arg\min_{\theta_{\text{learn}}} \mathcal{L}_{cls}(f_{(\theta_{\text{hyper}}, \theta_{\text{learn}})}, g_{\bar{\phi}_{\text{hyper}}}, \mathcal{E}^{\text{train}})$$

- Search Algorithm,

  - First, we freeze the sampler $g_{\bar{\phi}}$ (with constant ϕ_{hyper}) to search for the optimal predictor f_{θ^*} with (1) the hyper-parameters optimization for θ^*_{hyper} and (2) the stochastic gradient descent for θ^*_{learn}.
  - Then, we freeze the predictor f_{θ^*} and search for the optimal sampler g_{ϕ^*}, simplifying to pure hyper-parameters optimization for ϕ^*_{hyper} in a zero-gradient manner with low computation complexity.

Instantiating *One-shot-subgraph* Link prediction –

- Optimization,



(a) the search space of θ_{hyper}

(b) the two-step searching

(c) the edge split trick

Instantiating *One-shot-subgraph* Link prediction –

- Theory,

Theorem 1. *Let $\mathcal{G}_s^{train} \sim \mathbb{P}_{\mathcal{G}}$ and $\mathcal{G}_s^{test} \sim \mathbb{P}_{\mathcal{G}}$ be the training and testing graphs that are sampled from distribution $\mathbb{P}_{\mathcal{G}}$. Consider any two test entities $u, v \in \mathcal{V}_s^{test}$, for which we can make a prediction decision of fact (u, q, v) with the predictor $f_{\boldsymbol{\theta}}$, i.e., $\hat{\boldsymbol{y}}_v = f_{\boldsymbol{\theta}}(\mathcal{G}_s^{test})_v \neq \tau$. Let $\mathcal{G}^{test}$ be large enough to satisfy $\sqrt{|\mathcal{V}_s^{test}|}/\sqrt{\log(2|\mathcal{V}_s^{test}|/p)} \geq 4\sqrt{2}/d_{\min}$, where $d_{\min}$ is the constant of graphon degree (Diaconis & Janson, 2007). Then, for an arbitrary threshold $\tau \in [0, 1]$, the testing subgraph $\mathcal{G}_s^{test}$ satisfies that*

$$\frac{\sqrt{|\mathcal{V}_s^{test}|}}{\sqrt{\log(2|\mathcal{V}_s^{test}|/p)}} \geq \frac{2(C_1 + C_2\|g\|_\infty)}{|f_{\boldsymbol{\theta}}(\mathcal{G}_s^{test})_v - \tau|/L(M^{train})} \tag{6}$$

Experiments –

- Main Results,

type	models	WN18RR			NELL-995			YAGO3-10		
		MRR↑	H@1↑	H@10↑	MRR↑	H@1↑	H@10↑	MRR↑	H@1↑	H@10↑
Semantic Models	ConvE	0.427	39.2	49.8	0.511	44.6	61.9	0.520	45.0	66.0
	QuatE	0.480	44.0	55.1	0.533	46.6	64.3	0.379	30.1	53.4
	RotatE	0.477	42.8	57.1	0.508	44.8	60.8	0.495	40.2	67.0
Structural Models	MINERVA	0.448	41.3	51.3	0.513	41.3	63.7	–	–	–
	DRUM	0.486	42.5	58.6	0.532	46.0	**66.2**	0.531	45.3	67.6
	RNNLogic	0.483	44.6	55.8	0.416	36.3	47.8	0.554	50.9	62.2
	CompGCN	0.479	44.3	54.6	0.463	38.3	59.6	0.489	39.5	58.2
	DPMPN	0.482	44.4	55.8	0.513	45.2	61.5	0.553	48.4	67.9
	NBFNet	<u>0.551</u>	<u>49.7</u>	**66.6**	0.525	45.1	63.9	0.550	47.9	68.3
	RED-GNN	0.533	48.5	62.4	<u>0.543</u>	<u>47.6</u>	<u>65.1</u>	<u>0.559</u>	<u>48.3</u>	<u>68.9</u>
	one-shot-subgraph	**0.567**	**51.4**	**66.6**	**0.547**	**48.5**	<u>65.1</u>	**0.606**	**54.0**	**72.1**

