TransEdge: Translating Relation-Contextualized Embeddings for Knowledge Graphs

Zequn Sun[1], Jiacheng Huang[1], Wei Hu[1], Muhao Chen[2], Lingbing Guo[1], and Yuzhong Qu[1]

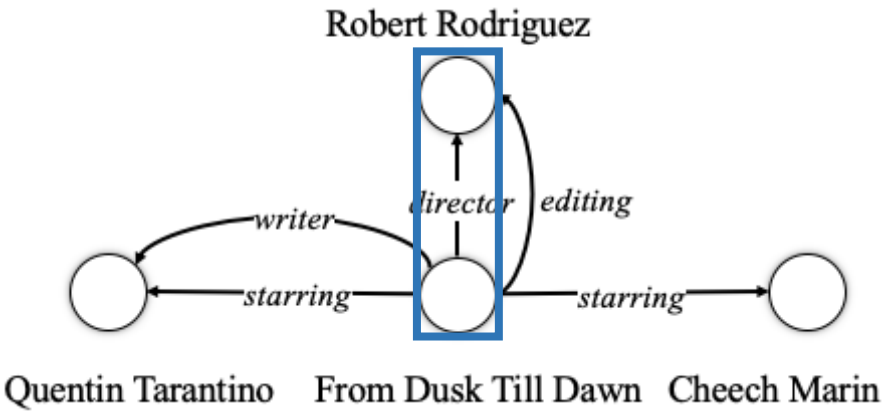
[1]State Key Laboratory for Novel Software Technology
Nanjing University

[2]Department of Computer Science
University of California, Los Angeles

Introduction

Knowledge Graphs

- A knowledge graph (KG) is a multi-relational graph, whose nodes denote entities and directed edges are associated with types indicating the specific relations between entities.
  - Each edge denotes a relational triple which represents a fact.
    $(From\ Dusk\ Till\ Dawn, director, Robert\ Rodrigues)$
- A KG = $(\mathcal{E}, \mathcal{R}, \mathcal{T})$
  - $\mathcal{E}$: the set of entities (nodes)
  - $\mathcal{R}$: the set of relations (edge types)
  - $\mathcal{T} = \mathcal{E} \times \mathcal{R} \times \mathcal{E}$:
    the set of relational triples (edges)

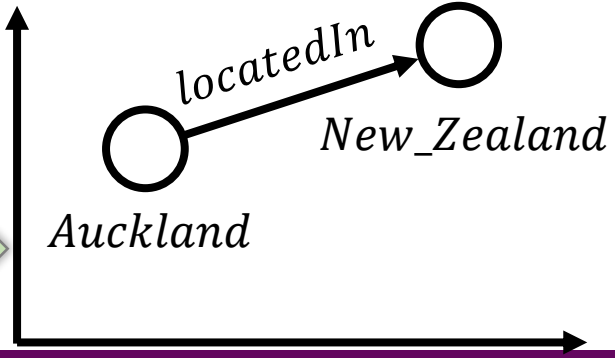


Knowledge Graph Embedding

- KG embedding (KGE) techniques seek to encode entities and relations into vector spaces, and capture semantics by the geometric structure of embeddings.
- Translational KGE models
  - TransE: for a relational triple (h, r, t), the vectors of h and t are connected by a translation vector of r, i.e., $\mathbf{h} + \mathbf{r} \approx \mathbf{t}$ (where the boldfaced letters denote the embeddings of entities and relations). The energy function of TransE is $f(h, r, t) = \|\mathbf{h} + \mathbf{r} - \mathbf{t}\|_{L_1/L_2}$.
- Tasks
  - Link prediction, entity alignment, …

