Knowledge Graph

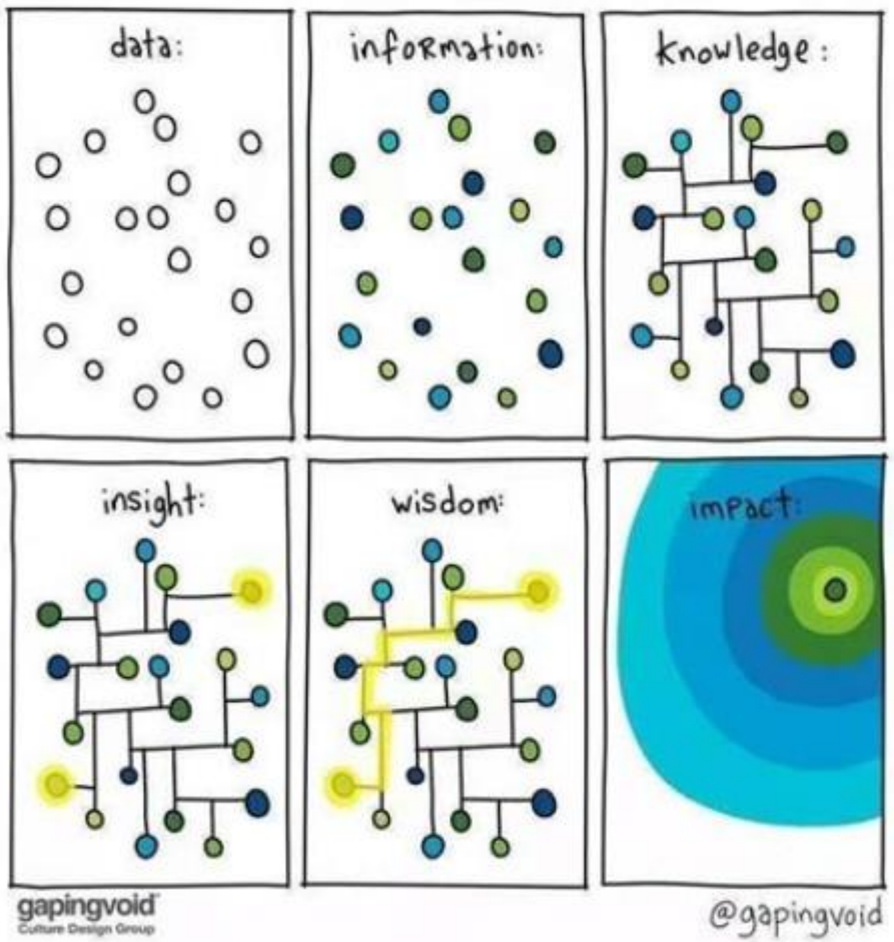
a brief intro

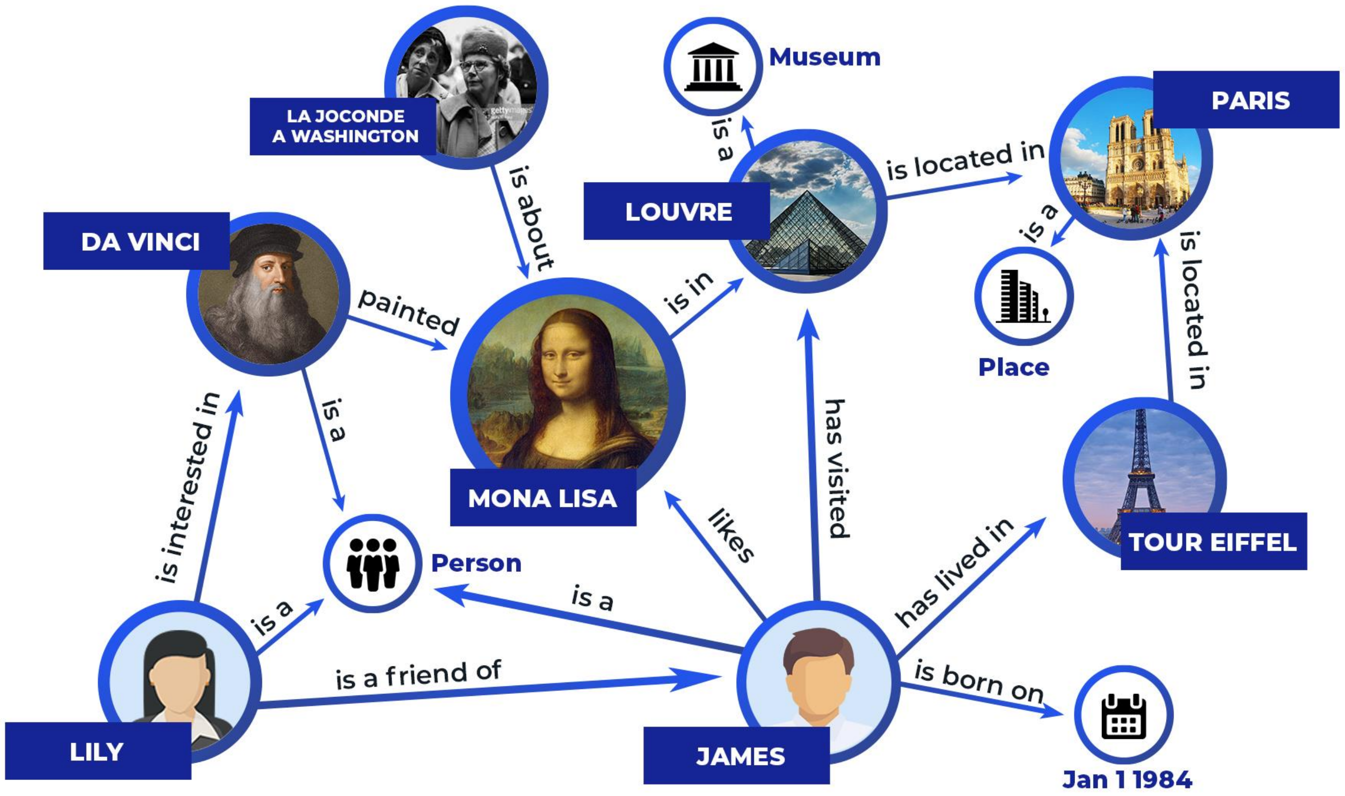
Zilong Chen

Xi’an Jiaotong University

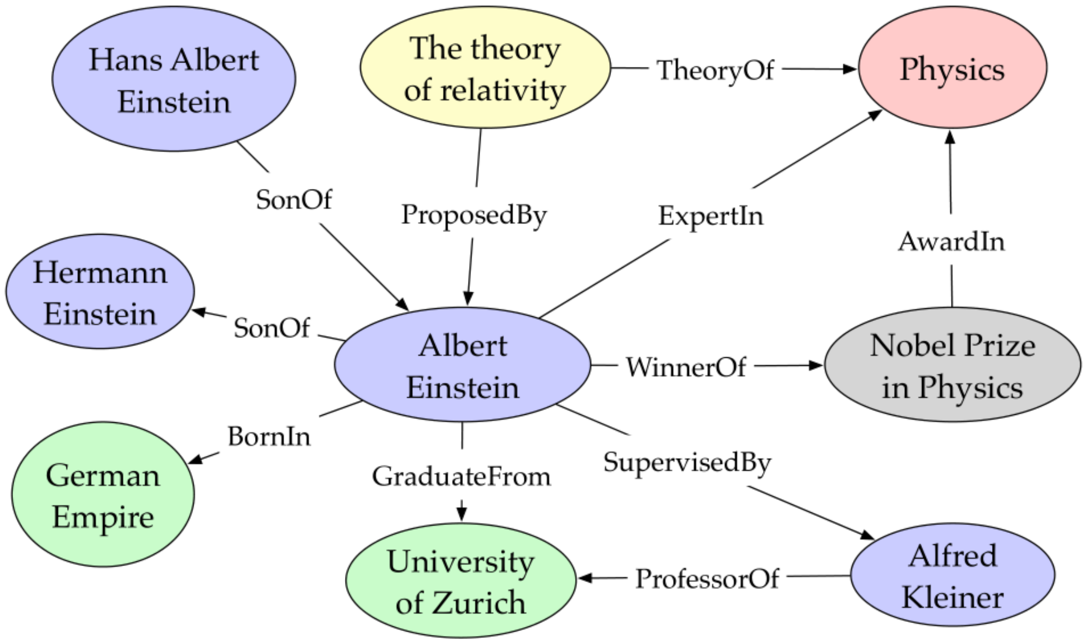
luoyangczl@stu.xjtu.edu.cn

Table of Contents


data:
information:
knowledge:
insight:
wisdom:
impact:
gapingvoid
Culture Design Group
@gapingvoid