Experiments –

- Main Results,

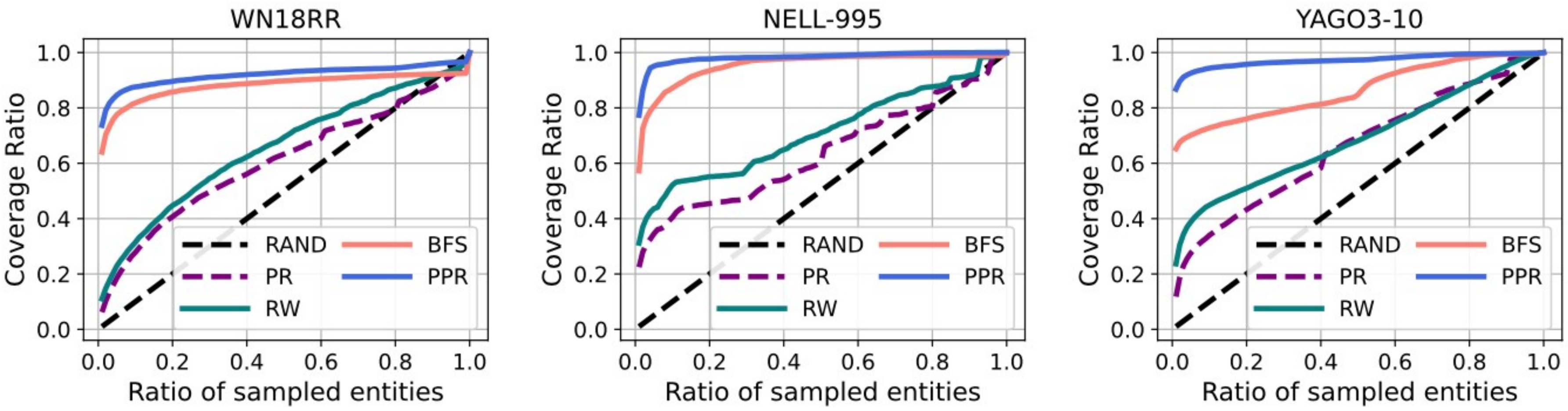
type	models	OGBL-BIOKG Test MRR↑	OGBL-BIOKG Valid MRR↑	OGBL-BIOKG #Params↓	OGBL-WIKIKG2 Test MRR↑	OGBL-WIKIKG2 Valid MRR↑	OGBL-WIKIKG2 #Params↓
Semantic Models	TripleRE	0.8348	0.8360	469,630,002	0.5794	0.6045	500,763,337
	AutoSF	0.8309	0.8317	93,824,000	0.5458	0.5510	500,227,800
	PairRE	0.8164	0.8172	187,750,000	0.5208	0.5423	500,334,800
	ComplEx	0.8095	0.8105	187,648,000	0.4027	0.3759	1,250,569,500
	DistMult	0.8043	0.8055	187,648,000	0.3729	0.3506	1,250,569,500
	RotatE	0.7989	0.7997	187,597,000	0.4332	0.4353	1,250,435,750
	TransE	0.7452	0.7456	187,648,000	0.4256	0.4272	1,250,569,500
Structural Models	**one-shot-subgraph**	**0.8430**	**0.8435**	**976,801**	**0.6755**	**0.7080**	**6,831,201**

Experiments –

- Entity/Relation coverage,

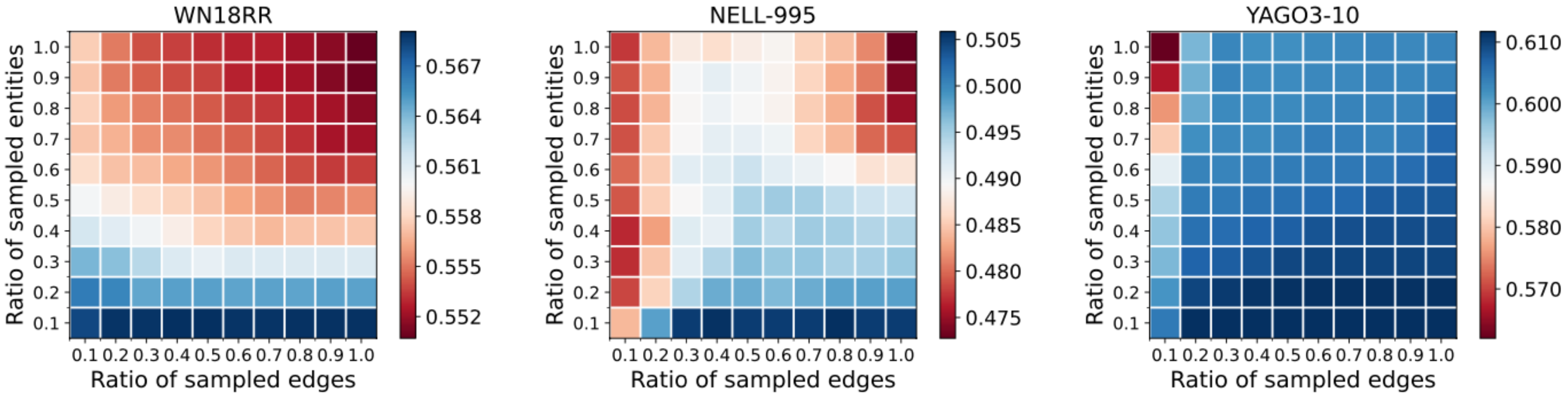
heuristics	WN18RR			NELL-995			YAGO3-10		
	$r_{\mathcal{V}}^q=0.1$	$r_{\mathcal{V}}^q=0.2$	$r_{\mathcal{V}}^q=0.5$	$r_{\mathcal{V}}^q=0.1$	$r_{\mathcal{V}}^q=0.2$	$r_{\mathcal{V}}^q=0.5$	$r_{\mathcal{V}}^q=0.1$	$r_{\mathcal{V}}^q=0.2$	$r_{\mathcal{V}}^q=0.5$
Random Sampling (RAND)	0.100	0.200	0.500	0.100	0.200	0.500	0.100	0.200	0.500
PageRank (PR)	0.278	0.407	0.633	0.405	0.454	0.603	0.340	0.432	0.694
Random Walk (RW)	0.315	0.447	0.694	0.522	0.552	0.710	0.449	0.510	0.681
Breadth-first-searching (BFS)	0.818	0.858	0.898	0.872	0.935	0.982	0.728	0.760	0.848
Personalized PageRank (PPR)	**0.876**	**0.896**	**0.929**	**0.965**	**0.977**	**0.987**	**0.943**	**0.957**	**0.973**