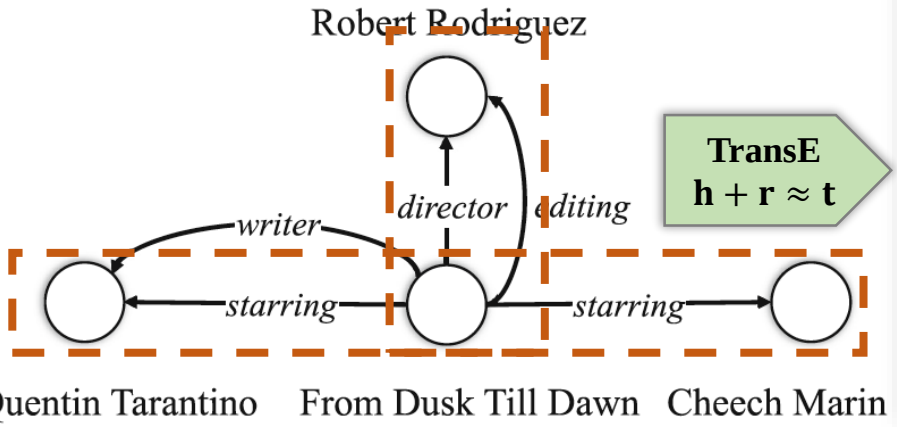
$(Auckland, locatedIn, New\_Zealand)$

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

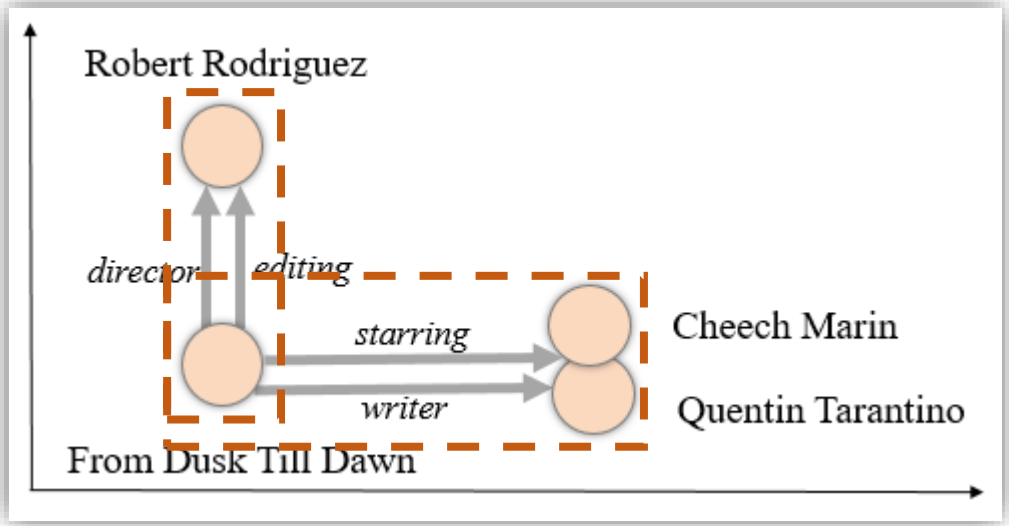


Weakness of Translational KGE

- The relation-level translation cannot model the complex relational structure of KGs.
  - One entity pair with multiple relations
  - Multiple entity pairs with one relation



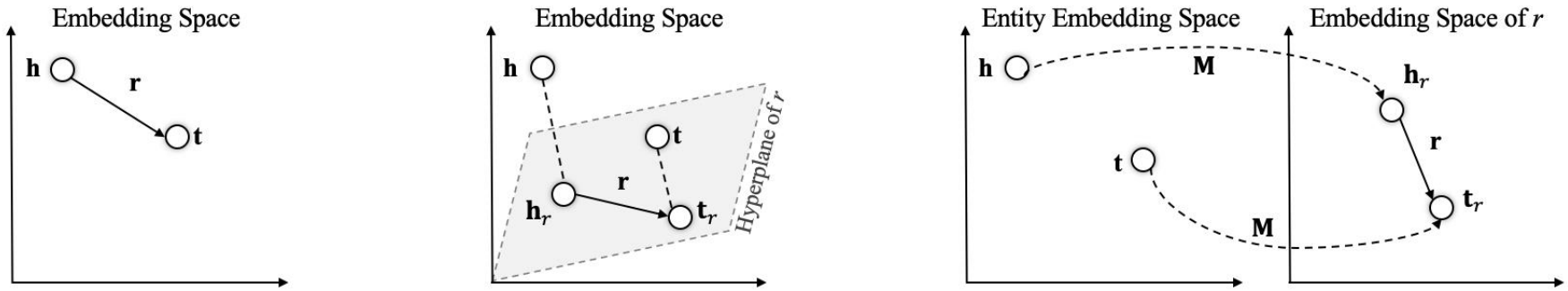
TransE
$\mathbf{h} + \mathbf{r} \approx \mathbf{t}$



