Artificial

Knowledge Graph Reasoning with Relational Digraph

Yongqi Zhang
4Paradigm Inc.
Beijing, China
zhangyongqi@4paradigm.com

Quanming Yao
EE, Tsinghua University
Beijing, China
qyaoaa@tsinghua.edu.cn

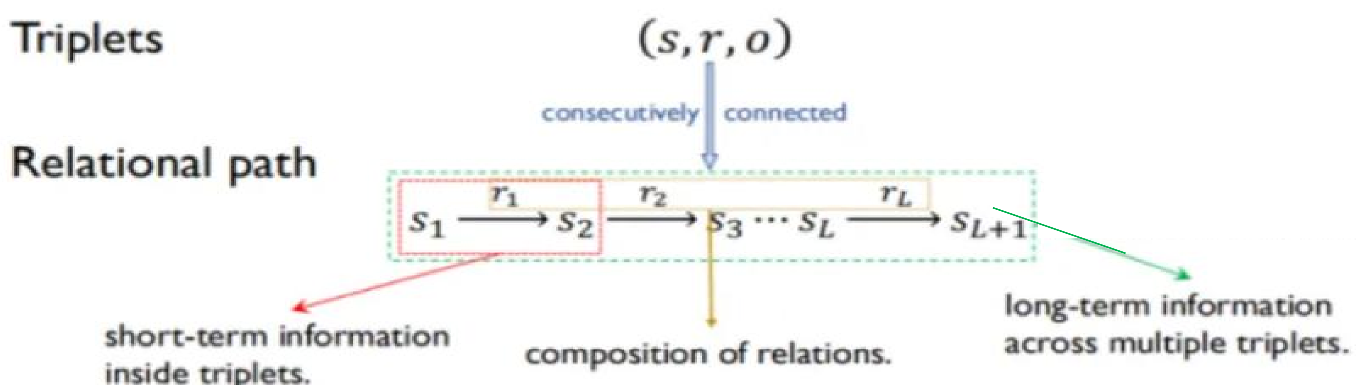
2022.11.28 • ChongQing

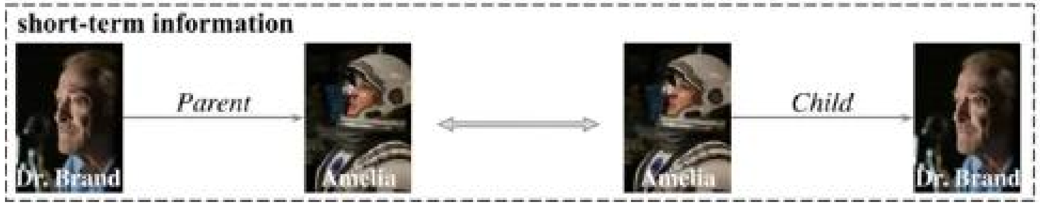
—— WWW22

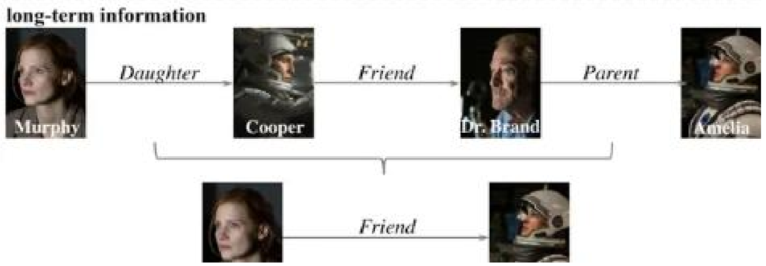
Reported by Qiang Cheng

Introduction

Relational path





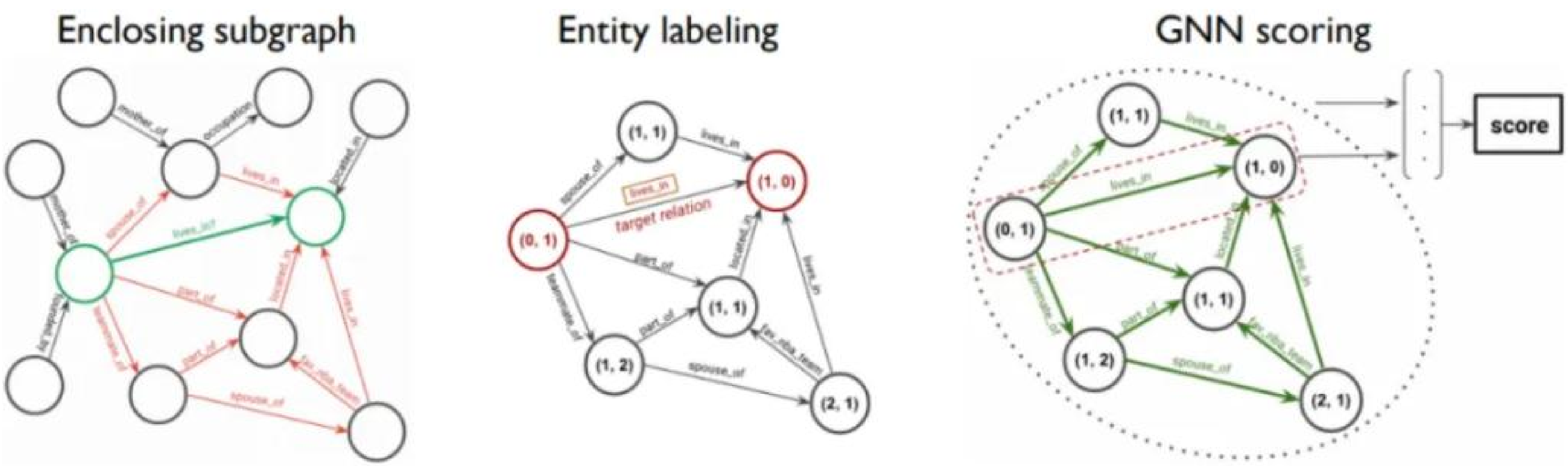


the triples are independently learned,
they cannot explicitly capture the local evidence

PathCon samples all the paths connecting the two entities to predict the relation between them, which is expensive for the reasoning task (*eq*,*rq*, *ea*).