LA JOCONDE
A WASHINGTON
Museum
PARIS
LOUVRE
is a
is located in
is about
DA VINCI
painted
is in
is a
Place
is located in
is a
is interested in
MONA LISA
has visited
likes
TOUR EIFFEL
Person
is a
is a
has lived in
is a friend of
is born on
LILY
JAMES
Jan 1 1984

(Albert Einstein, **BornIn**, German Empire)
(Albert Einstein, **SonOf**, Hermann Einstein)
(Albert Einstein, **GraduateFrom**, University of Zurich)
(Albert Einstein, **WinnerOf**, Nobel Prize in Physics)
(Albert Einstein, **ExpertIn**, Physics)
(Nobel Prize in Physics, **AwardIn**, Physics)
(The theory of relativity, **TheoryOf**, Physics)
(Albert Einstein, **SupervisedBy**, Alfred Kleiner)
(Alfred Kleiner, **ProfessorOf**, University of Zurich)
(The theory of relativity, **ProposedBy**, Albert Einstein)
(Hans Albert Einstein, **SonOf**, Albert Einstein)



Definition 1 (Ehrlinger and Wöß). A knowledge graph acquires and integrates information into an ontology and applies a reasoner to derive new knowledge.

Definition 2 (Wang et al.). A knowledge graph is a multi-relational graph composed of entities and relations which are regarded as nodes and different types of edges, respectively.

Knowledge graph is a kind of semantic network which reveals the relationship between entities. It can formally describe the real world things and their relationships.

Triplet are widely used to represent a knowledge graph:

$$(h, r, t)$$

(head, relation, tail)

where

$$E = \{e_1, e_2, \cdots, e_{|E|}\} \Longrightarrow \text{set of entities}$$

$$R = \{r_1, r_2, \cdots, r_{|R|}\} \Longrightarrow \text{set of relations}$$

$$S \subset E \times R \times E$$

Table of Contents

- Introduction
- Tasks and Datasets
- Embedding models
- Applications
- Exploration

Tasks

- knowledge graph completion
- entity classification
- link prediction

Benchmarks

Edit

Trend	Task	Dataset Variant	Best Model	Paper	Code
	Link Prediction	FB15k-237	GAATs		
	Link Prediction	FB15k	AutoKGE		
	Link Prediction	FB15k	MEI		
	Knowledge Graph Completion	FB15k-237	KBGAT		
	Knowledge Graphs	FB15k	HHolE		
	Link Prediction	FB15k (filtered)	ParTransH		
	Knowledge Graph Embedding	FB15k	AcrE		

Collapse benchmarks

Datasets

FB15k & FB15k-237

The **FB15k** dataset contains knowledge base relation triples and textual mentions of Freebase entity pairs. It has a total of 592,213 triplets with 14,951 entities and 1,345 relationships. FB15K-237 is a variant of the original dataset where inverse relations are removed, since it was found that a large number of test triplets could be obtained by inverting triplets in the training set.

DATA SET	WN	FB15K	FB1M
ENTITIES	40,943	14,951	1×10^6
RELATIONSHIPS	18	1,345	23,382
TRAIN. EX.	141,442	483,142	17.5×10^6
VALID EX.	5,000	50,000	50,000
TEST EX.	5,000	59,071	177,404

	Head	**Predicate**	**Tail**
Test	David	location	? (Ans: Florida)
Train	David David	placeOfBirth nationality	Atlanta U.S.A.
Test	Zurich	travelMonth	? (Ans: October)
Train	Zurich	travelMonth	Jan., Feb., Mar., Apr., May., Jun., Jul., Aug., Sep., Nov., Dec.

WN18RR

WN18RR is a link prediction dataset created from WN18, which is a subset of WordNet. WN18 consists of 18 relations and 40,943 entities. However, many text triples are obtained by inverting triples from the training set. Thus the WN18RR dataset is created to ensure that the evaluation dataset does not have inverse relation test leakage. In summary, WN18RR dataset contains 93,003 triples with 40,943 entities and 11 relation types.

Table 1: Statistics of the Four Datasets.

Dataset	$\|\mathcal{E}\|$	$\|\mathcal{R}\|$	#triples in Train/Valid/Test
FB15k	14,951	1,345	483,142 / 50,000 / 59,071
WN18	40,943	18	141,442 / 5,000 / 5,000
FB15k-237	14,541	237	272,115 / 17,535 / 20,466
WN18RR	40,943	11	86,835 / 3,034 / 3,134

More:

- YAGO3-10
- NELL995
- Kinship
- Country

Latest dataset

InferWiki64k and InferWiki16k(ACL 2021)

Are Missing Links Predictable? An Inferential Benchmark for Knowledge Graph Completion

Yixin Cao[1,2] **Xiang Ji**[1] **Xin Lv**[3]
Juanzi Li[3] **Yonggang Wen**[1] **Hanwang Zhang**[1]
[1]School of CSE, Nanyang Technological University, Singapore
[2]S-Lab, Nanyang Technological University, Singapore
[3]Department of CST, Tsinghua University, Beijing, China
caoyixin2011@gmail.com, lv-x18@mails.tsinghua.edu.cn
lijuanzi@tsinghua.edu.cn, {ygwen,hanwangzhang}@ntu.edu.sg

http://arxiv.org/abs/2108.01387

Table of Contents

- rule based

  …

- embedding based

key idea:

- model entities and relations in embedding space(R^d or …)
- find a function to measure the confidence of a given triple.

embed (h, r, t) ➔ embedding space

measure confidence ➔ scoring function

TransE

Translation based model

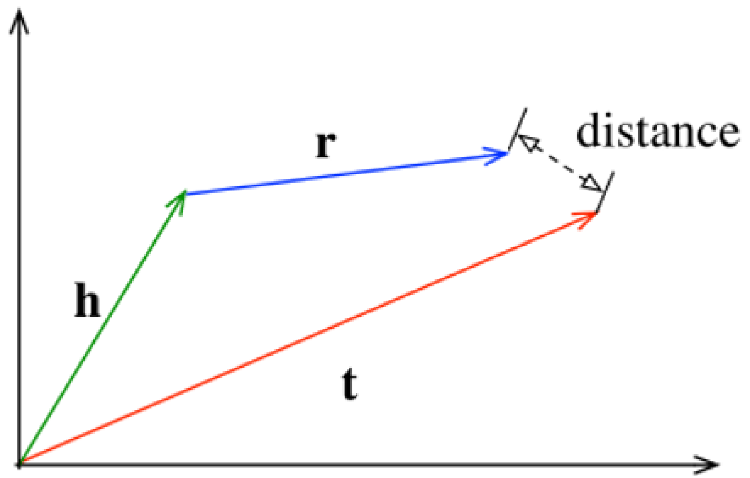
Idea: (distance based)

in vector(embedding) space:

$$\mathbf{h} + \mathbf{r} \approx \mathbf{t}$$

embedding space: $\mathbb{R}^d$

scoring function: $||\mathbf{h} + \mathbf{r} - \mathbf{t}||_{1/2}$



(a) Translational distance-based scoring of TransE.

details

Algorithm 1 Learning TransE

input Training set $S = \{(h, \ell, t)\}$, entities and rel. sets E and L, margin γ, embeddings dim. k.

initialize $\boldsymbol{\ell} \leftarrow \text{uniform}(-\frac{6}{\sqrt{k}}, \frac{6}{\sqrt{k}})$ for each $\ell \in L$

$\quad\quad \boldsymbol{\ell} \leftarrow \boldsymbol{\ell} / \|\boldsymbol{\ell}\|$ for each $\ell \in L$

$\quad\quad \mathbf{e} \leftarrow \text{uniform}(-\frac{6}{\sqrt{k}}, \frac{6}{\sqrt{k}})$ for each entity $e \in E$

loop

$\quad \mathbf{e} \leftarrow \mathbf{e} / \|\mathbf{e}\|$ for each entity $e \in E$

$\quad S_{batch} \leftarrow \text{sample}(S, b)$ // sample a minibatch of size b

$\quad T_{batch} \leftarrow \emptyset$ // initialize the set of pairs of triplets

$\quad$ **for** $(h, \ell, t) \in S_{batch}$ **do**

$\quad\quad (h', \ell, t') \leftarrow \text{sample}(S'_{(h,\ell,t)})$ // sample a corrupted triplet