Existing Approaches

- The energy function of TransX can be rewritten as:

$$f(\tau) = \left\| g_{1,r}(\mathbf{h}) + \mathbf{r} - g_{2,r}(\mathbf{t}) \right\|_{L_1/L_2}$$

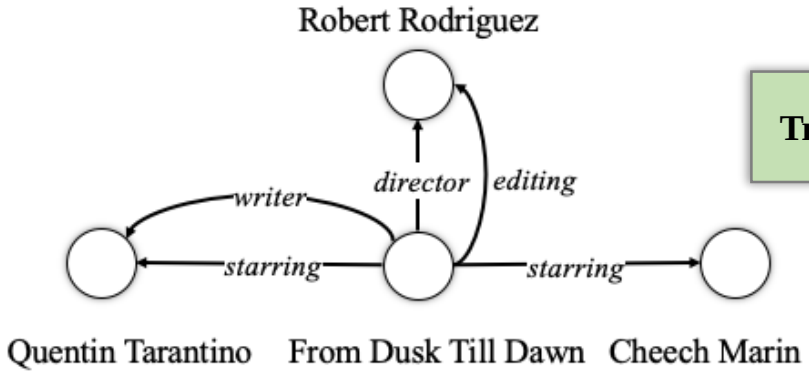


(a) TransE (b) TransH (c) TransR/TransD

However, such projections divest KG embeddings of relational structures by injecting ambiguity into entity embeddings.

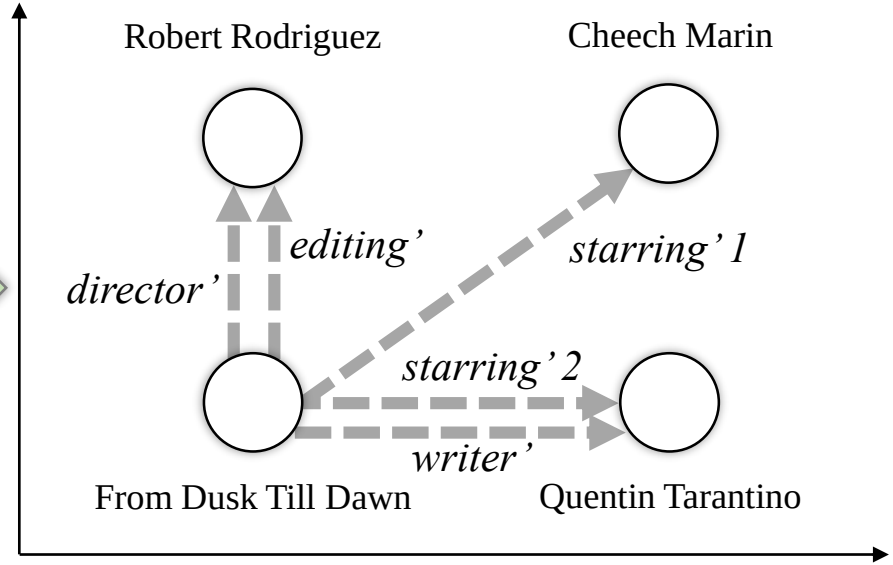
Our Approach - TransEdge

- Motivation
  - It is intuitive that
    - entities should have explicit embeddings,
    - relations should have contextualized representations as different edge embeddings.