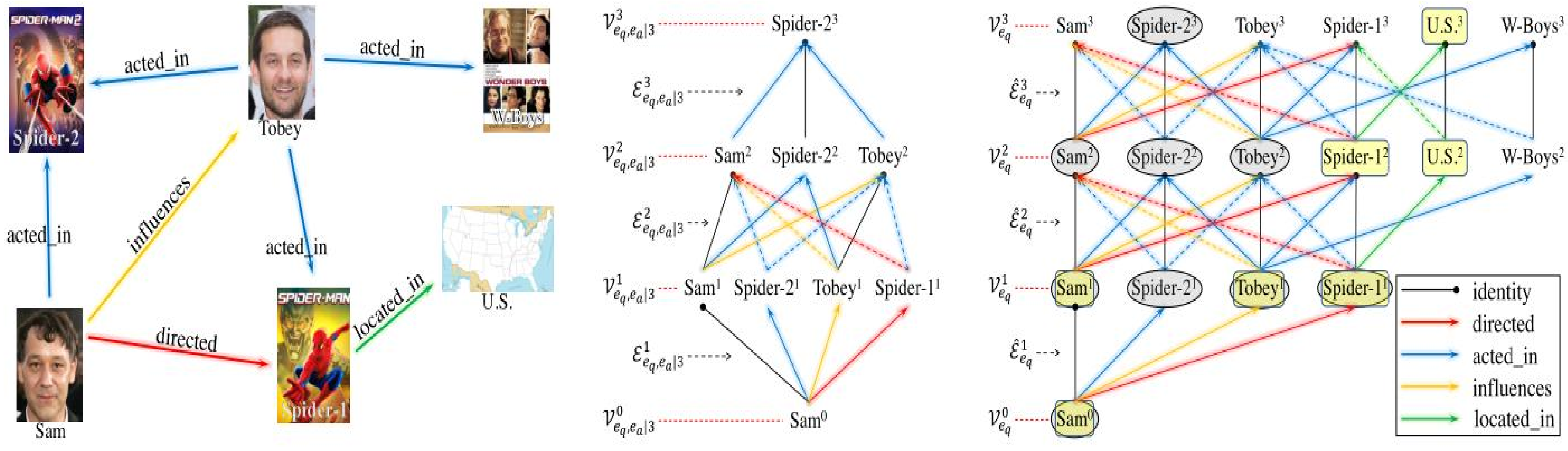
Introduction

GraIL [Teru et. al. 2020]



Learned attention weights are not interpretable，the computation cost is very high

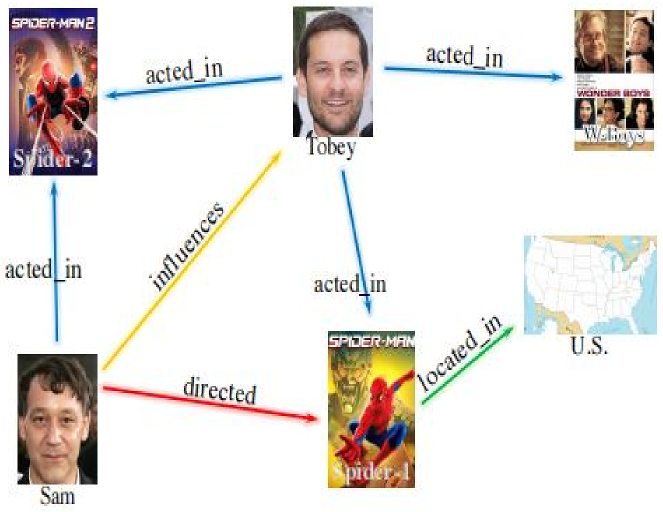
Overview



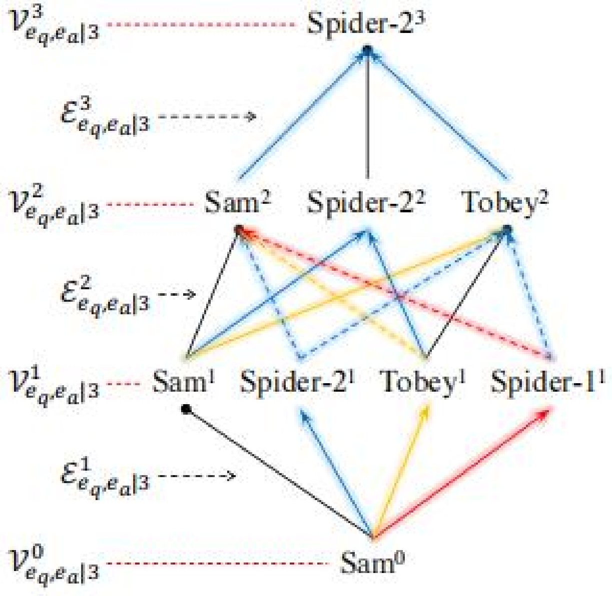
(a) Knowledge graph. (b) $\mathcal{G}_{\text{Sam,Spider-2}|3}$. (c) Recursive encoding.

Figure 1: Graphical illustration. In (c), the subgraph formed by the gray ellipses is $\mathcal{G}_{Sam,\ Spider\text{-}2|3}$ and the subgraph formed by the yellow rectangles is $\mathcal{G}_{Sam,U.S.|3}$. Dashed edges mean the reverse relations of corresponding color (best viewed in color).

Method



(a) Knowledge graph.