$\quad\quad T_{batch} \leftarrow T_{batch} \cup \{((h, \ell, t), (h', \ell, t'))\}$

$\quad$ **end for**

$\quad$ Update embeddings w.r.t. $\displaystyle\sum_{((h,\ell,t),(h',\ell,t')) \in T_{batch}} \nabla \left[\gamma + d(\boldsymbol{h} + \boldsymbol{\ell}, \boldsymbol{t}) - d(\boldsymbol{h'} + \boldsymbol{\ell}, \boldsymbol{t'})\right]_+$

end loop

To learn such embeddings, we minimize a margin-based ranking criterion over the training set:

$$\mathcal{L} = \sum_{(h,\ell,t) \in S} \sum_{(h',\ell,t') \in S'_{(h,\ell,t)}} \left[\gamma + d(\boldsymbol{h} + \boldsymbol{\ell}, \boldsymbol{t}) - d(\boldsymbol{h'} + \boldsymbol{\ell}, \boldsymbol{t'})\right]_+ \quad (1)$$

where $[x]_+$ denotes the positive part of x, $\gamma > 0$ is a margin hyperparameter, and

$$S'_{(h,\ell,t)} = \{(h', \ell, t) | h' \in E\} \cup \{(h, \ell, t') | t' \in E\}. \quad (2)$$

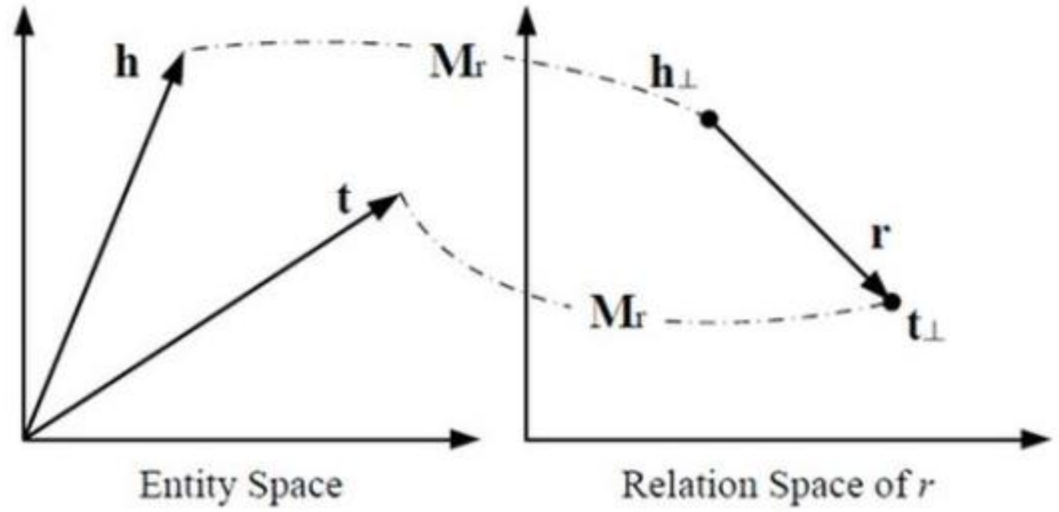
TransR

In relation-specific space

$$h_{\perp} + r \approx t_{\perp}$$

where

$$h_{\perp} = M_r h \quad t_{\perp} = M_r t$$



(project entities into relation-specific spaces)

embedding space: same as TransE

scoring function: $\|h_{\perp} + r - t_{\perp}\|_{1/2}$

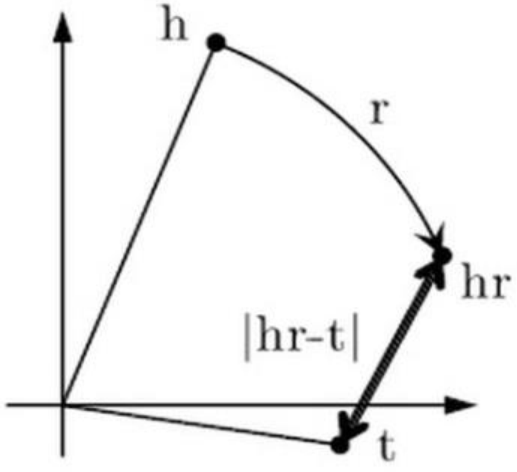
More TransX: TransH, TransG, TransD, TransM, …

RotatE

idea: embed entities and relations into complex space

motivation: express different types of relations

Model	Score Function	Symmetry	Antisymmetry	Inversion	Composition
SE	$-\|\boldsymbol{W}_{r,1}\mathbf{h} - \boldsymbol{W}_{r,2}\mathbf{t}\|$	✗	✗	✗	✗
TransE	$-\|\mathbf{h} + \mathbf{r} - \mathbf{t}\|$	✗	✓	✓	✓
TransX	$-\|g_{r,1}(\mathbf{h}) + \mathbf{r} - g_{r,2}(\mathbf{t})\|$	✓	✓	✗	✗
DistMult	$\langle \mathbf{h}, \mathbf{r}, \mathbf{t} \rangle$	✓	✗	✗	✗
ComplEx	$\mathrm{Re}(\langle \mathbf{h}, \mathbf{r}, \overline{\mathbf{t}} \rangle)$	✓	✓	✓	✗
RotatE	$-\|\mathbf{h} \circ \mathbf{r} - \mathbf{t}\|$	✓	✓	✓	✓



in each dimension

expected:

$$t_i = h_i r_i, \text{ where } h_i, r_i, t_i \in \mathbb{C} \text{ and } |r_i| = 1.$$

embedding space: $\mathbb{C}^d$

scoring function: $d_r(\mathbf{h}, \mathbf{t}) = \|\mathbf{h} \circ \mathbf{r} - \mathbf{t}\|$

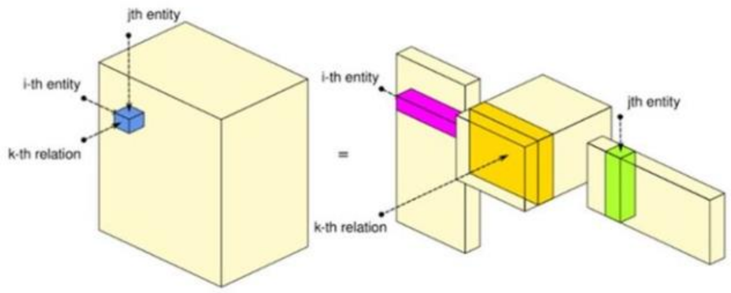
loss function:

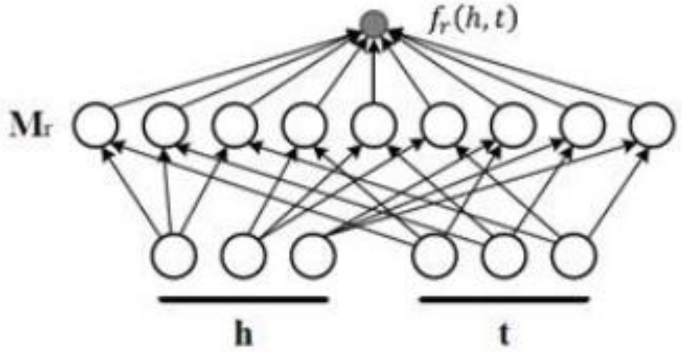
$$L = -\log \sigma(\gamma - d_r(\mathbf{h}, \mathbf{t})) - \sum_{i=1}^{n} \frac{1}{k} \log \sigma(d_r(\mathbf{h}'_i, \mathbf{t}'_i) - \gamma),$$

semantic matching models(or tensor decomposition)

Rescal

$$f_r(h,t) = \mathbf{h}^\top \mathbf{M}_r \mathbf{t} = \sum_{i=0}^{d-1} \sum_{j=0}^{d-1} [\mathbf{M}_r]_{ij} \cdot [\mathbf{h}]_i \cdot [\mathbf{t}]_j$$





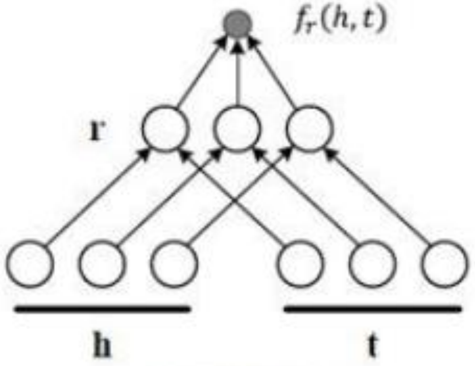
(a) RESCAL.

too many params, overfitting!