Graph structure of the KG

TransEdge



Our Approach - TransEdge

TransEdge

- Edge-Centric Knowledge Graph Embedding

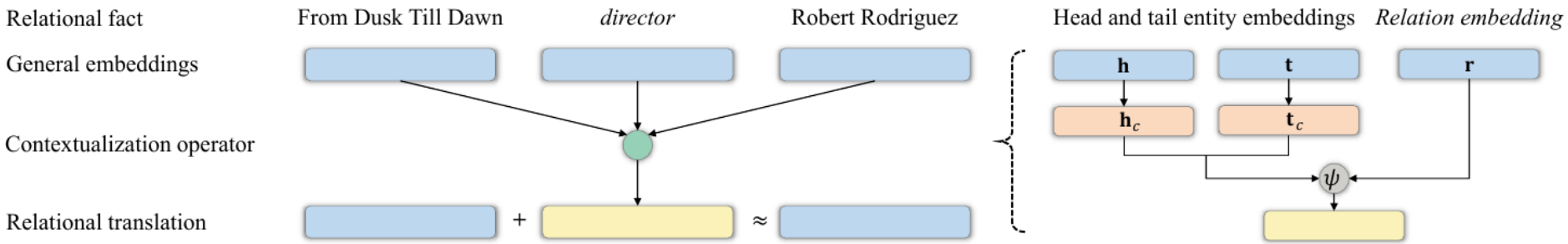


Fig. 3. Illustration of the key idea of relation-contextualized KG embeddings. The blue boxes denote the general embeddings of entities and relations, and the yellow boxes denote the contextualized representation for this relation, i.e., the edge embedding. The orange boxes are the interaction embeddings for entities, which participate in the calculation of edge embeddings. ψ is a combination operator. Better viewed in color.

TransEdge - Formulation

- Formulation

$$f(h,r,t) = ||\mathbf{h} + \underline{\psi(\mathbf{h}_c, \mathbf{t}_c, \mathbf{r})} - \mathbf{t}||$$

Edge embedding which generated by contextualization operation

Model	#Embeddings
TransE [2]	$O(n_e d + n_r d)$
TransH [32]	$O(n_e d + 2n_r d)$
TransR [17]	$O(n_e d + n_r d^2)$
TransD [12]	$O(2n_e d + 2n_r d)$
TransEdge (this paper)	$O(2n_e d + n_r d)$

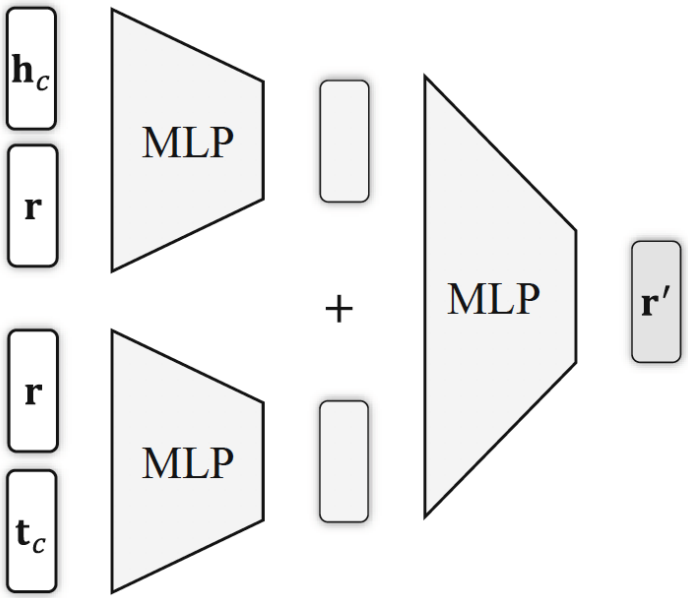
Table 1. Complexity comparison

where

- $\mathbf{r}$ is the relation embedding for relation r,
- $\mathbf{h}$ and $\mathbf{t}$ are the general embeddings for entity h and t.
  - The general embeddings capture the geometric positions and the relational semantics.
- $\mathbf{h}_c$ and $\mathbf{t}_c$ are the interaction embeddings for entity h and t.
  - The interaction embeddings are used to encode their participation in the calculation of edge embeddings.

TransEdge - Contextualization Operation $\psi(\mathbf{h}_c, \mathbf{t}_c, \mathbf{r})$