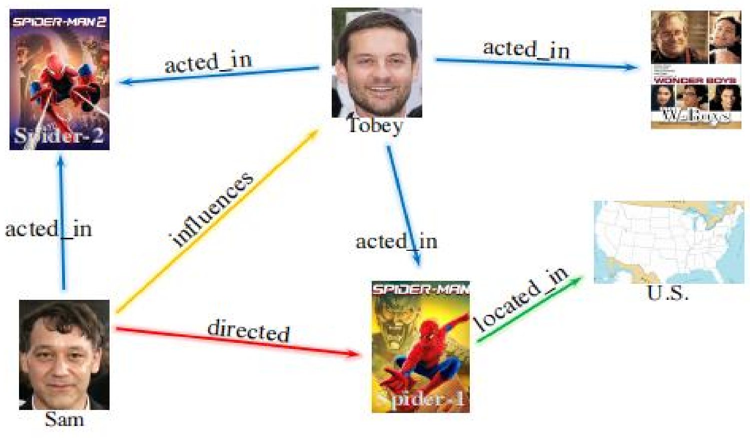


(b) $\mathcal{G}_{\text{Sam,Spider-2}|3}$.

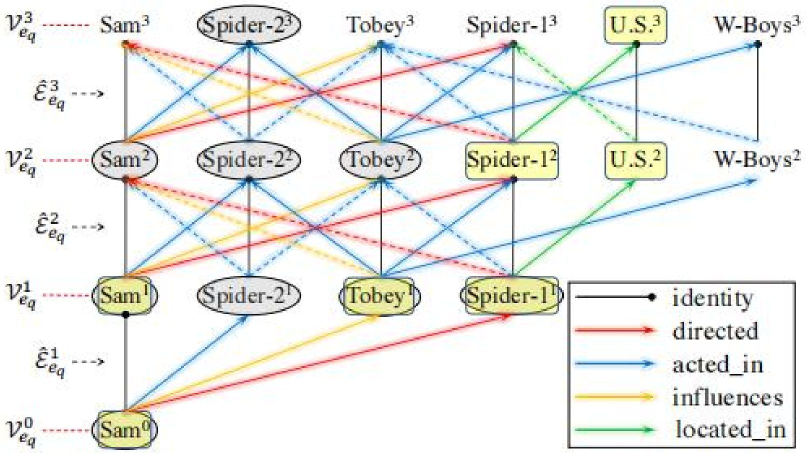
Algorithm 1 RED-Simp: Message passing on single r-digraph

initialize $h^0_{e_q}(e_q, r_q) = \mathbf{0}$ and the entity sets $\mathcal{V}^0_{e_q} = \{e_q\}$, $\mathcal{V}^0_{e_a} = \{e_a\}$;
for $\ell = 1, 2, \ldots, L$ **do**
  get the ℓ-hop out-edges $\hat{\mathcal{E}}^\ell_{e_q} = \{(e_s, r, e_o) \in \mathcal{F} | e_s \in \mathcal{V}^{\ell-1}_{e_q}\}$
  and entities $\mathcal{V}^\ell_{e_q} = \{e_o | (e_s, r, e_o) \in \hat{\mathcal{E}}^\ell_{e_q}\}$ of e_q;
  get the ℓ-hop in-edges $\check{\mathcal{E}}^\ell_{e_a} = \{(e_s, r, e_o) \in \mathcal{F} | e_o \in \mathcal{V}^{\ell-1}_{e_a}\}$
  and entities $\mathcal{V}^\ell_{e_a} = \{e_s | (e_s, r, e_o) \in \check{\mathcal{E}}^\ell_{e_a}\}$ of e_a;
end for
for $\ell = 0, 1, \ldots, L$ **do**
  intersection $\mathcal{E}^\ell_{e_q,e_a|L} = \hat{\mathcal{E}}^\ell_{e_q} \cap \check{\mathcal{E}}^{L-\ell}_{e_a}$ and $\mathcal{V}^\ell_{e_q,e_a|L} = \mathcal{V}^\ell_{e_q} \cap \mathcal{V}^{L-\ell}_{e_a}$;
end for
if $\mathcal{V}^L_{e_q,e_a|L} = \emptyset$ **return** $h^L_{e_a}(e_q, r_q) = \mathbf{0}$;
for $\ell = 1, 2, \ldots, L$ **do**
  message passing for entities $e_o \in \mathcal{V}^\ell_{e_q,e_a|L}$:
  $h^\ell_{e_o}(e_q, r_q) = \delta\Big(W^\ell \cdot \sum_{(e_s,r,e_o)\in\mathcal{E}^\ell_{e_q,e_a|L}} \phi\big(h^{\ell-1}_{e_s}(e_q, r_q), h^\ell_r\big)\Big)$;
end for
return $h^L_{e_a}(e_q, r_q)$.

Method



(a) Knowledge graph.



(c) Recursive encoding.

Algorithm 2 RED-GNN: recursive r-digraph encoding.

initialize $\boldsymbol{h}^0_{e_q}(e_q, r_q) = \boldsymbol{0}$ and the entity set $\mathcal{V}^0_{e_q} = \{e_q\}$;

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

  collect the ℓ-hop edges $\hat{\mathcal{E}}^\ell_{e_q} = \{(e_s, r, e_o) \in \mathcal{F} | e_s \in \mathcal{V}^{\ell-1}_{e_q}\}$ and entities $\mathcal{V}^\ell_{e_q} = \{e_o | (e_s, r, e_o) \in \hat{\mathcal{E}}^\ell_{e_q}\}$;

  message passing for entities $e_o \in \mathcal{V}^\ell_{e_q}$:

  $\boldsymbol{h}^\ell_{e_o}(e_q, r_q) = \delta\Big(\boldsymbol{W}^\ell \cdot \sum_{(e_s, r, e_o) \in \hat{\mathcal{E}}^\ell_{e_q}} \phi\big(\boldsymbol{h}^{\ell-1}_{e_s}(e_q, r_q), \boldsymbol{h}^\ell_r\big)\Big)$;

end for

assign $\boldsymbol{h}^L_{e_a}(e_q, r_q) = \boldsymbol{0}$ for all $e_a \notin \mathcal{V}^L_{e_q}$;

return $\boldsymbol{h}^L_{e_a}(e_q, r_q)$ for all $e_a \in \mathcal{V}$.