DisMult

rescal with diagonal M_r

$$f_r(h,t) = \mathbf{h}^\top \mathrm{diag}(\mathbf{r}) \mathbf{t} = \sum_{i=0}^{d-1} [\mathbf{r}]_i \cdot [\mathbf{h}]_i \cdot [\mathbf{t}]_i.$$



(b) DistMult.

only symmetric relations can be modeled

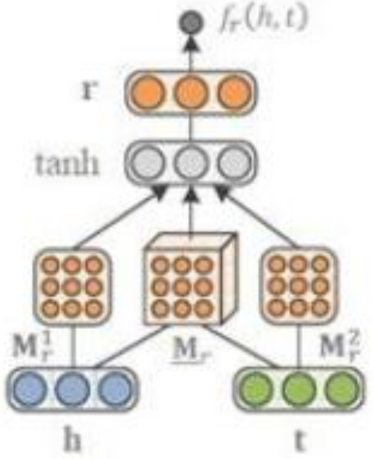
ComplE

$$\phi(h,r,t) = \mathrm{Re}(\langle e_h, e_r, \bar{e_t} \rangle)$$
$$= \mathrm{Re}(\sum_{k=1}^{d} e_h^{(k)} e_r^{(k)} \bar{e_t}^{(k)})$$

NTN

$$f_r(h,t) = \mathbf{r}^\top \tanh\left(\mathbf{h}^\top \underline{\mathbf{M}}_r \mathbf{t} + \mathbf{M}_r^1 \mathbf{h} + \mathbf{M}_r^2 \mathbf{t} + \mathbf{b}_r\right)$$

introduce non-linearity

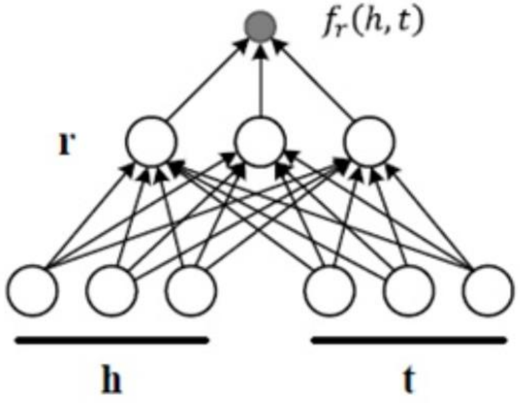


(b) NTN.

HolE

$$[\mathbf{h} \star \mathbf{t}]_i = \sum_{k=0}^{d-1} [\mathbf{h}]_k \cdot [\mathbf{t}]_{(k+i) \bmod d}$$

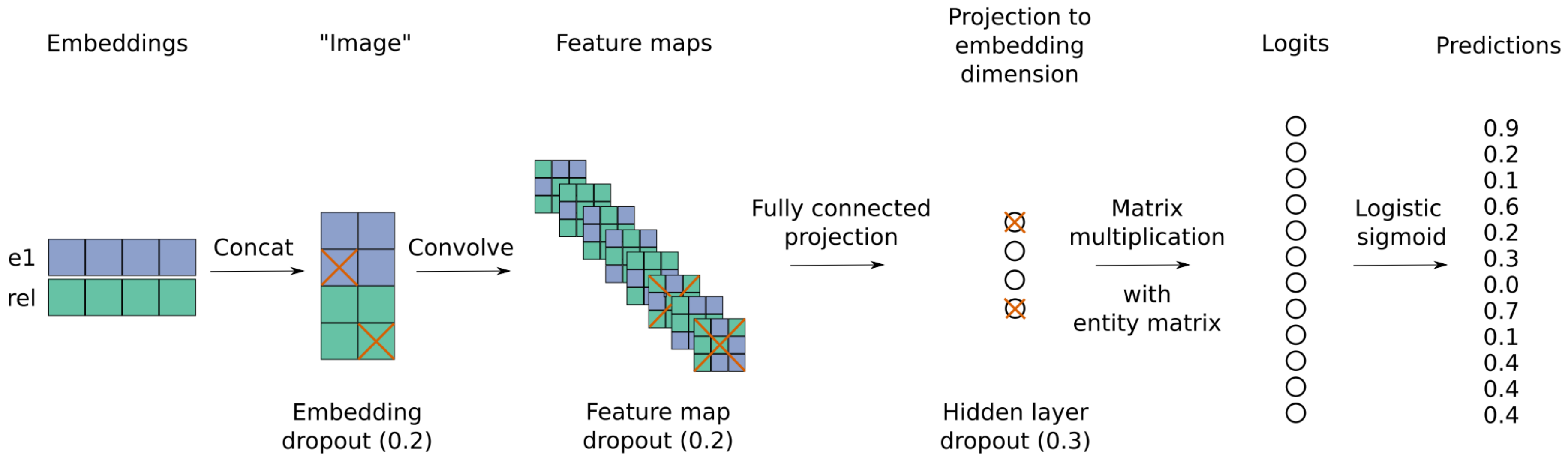
$$f_r(h,t) = \mathbf{r}^\top (\mathbf{h} \star \mathbf{t}) = \sum_{i=0}^{d-1} [\mathbf{r}]_i \sum_{k=0}^{d-1} [\mathbf{h}]_k \cdot [\mathbf{t}]_{(k+i)} \bmod d$$



(c) HolE.

ConvE

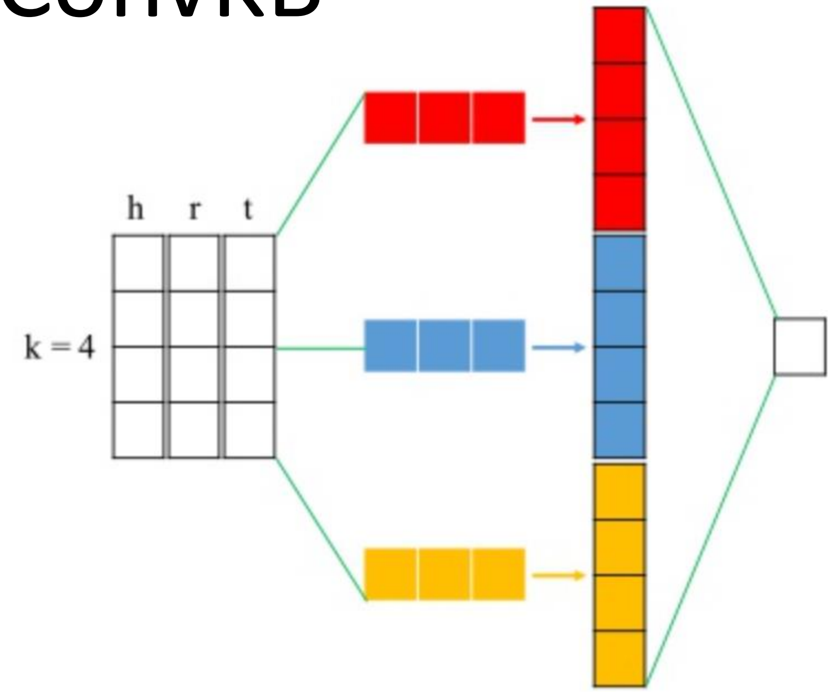
introduce conv into kgc



scoring function: $\psi_r(\mathbf{e}_s, \mathbf{e}_o) = f(\mathrm{vec}(f([\overline{\mathbf{e}_s}; \overline{\mathbf{r}_r}] * \omega))\mathbf{W})\mathbf{e}_o,$

loss: $\mathcal{L}(p, t) = -\frac{1}{N}\sum_i (t_i \cdot \log(p_i) + (1 - t_i) \cdot \log(1 - p_i)),$

ConvKB



ConvE: tail entity has not been operated

ConvKB: take t into consideration

motivation:

model can degenerate to TransE with w = [1,1,-1].