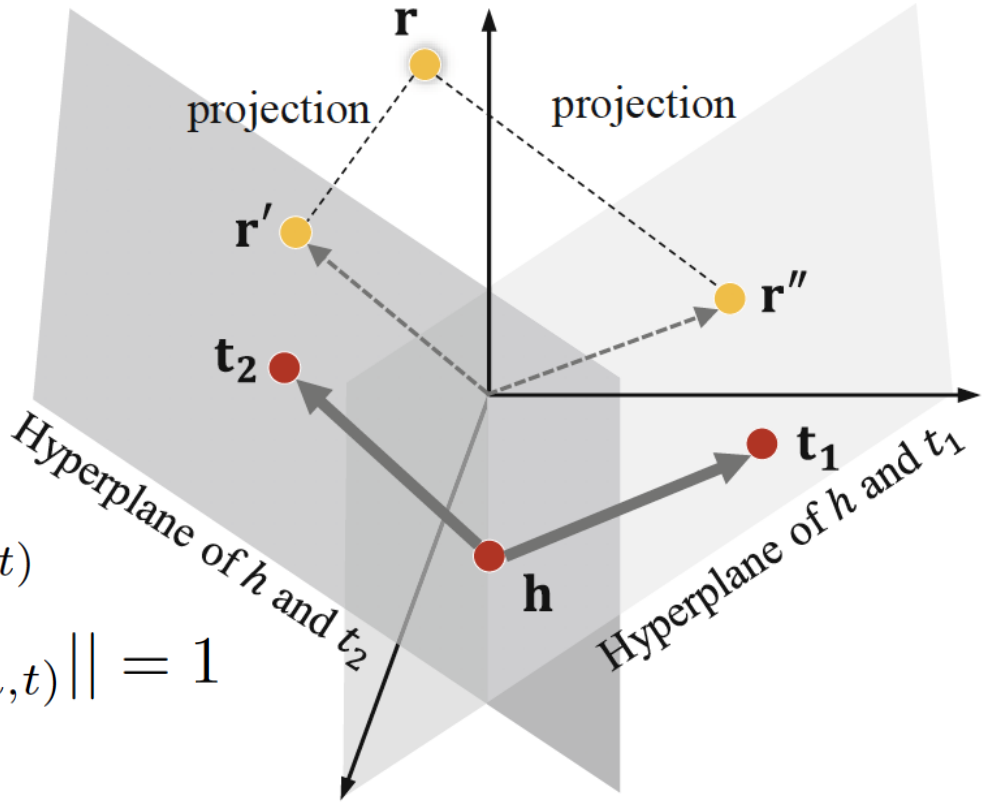
- Context compression (CC)
  - uses multilayer perceptrons (MLPs) to compress the embeddings of the edge direction and label



$$\psi(\mathbf{h}_c, \mathbf{t}_c, \mathbf{r}) = \mathrm{MLP}_1(\mathrm{MLP}_2([\mathbf{h}_c; \mathbf{r}]) + \mathrm{MLP}_3([\mathbf{r}; \mathbf{t}_c]))$$

TransEdge - Contextualization Operation $\psi(\mathbf{h}_c, \mathbf{t}_c, \mathbf{r})$

- Context projection (CP)
  - TransH: project entity onto hyperplanes
  - TransEdge: project relation r onto hyperplanes



$$\psi(\mathbf{h}_c, \mathbf{t}_c, \mathbf{r}) = \mathbf{r} - \mathbf{w}_{(h,t)}^{\top} \mathbf{r}\, \mathbf{w}_{(h,t)}$$

$$\mathbf{w}_{(h,t)} = \mathrm{MLP}([\mathbf{h}_c; \mathbf{t}_c]), \text{ s.t. } ||\mathbf{w}_{(h,t)}|| = 1$$

TransEdge – Loss Function

- Conventional marginal ranking loss

$$\mathcal{L} = \sum_{(h,r,t)\in\mathcal{T}} [f(h,r,t) - \gamma_1]_+ + \sum_{(h',r',t')\in\mathcal{T}^-} \alpha\,[\gamma_2 - f(h',r',t')]_+, \qquad (5)$$

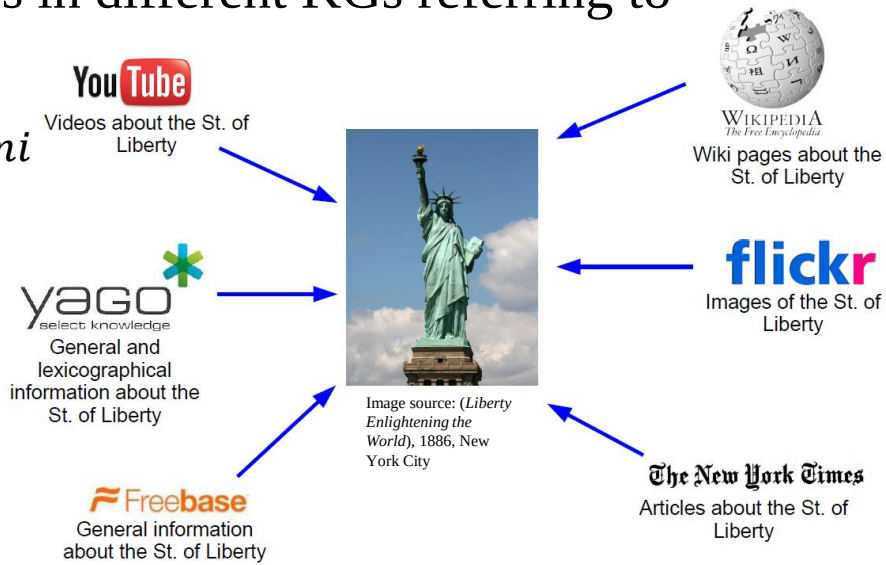
where $[x]_+ = \max(0, x)$, γ_1 and γ_2 are the hyper-parameters to control the energy of triples and $\gamma_1 < \gamma_2$. α is a hyper-parameter to balance the positive and negative relational triples. $\mathcal{T}^-$ denotes the set of negative triples, which can be generated by some heuristic strategies. Here, we adopt the truncated negative sampling method [27], which generates negative triples by replacing either the head or tail entities of positive relational triples with some random neighbors of these entities.