Method

message passing

$$\boldsymbol{h}_{e_o}^{\ell} = \delta\Big(\boldsymbol{W}^{\ell} \cdot \sum\nolimits_{(e_s,r,e_o)\in\mathcal{F}} \phi(\boldsymbol{h}_{e_s}^{\ell-1}, \boldsymbol{h}_r^{\ell})\Big), \quad (1)$$

$$\mathcal{G}_{e_q,e_o|\ell} = \cup_{(e_s,r,e_o)\in\hat{\mathcal{E}}_{e_q}^{\ell}} \mathcal{G}_{e_q,e_s|\ell-1} \otimes \left\{(e_s,r,e_o) \in \hat{\mathcal{E}}_{e_q}^{\ell}\right\}. \quad (2)$$

attention mechanism

$$\boldsymbol{h}_{e_o}^{\ell}(e_q,r_q) = \delta\Big(\boldsymbol{W}^{\ell} \cdot \sum\nolimits_{(e_s,r,e_o)\in\hat{\mathcal{E}}_{e_q}^{\ell}} \alpha_{e_s,r,e_o|r_q}^{\ell}(\boldsymbol{h}_{e_s}^{\ell-1}(e_q,r_q) + \boldsymbol{h}_r^{\ell})\Big), \quad (3)$$

$$\alpha_{e_s,r,e_o|r_q}^{\ell} = \sigma\left((\boldsymbol{w}_{\alpha}^{\ell})^{\top} \mathrm{ReLU}\left(\boldsymbol{W}_{\alpha}^{\ell} \cdot (\boldsymbol{h}_{e_s}^{\ell-1}(e_q,r_q) \oplus \boldsymbol{h}_r^{\ell} \oplus \boldsymbol{h}_{r_q}^{\ell})\right)\right), \quad (4)$$

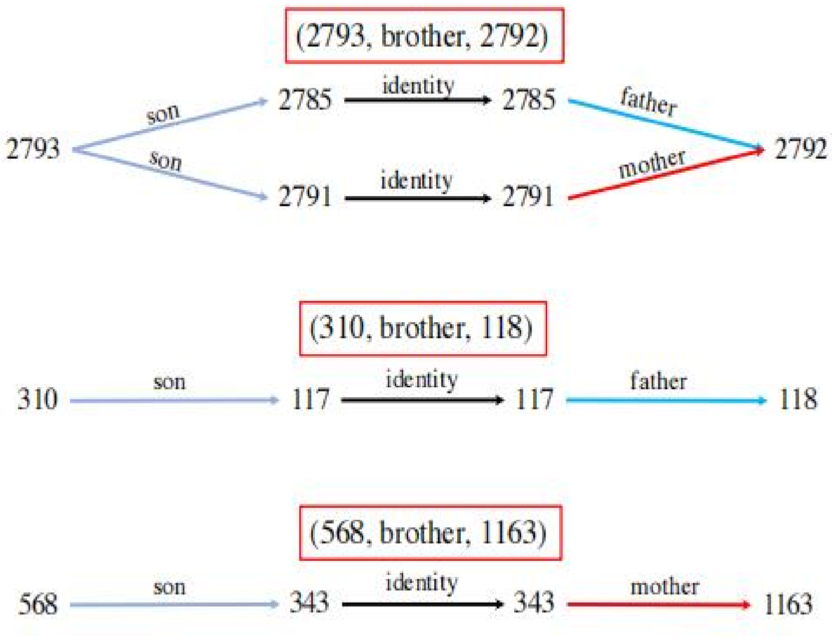
scoring function

$$f(e_q, r_q, e_a) = \boldsymbol{w}^{\top} \boldsymbol{h}_{e_a}^{L}(e_q, r_q), \quad (5)$$

multi-class
log-loss

$$\sum\nolimits_{(e_q,r_q,e_a)\in\mathcal{T}_{\text{tra}}} \Big(-f(e_q,r_q,e_a) + \log\Big(\sum\nolimits_{\forall e\in\mathcal{V}} e^{f(e_q,r_q,e)}\Big)\Big). \quad (6)$$

Method



Algorithm 3 Visualization.

Require: entities $\mathcal{V}$, triples $\mathcal{F}$, query triple (e_q, r_q, e_a), depth L, parameters Θ, threshold θ.

Run Algorithm 2 to obtain the attention weights $\alpha^{\ell}_{e_s,r,e_o|r_q}$, $\ell = 1 \ldots L$ on the edges in each layer.

Initialize $\mathcal{V}^{L}_{e_q,e_a|L}(\theta) = \{e_a\}$.

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

  collect the edges $\mathcal{E}^{\ell}_{e_q,e_a|L}(\theta) = \{(e_s, r, e_o) | e_o \in \mathcal{V}^{L}_{e_q,e_a|L}, \alpha^{\ell}_{e_s,r,e_o|r_q} \geq \theta\}$.