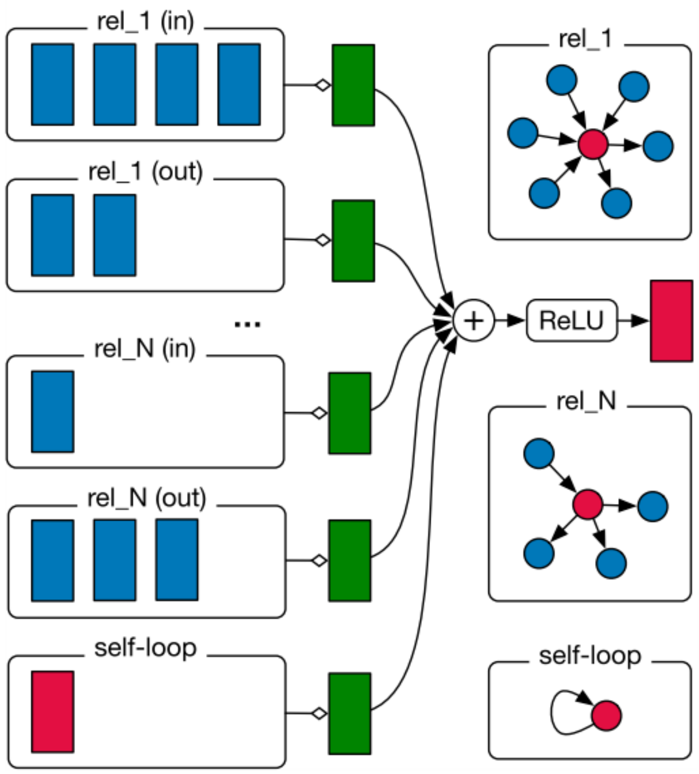
scoring function: $f(h,r,t) = \text{concat}\left(g\left([\boldsymbol{v}_h, \boldsymbol{v}_r, \boldsymbol{v}_t] * \boldsymbol{\Omega}\right)\right) \cdot \mathbf{w}$

loss:

$$\mathcal{L} = \sum_{(h,r,t) \in \{\mathcal{G} \cup \mathcal{G}'\}} \log\left(1 + \exp\left(l_{(h,r,t)} \cdot f(h,r,t)\right)\right) + \frac{\lambda}{2}\|\mathbf{w}\|_2^2$$

$$\text{in which, } l_{(h,r,t)} = \begin{cases} 1 \text{ for } (h,r,t) \in \mathcal{G} \\ -1 \text{ for } (h,r,t) \in \mathcal{G}' \end{cases}$$

R-GCN structure information



encoder: R-GCN

$$h_i^{(l+1)} = \sigma\left(\sum_{r\in\mathcal{R}}\sum_{j\in\mathcal{N}_i^r}\frac{1}{c_{i,r}}W_r^{(l)}h_j^{(l)} + W_0^{(l)}h_i^{(l)}\right)$$

decoder: DisMult

$$f(s,r,o) = e_s^T R_r e_o \,.$$

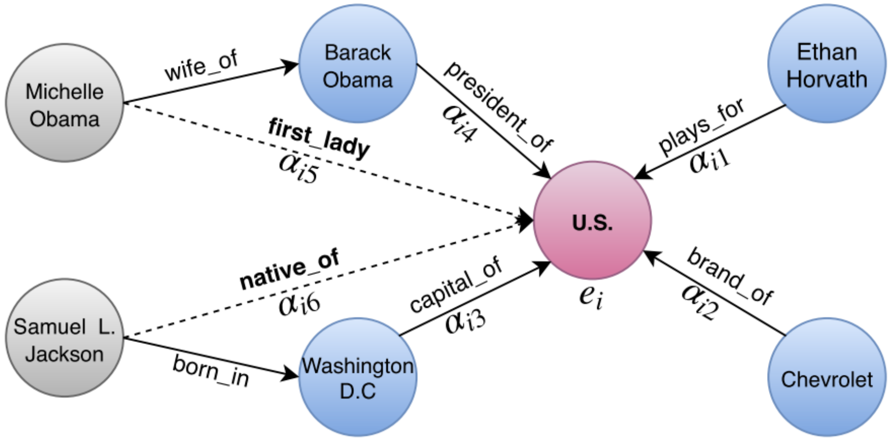
with regularization

$$W_r^{(l)} = \sum_{b=1}^{B} a_{rb}^{(l)} V_b^{(l)}, \qquad W_r^{(l)} = \bigoplus_{b=1}^{B} Q_{br}^{(l)}.$$

loss function:

$$\mathcal{L} = -\frac{1}{(1+\omega)|\hat{\mathcal{E}}|}\sum_{(s,r,o,y)\in\mathcal{T}} y\log l\big(f(s,r,o)\big) + (1-y)\log\big(1-l\big(f(s,r,o)\big)\big)\,,$$

KBGAT



Basic encoder: GAT

$$\vec{x}_i' = \sigma\left(\sum_{j\in\mathcal{N}_i} \alpha_{ij}\mathbf{W}\vec{x}_j\right) \qquad e_{ij} = a(\mathbf{W}\vec{x}_i, \mathbf{W}\vec{x}_j)$$

The final layer

$$\vec{x}_i' = \Big\Vert_{k=1}^{K} \sigma\left(\sum_{j\in\mathcal{N}_i} \alpha_{ij}^k\mathbf{W}^k\vec{x}_j\right) \qquad \vec{x}_i' = \sigma\left(\frac{1}{K}\sum_{k=1}^{K}\sum_{j\in\mathcal{N}_i} \alpha_{ij}^k\mathbf{W}^k\vec{x}_j\right)$$

Aggregate by sum instead of concat

relations not involved in original GAT

GAT with triple emebedding

update entity embedding:

$$\vec{c_{ijk}} = \mathbf{W}_1[\vec{h}_i \Vert \vec{h}_j \Vert \vec{g}_k] \qquad b_{ijk} = \mathrm{LeakyReLU}\Big(\mathbf{W}_2 c_{ijk}\Big)$$

$$\alpha_{ijk} = \mathrm{softmax}_{jk}(b_{ijk}) \qquad \vec{h}_i' = \Big\Vert_{m=1}^{M} \sigma\left(\sum_{j\in\mathcal{N}_i} \alpha_{ijk}^m c_{ijk}^m\right)$$

retaining initial feature

$$\mathbf{H}'' = \mathbf{W}^E\mathbf{H}^t + \mathbf{H}^f$$

initial embedding

embedding from final layer

update relation embedding

$$G' = G.\mathbf{W}^R$$

exploiting TransE embedding as initial feature

loss

For encoder

$$L(\Omega) = \sum_{t_{ij} \in S} \sum_{t'_{ij} \in S'} \max\{d_{t'_{ij}} - d_{t_{ij}} + \gamma, 0\}$$

$$d_{t_{ij}} = \|\vec{h_i} + \vec{g_k} - \vec{h_j}\|_1$$

$$S' = \underbrace{\{t^k_{i'j} \mid e'_i \in \mathcal{E} \setminus e_i\}}_{\text{replace head entity}} \cup \underbrace{\{t^k_{ij'} \mid e'_j \in \mathcal{E} \setminus e_j\}}_{\text{replace tail entity}}$$

For decoder: ConvKB

$$f(t^k_{ij}) = \left(\Big\Vert_{m=1}^{\Omega} \mathrm{ReLU}([\vec{h}_i, \vec{g}_k, \vec{h}_j] * \omega^m) \right).\mathbf{W}$$

kgc model can be trained
without end-to-end manner

Metrics

MR: mean rank

For a given triple (h,r,t) in test set, consider the confidence of (h,r,?) where ? is an arbitrary entity in dataset.

MR = mean(rank of (h,r,t) in {(h,r,?)})

MRR: mean rank reciprocal

MRR = mean(1/(rank of (h,r,t) in {(h,r,?)}))

Hit@k

Hit@k = #(rank(h,r,t)<=k)/#(num of triples)

Evaluation protocol

no filter: same as raw MR and MRR

filter: remove (h,r,t) in training dataset from {(h,r,?)}

top: the given (h,r,t) ranks on the top of all triples with the same confidence score.

bottom:the given (h,r,t) ranks on the bottom of all triples with the same confidence score.

random: ranks randomly.