$$\mathrm{CR} = {}^1\!/\!{|\mathcal{E}^{\text{test}}|} \sum_{(u,q,v)\in\mathcal{E}^{\text{test}}} \mathbb{I}\{v \in \mathcal{V}_s\}$$



Experiments –

- Ablation on sampling edges/nodes,



- Ablation subgraph sampling probability distribution method,

heuristics	WN18RR MRR	WN18RR H@1	WN18RR H@10	YAGO3-10 MRR	YAGO3-10 H@1	YAGO3-10 H@10
Random Sampling (RAND)	0.03	43.4	3.5	0.057	5.1	6.5
PageRank (PR)	0.124	11.5	14.2	0.315	28.9	35.9
Random Walk (RW)	0.507	45.8	59.8	0.538	46.3	67.2
Breadth-first-searching (BFS)	0.543	49.6	63.0	0.562	49.4	69.0
Personalized PageRank (PPR)	**0.567**	**51.4**	**66.6**	**0.606**	**54.0**	**72.1**

Experiments –

- Runtime comparison,

phase	$r^q_{\mathcal{V}}$	$r^q_{\mathcal{E}}$	WN18RR Time	WN18RR Memory	NELL-995 Time	NELL-995 Memory	YAGO3-10 Time	YAGO3-10 Memory
Training	1.0	1.0	Out of memory		Out of memory		Out of memory	
	0.5	0.5	26.3m	20.3GB	1.6h	20.1GB	Out of memory	
	0.2	1.0	12.8m	20.2GB	1.2h	18.5GB	Out of memory	
	0.2	0.2	6.7m	6.4GB	0.6h	8.9GB	2.1h	23.1GB
	0.1	1.0	7.2m	9.8GB	0.8h	12.1GB	1.3h	13.9GB
	0.1	0.1	6.6m	5.1GB	0.3h	5.3GB	0.9h	10.2GB
Inference	1.0	1.0	7.3m	6.7GB	17.5m	12.8GB	1.6h	15.0GB
	0.5	0.5	6.0m	4.3GB	8.3m	4.5GB	1.1h	10.1GB
	0.2	1.0	3.2m	5.8GB	4.2m	12.1GB	0.7h	14.7GB
	0.2	0.2	2.8m	1.9GB	3.6m	2.5GB	0.6h	3.7GB
	0.1	1.0	2.7m	2.7GB	3.1m	9.4GB	0.4h	9.7GB
	0.1	0.1	2.3m	1.7GB	2.9m	1.9GB	0.4h	3.1GB

Experiments –

- Extra,

Table 6: Comparison of prediction performance with two recent GNN methods.

methods	WN18RR				YAGO3-10			
	MRR	H@1	H@10	Time	MRR	H@1	H@10	Time
NBFNet (100% entities)	0.551	49.7	66.6	32.3 min	0.550	47.9	68.3	493.8 min
NBFNet + **one-shot-subgraph (10% entities)**	**0.554**	**50.5**	**66.3**	**2.6 min**	**0.565**	**49.6**	**69.2**	**28.2 min**
RED-GNN (100% entities)	0.533	48.5	62.4	68.7 min	0.559	48.3	68.9	1382.9 min
RED-GNN + **one-shot-subgraph (10% entities)**	**0.567**	**51.4**	**66.6**	**4.5 min**	**0.606**	**54.0**	**72.1**	**76.3 min**