Task 1: Entity Alignment

- Entity alignment aims to find entities in different KGs referring to the same real-world identity
- Implementation: minimize $\mathcal{L} + \mathcal{L}_{semi}$
  - Semi-supervised: use bootstrapping strategy to select the likely-aligned entity pairs $\mathcal{D}$, and minimize the following loss function.

$$\mathcal{L}_{semi} = \sum_{(\mathbf{e}_1, \mathbf{e}_2) \in \mathcal{D}} ||\mathbf{e}_1 - \mathbf{e}_2||$$

where $\mathcal{D} = \{(\mathbf{e}_1, \mathbf{e}_2) | \cos(\mathbf{e}_1, \mathbf{e}_2) > s\}$



Task 2: Link Prediction

- Link prediction is the task of inferring the missing head or tail entities when given incomplete relational triples.
  - e.g., given $(\_\_, capitalOf, New\_Zealand)$, the link prediction models are expected to rank the right head entity Wellington at the first place.
- Implementation: minimize $\mathcal{L}$

$$\mathcal{L} = \sum_{(h,r,t)\in\mathcal{T}} [f(h,r,t) - \gamma_1]_+ + \sum_{(h',r',t')\in\mathcal{T}^-} \alpha\,[\gamma_2 - f(h',r',t')]_+, \tag{5}$$

Experiments

	DBP-WD				DBP-YG			
	Hits@1	Hits@10	MRR	MR	Hits@1	Hits@10	MRR	MR
MTransE [4] ‡	0.281	0.520	0.363	–	0.252	0.493	0.334	–
IPTransE [36] ‡	0.349	0.638	0.447	–	0.297	0.558	0.386	–
JAPE [25] ‡	0.318	0.589	0.411	–	0.236	0.484	0.320	–
AlignE [26] ‡	0.566	0.827	0.655	–	0.633	0.848	0.707	–
BootEA [26] ‡	0.748	0.898	0.801	–	0.761	0.894	0.808	–
GCN-Align [33] ▽	0.479	0.760	0.578	1988	0.601	0.841	0.686	299
TransH [32] △	0.351	0.641	0.450	117	0.314	0.574	0.402	90
TransR [17] △	0.013	0.062	0.031	2773	0.010	0.052	0.026	2852
TransD [12] △	0.362	0.651	0.456	152	0.335	0.597	0.421	90
HolE [20] △	0.223	0.452	0.289	811	0.250	0.484	0.327	437
SimplE [14] ◇	0.169	0.328	0.223	3278	0.131	0.282	0.183	3282
ProjE [24] ◇	0.312	0.504	0.382	2518	0.366	0.573	0.436	1672
ConvE [8] ◇	0.403	0.628	0.483	1428	0.503	0.736	0.582	837
TransEdge-CC (w/o semi)	0.687	0.910	0.767	70	0.759	0.935	0.822	24
TransEdge-CP (w/o semi)	0.692	0.898	0.770	106	0.726	0.909	0.792	46
TransEdge-CC	0.732	0.926	0.803	**65**	0.784	**0.948**	**0.844**	**22**
TransEdge-CP	**0.788**	**0.938**	**0.824**	72	**0.792**	0.936	0.832	43

Experiments – Link

- Datasets:

FB15K-237, WN18RR

- Competitors
  - Translational Models
  - Bilinear Models
  - Neural Models
- Metrics:
  - Hits@k
  - MR
  - MRR