Performance

		FB15k				WN18				FB15k-237				WN18RR				YAGO3-10			
		H@1	H@10	MR	MRR	H@1	H@10	MR	MRR	H@1	H@10	MR	MRR	H@1	H@10	MR	MRR	H@1	H@10	MR	MRR
Tensor Decomposition Models	DistMult	73.61	86.32	173	0.784	72.60	94.61	675	0.824	22.44	49.01	199	0.313	39.68	50.22	5913	0.433	41.26	66.12	1107	0.501
	ComplEx	**81.56**	**90.53**	**34**	**0.848**	94.53	95.50	3623	0.949	25.72	52.97	202	0.349	42.55	52.12	4907	0.458	**50.48**	**70.35**	1112	**0.576**
	ANALOGY	65.59	83.74	126	0.726	92.61	94.42	808	0.934	12.59	35.38	476	0.202	35.82	38.00	9266	0.366	19.21	45.65	2423	0.283
	SimplE	66.13	83.63	138	0.726	93.25	94.58	759	0.938	10.03	34.35	651	0.179	38.27	42.65	8764	0.398	35.76	63.16	2849	0.453
	HolE	75.85	86.78	211	0.800	93.11	94.94	650	0.938	21.37	47.64	186	0.303	40.28	48.79	8401	0.432	41.84	65.19	6489	0.502
	TuckER	72.89	88.88	39	0.788	**94.64**	95.80	510	**0.951**	**25.90**	**53.61**	**162**	**0.352**	42.95	51.40	6239	0.459	46.56	68.09	2417	0.544
																3936	0.206	40.57	67.39	1187	0.501
																5172	0.226	3.28	7.35	5797	0.049
																5212	0.405	33.09	65.45	3839	0.446
																4873	0.463	27.43	47.44	19455	0.342
																3318	0.475	40.52	67.07	1827	0.498
																4944	0.427	39.93	65.75	2429	0.488
																3429	0.249	32.16	60.47	1683	0.420
																5646	0.467	44.62	67.33	2582	0.527
Deep L	CapsE	1.93	21.78	610	0.087	84.55	95.08	233	0.890	7.34	35.60	405	0.160	33.69	55.98	**720**	0.415	0.00	0.00	60676	0.000
	RSN	72.34	87.01	51	0.777	91.23	95.10	346	0.928	19.84	44.44	248	0.280	34.59	48.34	4210	0.395	42.65	66.43	1339	0.511
	AnyBURL	81.09	87.86	288	0.835	94.63	95.96	233	**0.951**	24.03	48.93	480	0.324	**44.93**	55.97	2530	**0.485**	45.83	66.07	**815**	0.528

Table 3. Global H@1, H@10, MR and MRR results for all LP models on each dataset. The best results of each metric for each dataset are marked in bold and underlined.

	Reported			RANDOM			TOP			BOTTOM		
	MRR ↑	MR ↓	H@10 ↑	MRR ↑	MR ↓	H@10 ↑	MRR ↑	MR ↓	H@10 ↑	MRR ↑	MR ↓	H@10 ↑
ConvE	.325	244	.501	.324 ± .0	285 ± 0	.501 ± .0	.324	285	.501	.324	285	.501
RotatE	.338	177	.533	.336 ± .0	178 ± 0	.530 ± .0	.336	178	.530	.336	178	.530
TuckER	.358	-	.544	.353 ± .0	162 ± 0	.536 ± .0	.353	162	.536	.353	162	.536
ConvKB	.396	257	.517	.243 ± .0	309 ± 2	.421 ± .0	.407 (+.164)	246 (-63)	.527 (+.106)	.130 (-.113)	373 (+64)	.383 (-.038)
CapsE	.523	303	.593	.150 ± .0	403 ± 2	.356 ± .0	.511 (+.361)	305 (-99)	.586 (+.229)	.134 (-.016)	502 (+99)	.297 (-.059)
KBAT	.518†	210†	.626†	.157 ± .0	270 ± 0	.331 ± .0	.157	270	.331	.157	270	.331

summary

Family and Group		Model	Loss	Constraints	Space Complexity
Tensor Decomposition Models	Bilinear	DistMult	$\boldsymbol{h} \times \boldsymbol{r} \times \boldsymbol{t}$	$\forall r \in \mathcal{R}$: $\boldsymbol{r}$ is diagonal;	$\mathcal{O}(\|\mathcal{E}\|d + \|\mathcal{R}\|d)$
		ComplEx	$\boldsymbol{h} \times \boldsymbol{r} \times \bar{\boldsymbol{t}}$	$\boldsymbol{h} \in \mathbb{C}^d$; $\boldsymbol{t} \in \mathbb{C}^d$; $\boldsymbol{r} \in \mathbb{C}^{d\times d}$; $\forall r \in \mathcal{R}$: $\boldsymbol{r}$ is diagonal;	$\mathcal{O}(\|\mathcal{E}\|d + \|\mathcal{R}\|d)$
		Analogy	$\boldsymbol{h} \times \boldsymbol{r} \times \boldsymbol{t}$	$\forall r \in \mathcal{R}$: $\boldsymbol{r}\times\boldsymbol{r}^T = \boldsymbol{r}^T\times\boldsymbol{r}$; $\forall(r_1, r_2) \in \mathcal{R}\times\mathcal{R}$: $\boldsymbol{r_1}\times\boldsymbol{r_2} = \boldsymbol{r_2}\times\boldsymbol{r_1}$;	$\mathcal{O}(\|\mathcal{E}\|d + \|\mathcal{R}\|d)$
		SimplE	$½(\boldsymbol{h_h} \times r \times \boldsymbol{t_t}) + ½(\boldsymbol{h_t} \times \boldsymbol{r_{-1}} \times \boldsymbol{t_h})$	$\forall r \in \mathcal{R}$: $\boldsymbol{r}, \boldsymbol{r_{-1}}$ are diagonal;	$\mathcal{O}(2\|\mathcal{E}\|d + 2\|\mathcal{R}\|d)$
	Non-bilinear	HolE	$(\boldsymbol{h} \star \boldsymbol{t}) \times \boldsymbol{r}$		$\mathcal{O}(\|\mathcal{E}\|d + \|\mathcal{R}\|d)$
		TuckER	$\boldsymbol{W} \times_1 \boldsymbol{h} \times_2 \boldsymbol{r} \times_3 \boldsymbol{t}$		$\mathcal{O}(\|\mathcal{E}\|d_e + \|\mathcal{R}\|d_r + d_e d_r d_e)$
Geometric Models	Pure Translation	TransE	$\parallel \boldsymbol{h} + \boldsymbol{r} - \boldsymbol{t} \parallel$		$\mathcal{O}(\|\mathcal{E}\|d + \|\mathcal{R}\|d)$
	Additional Embeddings	STransE	$\parallel \boldsymbol{W_r^h} \times \boldsymbol{h} + \boldsymbol{r} - \boldsymbol{W_r^t} \times \boldsymbol{t} \parallel$		$\mathcal{O}(\|\mathcal{E}\|d + \|\mathcal{R}\|(d + 2d^2))$
		CrossE	$\sigma(tanh(\boldsymbol{h} \odot \boldsymbol{c_r} + \boldsymbol{r} \odot \boldsymbol{h} \odot \boldsymbol{c_r}) \times \boldsymbol{t}^T)$		$\mathcal{O}(\|\mathcal{E}\|d + 2\|\mathcal{R}\|d)$
	Roto-translation	TorusE	$min_{(\boldsymbol{x}, \boldsymbol{y}) \in ([\boldsymbol{h}]+[\boldsymbol{r}])\times[\boldsymbol{t}]} \|\boldsymbol{x} - \boldsymbol{y}\|_i$		$\mathcal{O}(\|\mathcal{E}\|d + \|\mathcal{R}\|d)$
		RotatE	$-\parallel \boldsymbol{h} \odot \boldsymbol{r} - \boldsymbol{t} \parallel$	$\boldsymbol{h} \in \mathbb{C}^d$; $\boldsymbol{r} \in \mathbb{C}^d$; $\boldsymbol{t} \in \mathbb{C}^d$; $\forall \boldsymbol{r_i} \in \boldsymbol{r}$: $\|\boldsymbol{r_i}\| = 1$;	$\mathcal{O}(\|\mathcal{E}\|d + \|\mathcal{R}\|d)$
Deep Learning Models	Convolution	ConvE	$g(\boldsymbol{W} \times g([\boldsymbol{h}; \boldsymbol{r}] \circledast \boldsymbol{\omega}) + \mathrm{b}) \times \boldsymbol{t}$		$\mathcal{O}(\|\mathcal{E}\|d + \|\mathcal{R}\|d + Tmn + Td(2d_m - m + 1)(d_n - n + 1))$
		ConvKB	$g(\boldsymbol{W} \times g([\boldsymbol{h}; \boldsymbol{r}; \boldsymbol{t}] \circledast \boldsymbol{\omega}) + \mathrm{b})$		$\mathcal{O}(\|\mathcal{E}\|d + \|\mathcal{R}\|d + 4T)$
		ConvR	$g(\boldsymbol{W} \times g([\boldsymbol{h}] \circledast \boldsymbol{\omega_r}) + \mathrm{b}) \times \boldsymbol{t}$		$\mathcal{O}\left(\|\mathcal{E}\|d_e + \|\mathcal{R}\|d_r + Tmn + Td_e(2d_{e_m} - m + 1)(d_{e_n} - n + 1)\right)$
	Capsule	CapsE	$\|capsnet(g([\boldsymbol{h}; \boldsymbol{r}; \boldsymbol{t}] \circledast \boldsymbol{\omega}))\|$		$\mathcal{O}(\|\mathcal{E}\|d + \|\mathcal{R}\|d + 3T + Td)$
	Recurrent	RSN	$\sigma(rsn(\boldsymbol{h}, \boldsymbol{r}) \times \boldsymbol{t})$		$\mathcal{O}(2\|\mathcal{E}\|d + 2\|\mathcal{R}\|d + Lknd)$

Table 1. Loss Function, constraints and space complexity for the models included in our analysis.