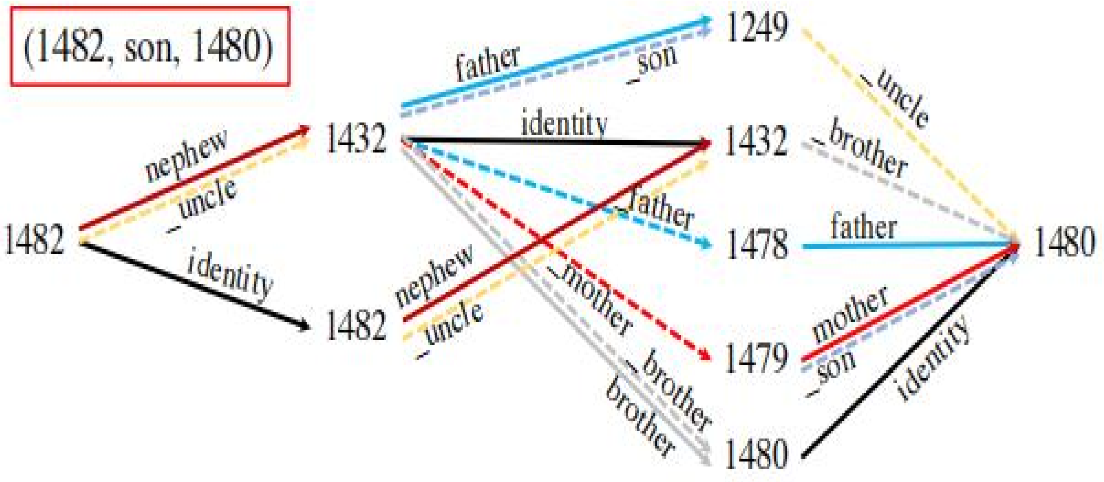
  collect the entities $\mathcal{V}^{\ell-1}_{e_q,e_a|L}(\theta) = \{e_s | (e_s, r, e_o) \in \mathcal{E}^{\ell}_{e_q,e_a|L}(\theta)\}$.

end for

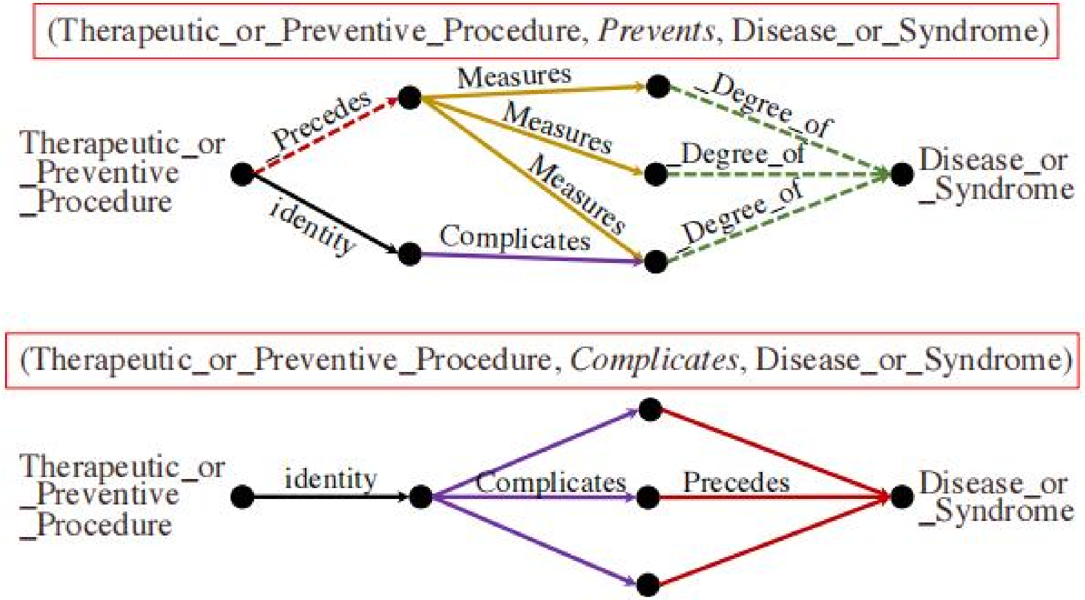
return $\mathcal{G}_{e_q,e_a|L}(\theta) = \mathcal{E}^{1}_{e_q,e_a|L}(\theta) \otimes \mathcal{E}^{2}_{e_q,e_a|L}(\theta) \cdots \otimes \mathcal{E}^{L}_{e_q,e_a|L}(\theta)$.

Figure 5: Visualization of the learned structure on Family.

Method



(a) Family.



(b) UMLS.

Figure 3(a) shows one triple that DRUM fails. As shown, inferring id-1482 as the son of id-1480 requires the knowledge that id-1480 is the only brother of the uncle id-1432 .

Figure 3: Visualization of the learned structures. Dashed lines mean inverse relations. The query triples are indicated by the red rectangles. Due to space limitation, entities in UMLS dataset are shown as black circles (best viewed in color).

Experiments

Table 1: Inductive reasoning. Best performance is indicated by the bold face numbers.

		WN18RR				FB15k-237				NELL-995			
		V1	V2	V3	V4	V1	V2	V3	V4	V1	V2	V3	V4
MRR	RuleN	.668	.645	.368	.624	.363	.433	.439	.429	.615	.385	.381	.333
	Neural LP	.649	.635	.361	.628	.325	.389	.400	.396	.610	.361	.367	.261
	DRUM	.666	.646	.380	.627	.333	.395	.402	.410	.628	.365	.375	.273
	GraIL	.627	.625	.323	.553	.279	.276	.251	.227	.481	.297	.322	.262
	RED-GNN	**.701**	**.690**	**.427**	**.651**	**.369**	**.469**	**.445**	**.442**	**.637**	**.419**	**.436**	**.363**
Hit@1 (%)	RuleN	63.5	61.1	34.7	59.2	30.9	34.7	34.5	33.8	**54.5**	30.4	30.3	24.8
	Neural LP	59.2	57.5	30.4	58.3	24.3	28.6	30.9	28.9	50.0	24.9	26.7	13.7
	DRUM	61.3	59.5	33.0	58.6	24.7	28.4	30.8	30.9	50.0	27.1	26.2	16.3
	GraIL	55.4	54.2	27.8	44.3	20.5	20.2	16.5	14.3	42.5	19.9	22.4	15.3
	RED-GNN	**65.3**	**63.3**	**36.8**	**60.6**	**30.2**	**38.1**	**35.1**	**34.0**	52.5	**31.9**	**34.5**	**25.9**
Hit@10 (%)	RuleN	73.0	69.4	40.7	68.1	44.6	59.9	60.0	60.5	76.0	51.4	53.1	48.4
	Neural LP	77.2	74.9	47.6	70.6	46.8	58.6	57.1	59.3	**87.1**	56.4	57.6	53.9
	DRUM	77.7	74.7	47.7	70.2	47.4	59.5	57.1	59.3	87.3	54.0	57.7	53.1
	GraIL	76.0	77.6	40.9	68.7	42.9	42.4	42.4	38.9	56.5	49.6	51.8	50.6
	RED-GNN	**79.9**	**78.0**	**52.4**	**72.1**	**48.3**	**62.9**	**60.3**	**62.1**	86.6	**60.1**	**59.4**	**55.6**