Model	Type	FB15K-237				WN18RR			
		Hits@1	Hits@10	MRR	MR	Hits@1	Hits@10	MRR	MR
TransE [2] †	Trans.	–	0.436	0.269	285	–	0.453	0.412	5429
TransH [32] †	Trans.	–	0.453	0.281	292	–	0.429	0.435	5102
TransR [17] ‡▽	Trans.	–	0.429	0.162	337	0.017	0.257	0.094	3708
TransD [12] ‡▽	Trans.	–	0.428	0.162	305	0.015	0.139	0.060	6644
PTransE [16] △	Trans.	0.210	0.501	0.314	299	0.272	0.424	0.337	5686
DistMult [34] §	Bilinear	0.155	0.419	0.241	254	0.390	0.490	0.430	5110
HolE [20] ♮▽	Bilinear	0.133	0.391	0.222	–	0.284	0.346	0.308	4874
ComplEx [29] §	Bilinear	0.158	0.428	0.247	339	0.410	0.510	0.440	5261
Analogy [18] ♯▽	Bilinear	0.131	0.405	0.219	–	0.389	0.441	0.407	3836
ProjE [24]	Neural	–	0.461	0.294	246	–	0.474	0.453	4407
ConvE [8]	Neural	0.239	0.491	0.316	246	0.390	0.480	0.460	5277
R-GCN [23]	Neural	0.153	0.414	0.248	–	–	–	–	–
ConvKB [19]	Neural	–	0.517	**0.396**	257	–	0.525	0.248	2554
CACL [21]	Neural	–	0.487	0.349	235	–	0.543	0.472	3154
SimplE [14] □	Bilinear	0.225	0.461	0.230	–	–	–	–	–
CrossE [35] ◇	Bilinear	0.211	0.474	0.299	–	0.373	0.394	0.374	6091
RotatE [27]	Bilinear	0.241	**0.533**	0.338	**177**	0.428	**0.571**	**0.476**	3340
TransEdge-CC	Trans.	0.227	0.482	0.310	305	0.411	0.516	0.439	**2452**
TransEdge-CP	Trans.	**0.243**	0.512	0.333	219	**0.433**	0.487	0.451	4866

Experiments - Analysis

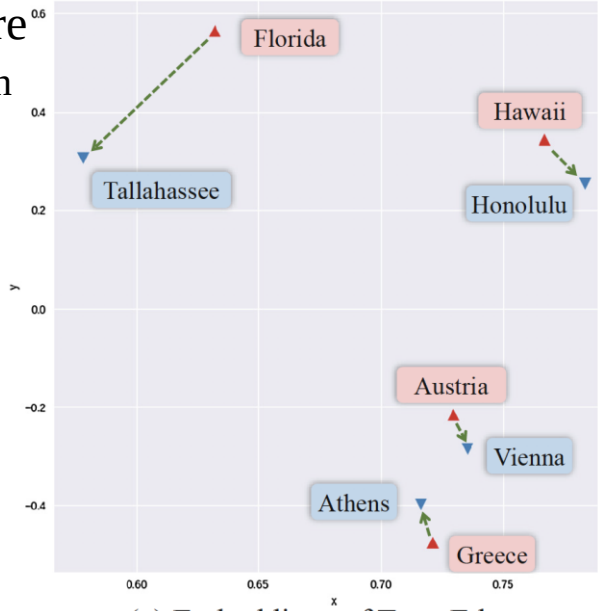
- One Entity Pair with Multiple Relations
  - Synthesize KG with double relations
    - Create a dummy relation r' for each relation r, and create a dummy triple (h, r', t) for each triple (h, r, t).
  - Results
    - TransEdge shows less variation than MTransE