Table of Contents

Applications

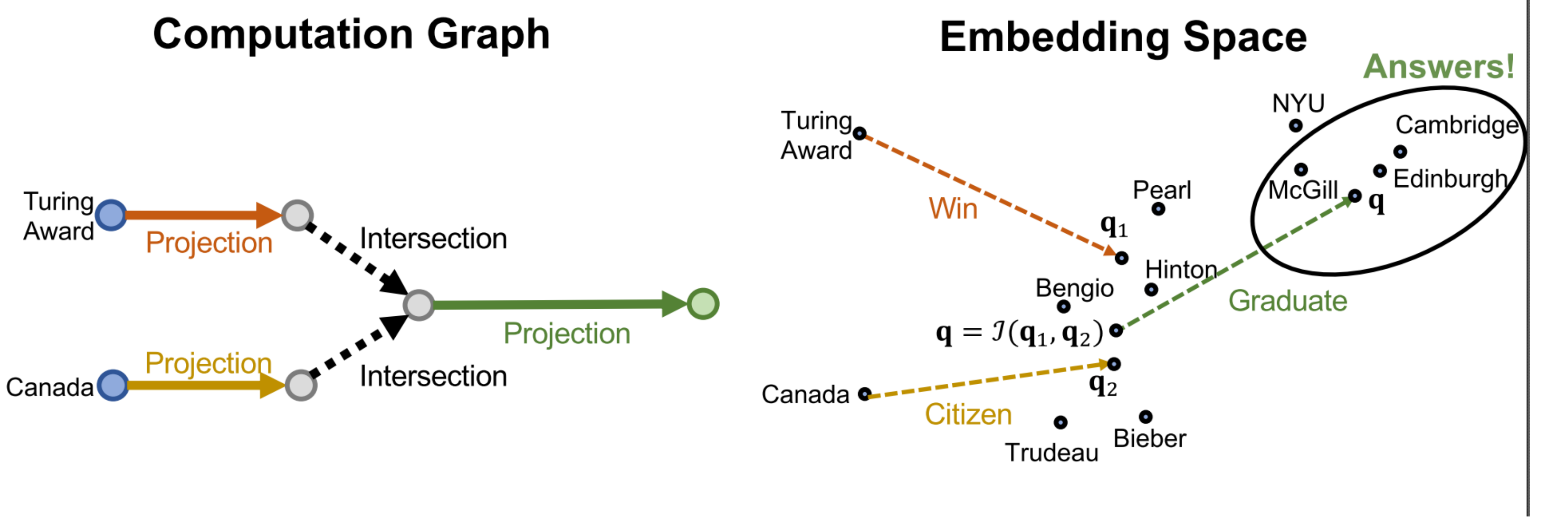
QA with KG

multi-hop QA: Query2box

simplest idea: represent queries in embedding space(traversing KG)

"Where did Canadian citizens with Turing Award graduate?"

Follow the computation graph:



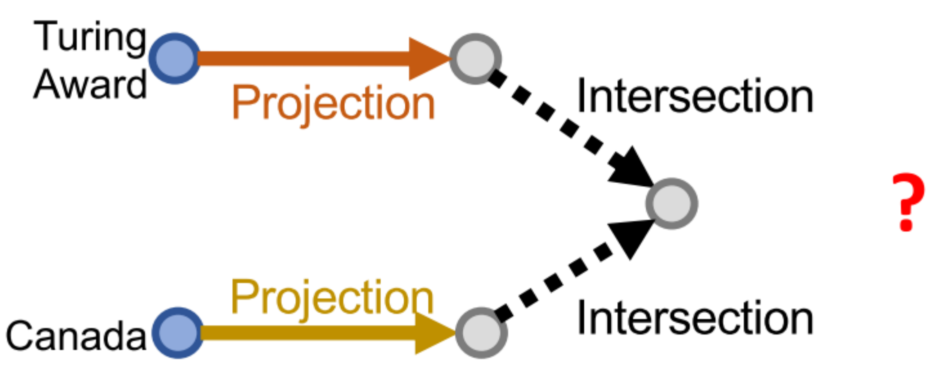
- Embed queries in vector space

"Where did Canadian citizens with Turing Award graduate?"

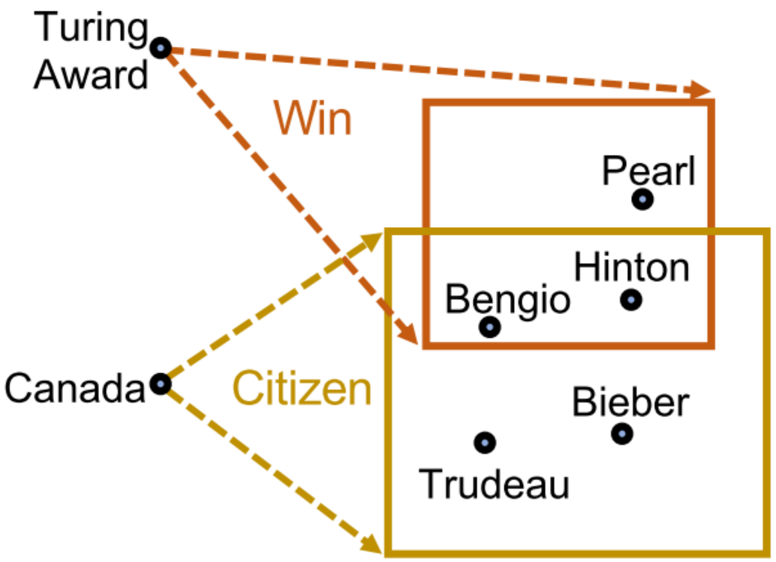
Note that computation graph stays the same!

Follow the computation graph:

Computation Graph

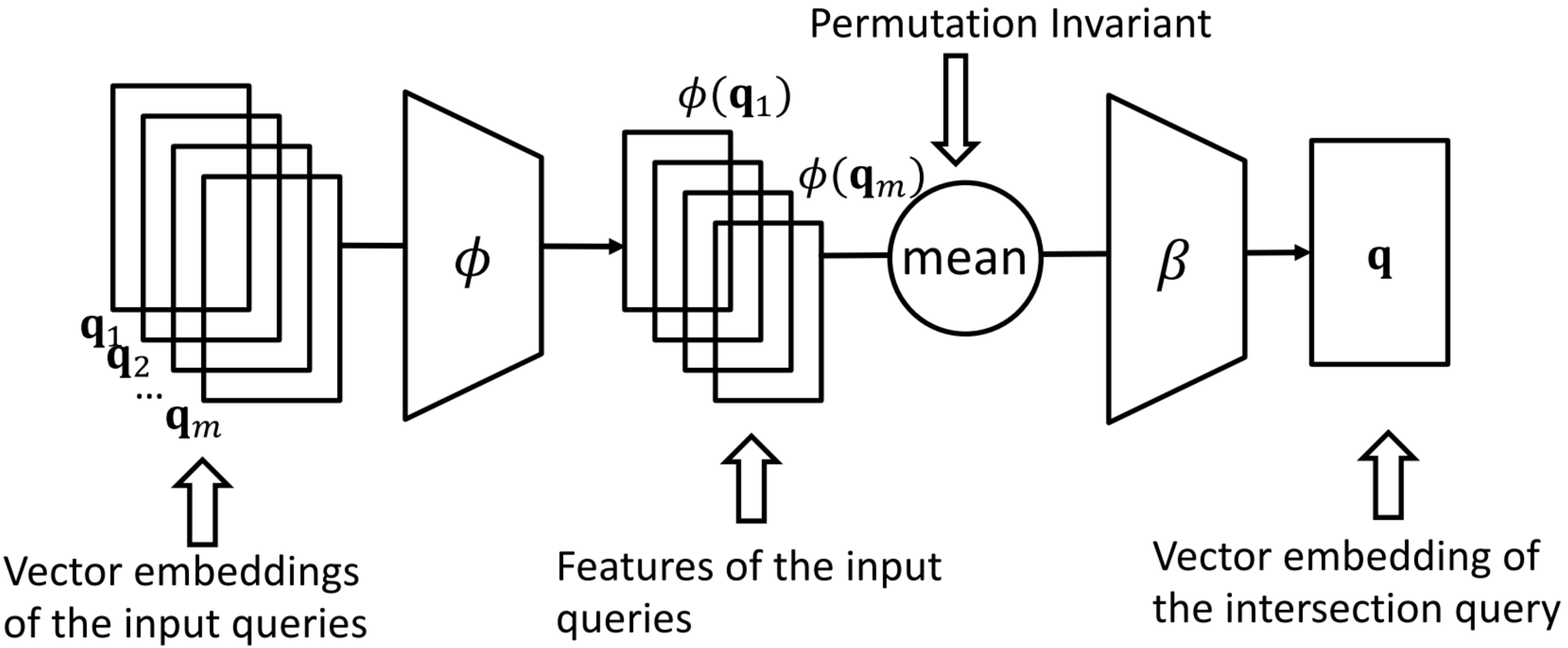


Embedding Space



- Geometric Intersection Operator $\mathcal{I}$

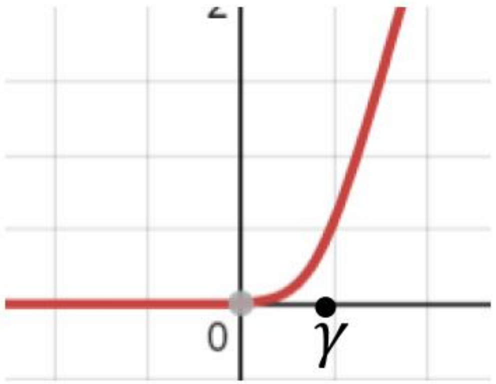
- DeepSets architecture



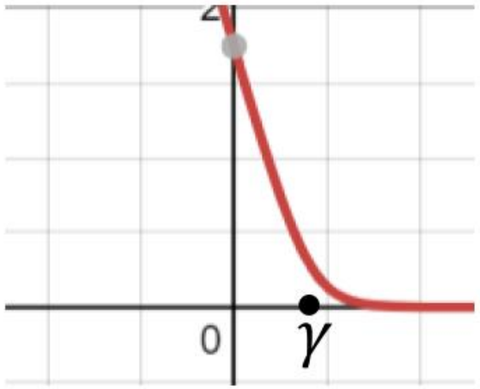
$$\mathrm{Off}(\mathbf{p}_{\text{inter}}) = \mathrm{Min}(\{\mathrm{Off}(\mathbf{p_1}), \ldots, \mathrm{Off}(\mathbf{p_n})\}) \odot \sigma(\mathrm{DeepSets}(\{\mathbf{p_1}, \ldots, \mathbf{p_n}\})),$$

- Given a set of queries and answers,

$$\mathcal{L} = -\log \sigma\big(\gamma - d_{box}(q,v)\big) - \log \sigma(d_{box}(q,v_i') - \gamma)$$



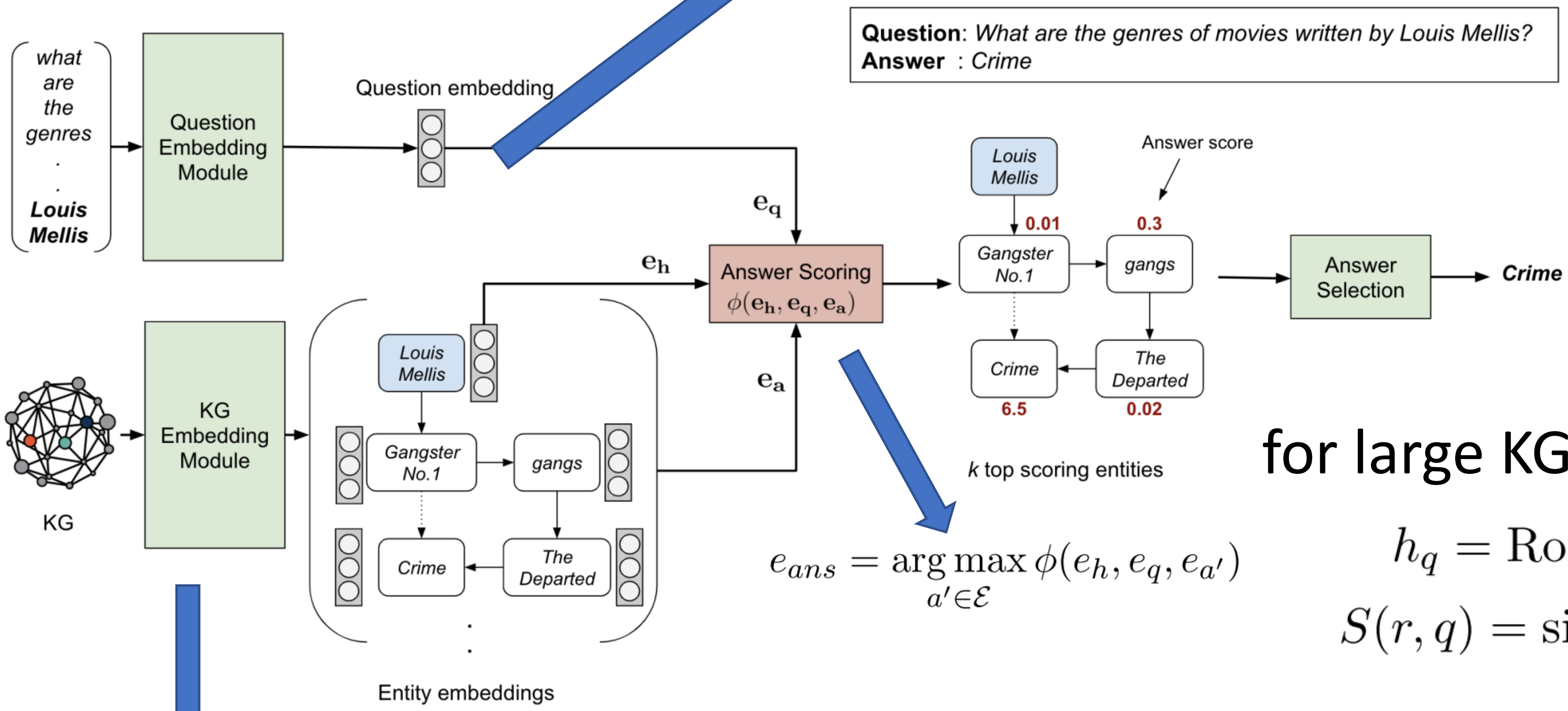
$-\log \sigma\big(\gamma - d_{box}(q,v)\big)$
minimize loss → minimize $d_{box}(q,v)$



$-\log \sigma(d_{box}(q,v') - \gamma)$
minimize loss → maximize $d_{box}(q,v')$

EmbedKGQA

RoBERTa+4×MLP



$$e_{ans} = \arg\max_{a' \in \mathcal{E}} \phi(e_h, e_q, e_{a'})$$

$$\phi(h, r, t) = \mathrm{Re}(\langle e_h, e_r, \bar{e}_t \rangle)$$

$$= \mathrm{Re}(\sum_{k=1}^{d} e_h^{(k)} e_r^{(k)} \bar{e}_t^{(k)})$$

for large KG, pruning:

$$h_q = \mathrm{RoBERTa}(q')$$

$$S(r, q) = \mathrm{sigmoid}(h_q^T h_r)$$

$$\mathrm{RelScore}_{a'} = |\mathcal{R}_a \cap \mathcal{R}_{a'}|$$

$$e_{ans} = \arg\max_{a' \in \mathcal{N}_h} \phi(e_h, e_q, e_{a'}) + \gamma * \mathrm{RelScore}_{a'}$$

More applications

- knowledge graph reasoning
- knowledge graph enhanced LM
- recommendation system
- KG enhanced semantic CV
- Temporal KG with dynamic tasks
- …

Table of Contents

ACL 2021 KG related papers

1. Robust Knowledge Graph Completion with Stacked Convolutions and a Student Re-Ranking Network
2. Using Meta-Knowledge Mined from Identifiers to Improve Intent Recognition in Conversational Systems
3. Topic-Driven and Knowledge-Aware Transformer for Dialogue Emotion Detection
4. **Employing Argumentation Knowledge Graphs for Neural Argument Generation**
5. **PairRE: Knowledge Graph Embeddings via Paired Relation Vectors**
6. How Knowledge Graph and Attention Help? A Qualitative Analysis into Bag-level Relation Extraction
7. **Are Missing Links Predictable? An Inferential Benchmark for Knowledge Graph Completion**

Thx for Attention

Zilong Chen

Xi’an Jiaotong University

luoyangczl@stu.xjtu.edu.cn