Experiments

Table 2: Transductive reasoning. Best performance is indicated by the bold face numbers. '-' means unavailable results and results for methods with '*' are copied from the original papers.

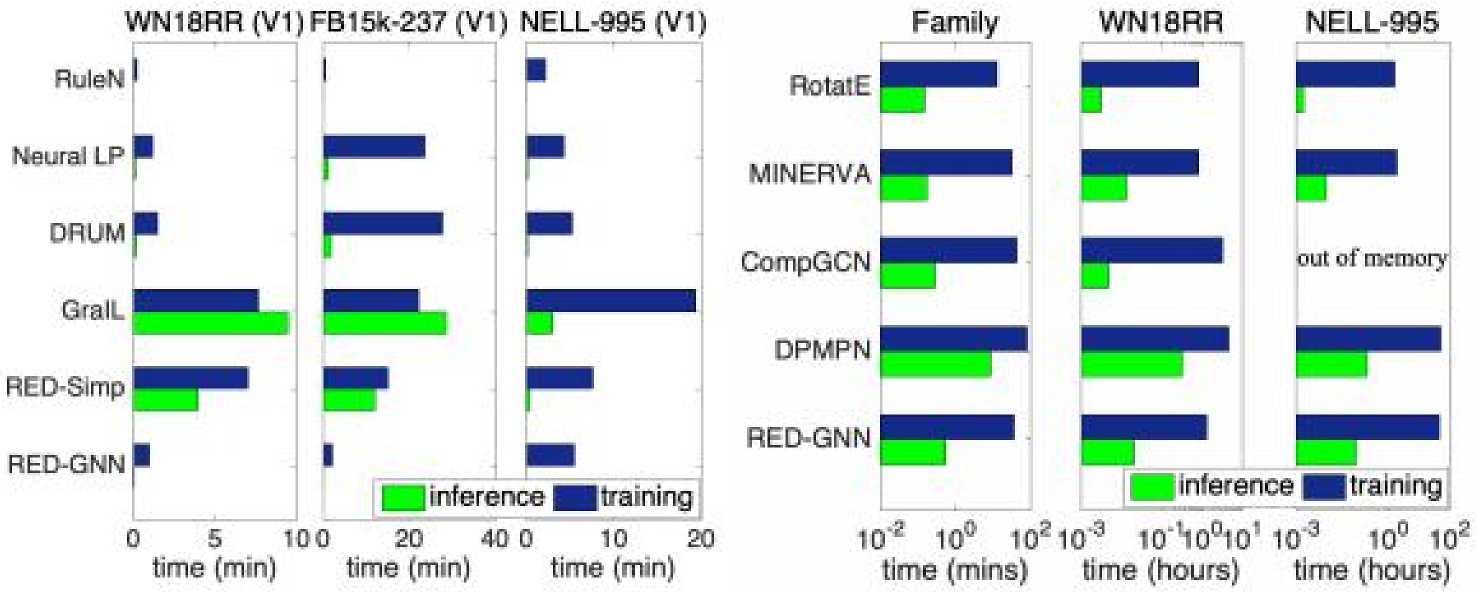
type	models	Family			UMLS			WN18RR			FB15k-237			NELL-995		
		MRR	Hit@1	Hit@10	MRR	Hit@1	Hit@10	MRR	Hit@1	Hit@10	MRR	Hit@1	Hit@10	MRR	Hit@1	Hit@10
triple	ConvE*	-	-	-	.94	92.	96.	.43	39.	49.	.325	23.7	50.1	-	-	-
	RotatE	.921	86.6	98.8	.925	86.3	99.3	.477	42.8	57.1	.337	24.1	53.3	.508	44.8	60.8
	QuatE	.941	89.6	99.1	.944	90.5	99.3	.480	44.0	55.1	.350	25.6	53.8	.533	46.6	64.3
path	MINERVA	.885	82.5	96.1	.825	72.8	96.8	.448	41.3	51.3	.293	21.7	45.6	.513	41.3	63.7
	Neural LP	.924	87.1	99.4	.745	62.7	91.8	.435	37.1	56.6	.252	18.9	37.5	out of memory		
	DRUM	.934	88.1	99.6	.813	67.4	97.6	.486	42.5	58.6	.343	25.5	51.6	out of memory		
	RNNLogic*	–	–	–	.842	77.2	96.5	.483	44.6	55.8	.344	25.2	53.0	–	–	–
GNN	pLogicNet*	–	–	–	.842	77.2	96.5	.441	39.8	53.7	.332	23.7	52.8	–	–	–
	CompGCN	.933	88.3	99.1	.927	86.7	**99.4**	.479	44.3	54.6	.355	26.4	53.5	out of memory		
	DPMPN	.981	97.4	98.1	.930	89.9	98.2	.482	44.4	55.8	.369	**28.6**	53.0	.513	45.2	61.5
	RED-GNN	**.992**	**98.8**	**99.7**	**.964**	**94.6**	99.0	**.533**	**48.5**	**62.4**	**.374**	28.3	**55.8**	**.543**	**47.6**	**65.1**

Experiments

Table 3: Statistics of transductive reasoning datasets. Note that NELL-995* is different as the version in [9] since the training triples contains valid and test triples there.