Table 5. Entity alignment results on DBP15K with double relations

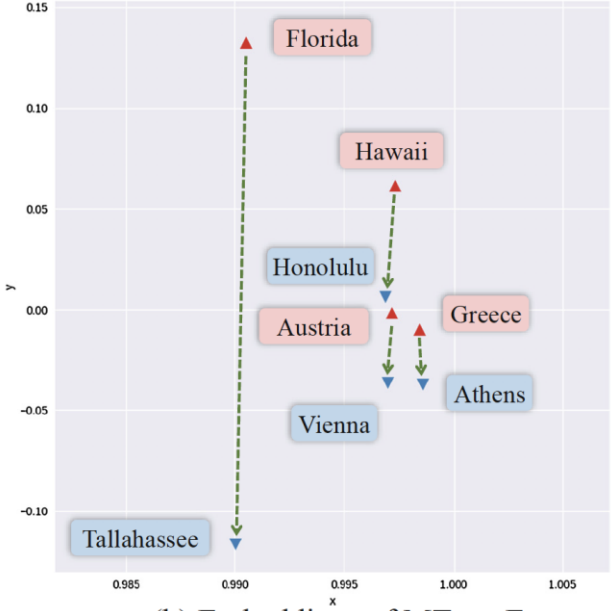
	DBP_{ZH-EN} (double)		DBP_{JA-EN} (double)		DBP_{FR-EN} (double)	
	Hits@1	Hits@1↓	Hits@1	Hits@1↓	Hits@1	Hits@1↓
MTransE [4]	0.230	25.32%	0.232	16.85%	0.208	14.75%
TransEdge-CC (w/o semi)	0.601	3.38%	0.578	3.82%	0.585	5.18%
TransEdge-CP (w/o semi)	**0.652**	**1.06%**	**0.623**	**3.56%**	**0.641**	**1.23%**

Experiments - Analysis

- Multiple Entity Pairs with One Relation
  - Triples w.r.t. relation *capital*
  - Translation vectors are
    - flexible and robust in TransEdge,
    - parallel in MTransE



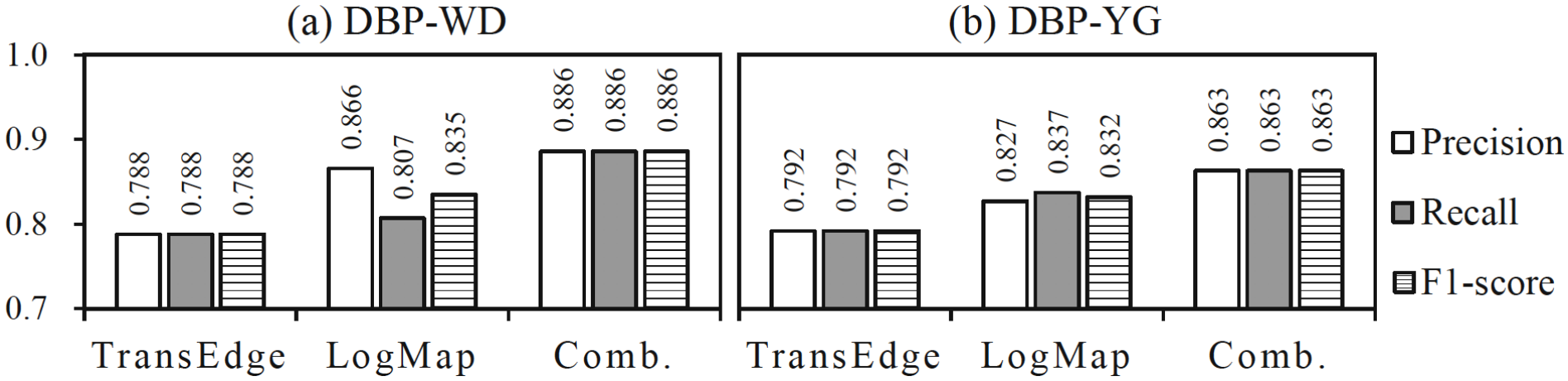
(a) Embeddings of TransEdge



(b) Embeddings of MTransE

Experiments – Conventional Entity Alignment

- TransEdge vs LogMap vs Comb.
  - Comb. combines results from TransEdge and LogMap by voting based on the predicted similarity.
- Datasets (monolingual): DBP-WD, DBP-YG



Conclusion

Conclusion

- TransEdge is a novel translational KGE model, which translating the edge embeddings rather than relation embeddings.
- To the best of our knowledge, TransEdge is the first KGE model that achieves the state-of-the-art (Hits@1) performance on both entity alignment and link prediction.
- For future work, we plan to study techniques like language models to represent multi-hop relation contexts. We also want to incorporate other proximity measures into the preserved KG structures, such as attribute similarity.

Thank you for your time!

- Source code: https://github.com/nju-websoft/TransEdge