	$\lvert\mathcal{V}\rvert$	$\lvert\mathcal{R}\rvert$	$\lvert\mathcal{F}\rvert$	$\lvert\mathcal{T}_{\text{val}}\rvert$	$\lvert\mathcal{T}_{\text{tst}}\rvert$
Family	3,007	12	23,483	2,038	2,835
UMLS	135	46	5,327	569	633
WN18RR	40,943	11	86,835	3,034	3,134
FB15k-237	14,541	237	272,115	17,535	20,466
NELL-995*	74,536	200	149,678	543	2,818

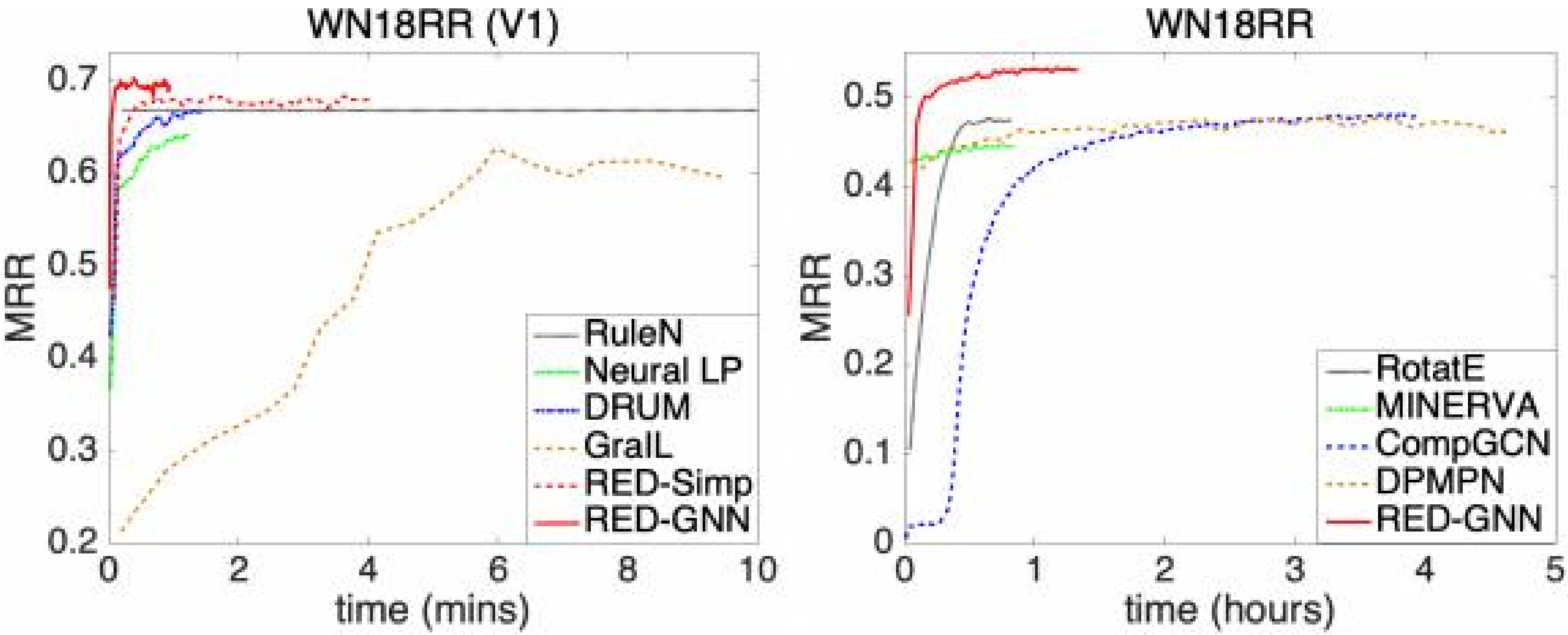
Experiments



(a) Running time.

Figure 2: Running time analysis and learning curve. Left: inductive setting; Right: transductive setting.

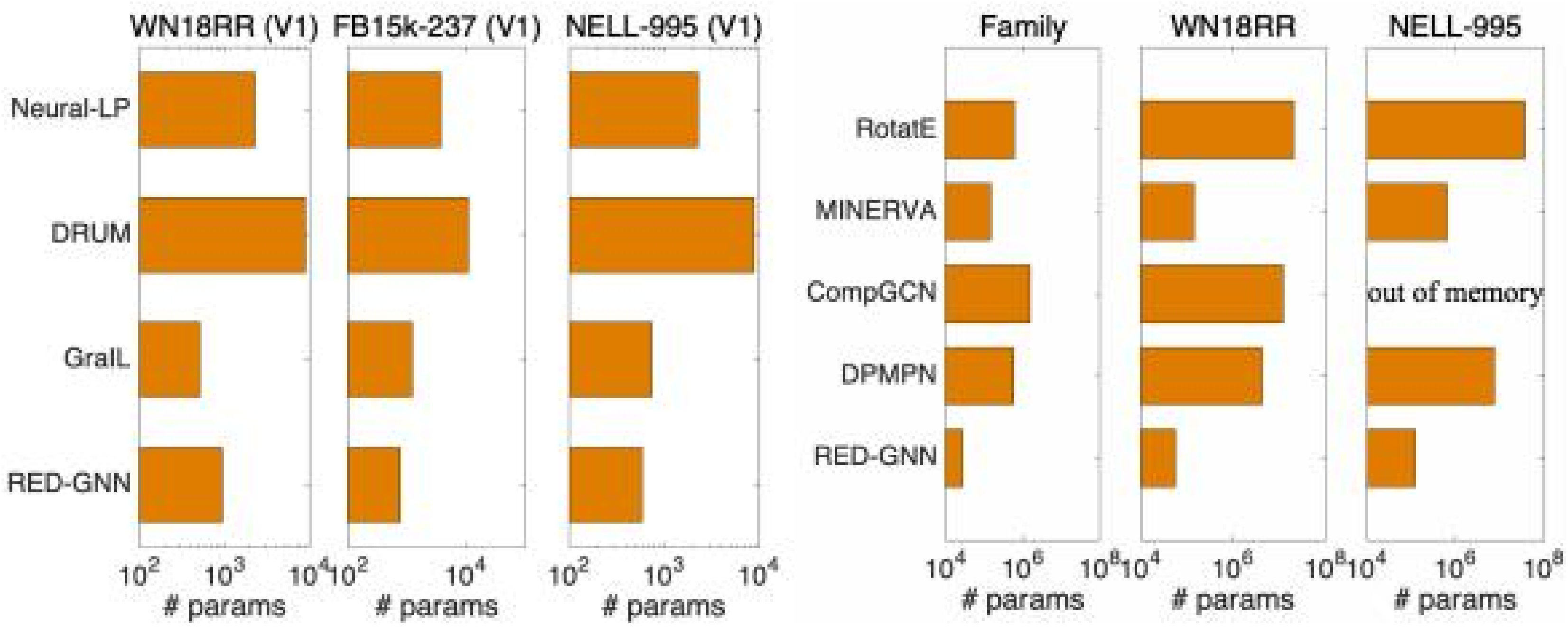
Experiments



(b) Learning curve.

Figure 2: Running time analysis and learning curve. Left: inductive setting; Right: transductive setting.

Experiments



(c) Model parameters.

Figure 2: Running time analysis and learning curve. Left: inductive setting; Right: transductive setting.

Experiments

Table 4: Comparison of different variants of RED-GNN.

methods	WN18RR (V1) MRR	WN18RR (V1) H@10	FB15k-237 (V1) MRR	FB15k-237 (V1) H@10	NELL (V1) MRR	NELL (V1) H@10
Attn-w.o.-r_q	.659	78.3	.268	37.6	.517	73.4
RED-Simp	.683	79.6	.311	45.3	.563	75.8
RED-GNN	.701	79.9	.369	48.3	.637	86.6

Experiments

Table 4: Comparison of different variants of RED-GNN.

methods	WN18RR (V1) MRR	WN18RR (V1) H@10	FB15k-237 (V1) MRR	FB15k-237 (V1) H@10	NELL (V1) MRR	NELL (V1) H@10
Attn-w.o.-r_q	.659	78.3	.268	37.6	.517	73.4
RED-Simp	.683	79.6	.311	45.3	.563	75.8
RED-GNN	.701	79.9	.369	48.3	.637	86.6

Experiments

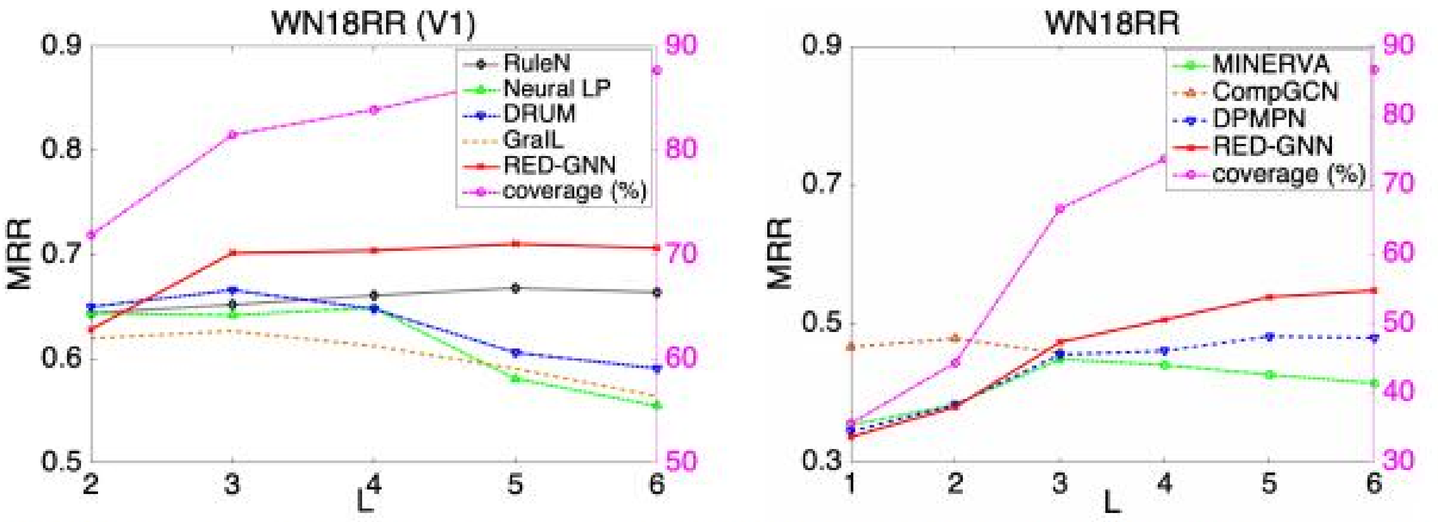


Figure 4: The MRR performance with different L and coverage of triples within L steps.

Experiments

Table 5: The per-distance evaluation for MRR on WN18RR v.s. the length of shortest path.

distance	1	2	3	4	5	>5
ratios (%)	34.9	9.3	21.5	7.5	8.9	17.9
CompGCN	.993	.327	.337	.062	.061	.016
DPMPN	.982	.381	.333	.102	.057	.001
RED-GNN	.993	.563	.536	.186	.089	.005

Thanks!