PAKDD 2023

Improving Knowledge Graph Entity Alignment with Graph Augmentation

Feng Xie, Xiang Zeng, Bin Zhou✉, Yusong Tan
College of Computer, National University of Defense Technology
Changsha, China
{xiefeng,zengxiang,binzhou,ystan}@nudt.edu.cn

25-28 May, 2023, Osaka, Japan

Outline

- Introduction
- Methodology
    - Entity-Relation Encoder
    - Model Training with Graph Augmentation
- Experiments
- Conclusions

Introduction

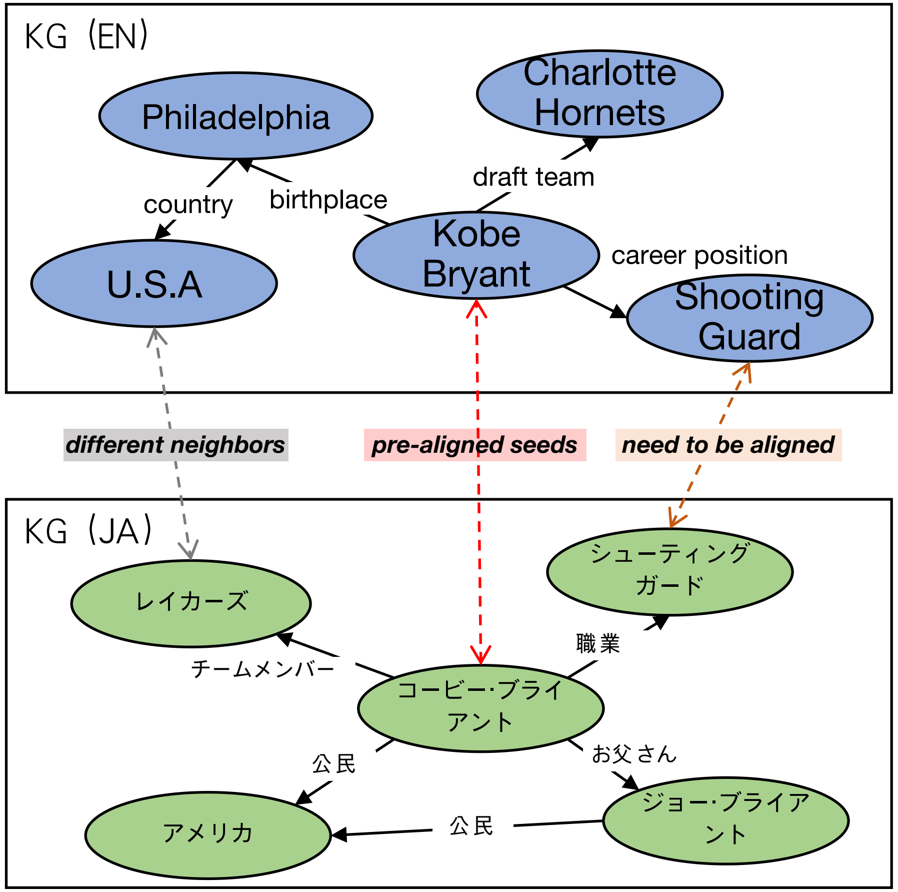
Knowledge graphs (KGs) can effectively organize and represent facts about the world in a structured fashion.

However, knowledge contained in different KGs is far from complete yet complementary [1].

Entity Alignment

Definition: link semantically equivalent entities located on different KGs

- ✓ facilitate knowledge integration
- ✓ promote knowledge-driven applications

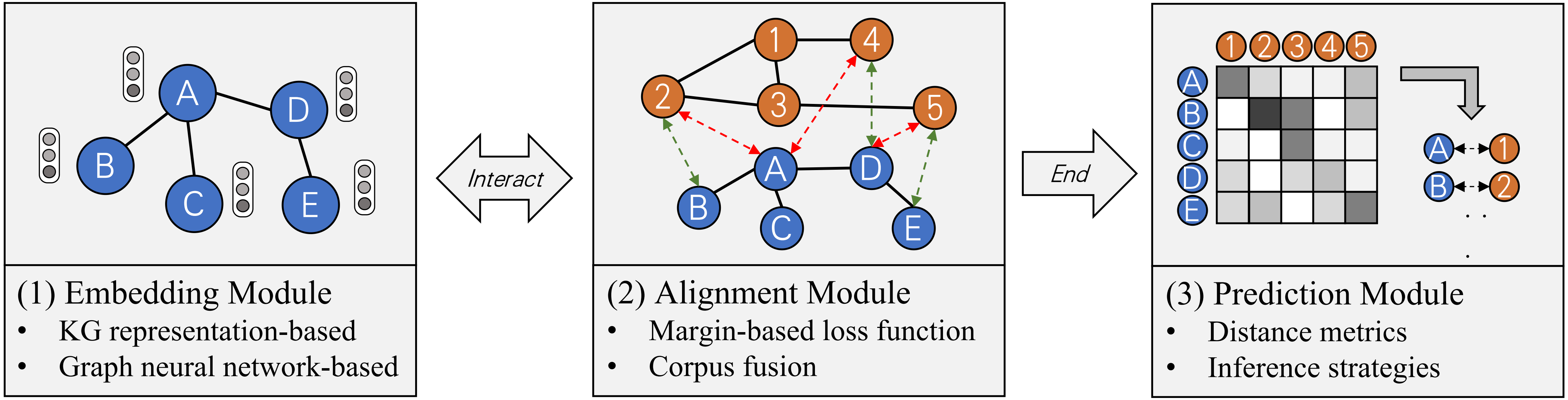


[1] Informed multi-context entity alignment, 2022, WSDM

Embedding-based EA

Embedding-based EA methods dominate current EA research and achieve promising results [1]:

- generating low-dimensional embeddings (latent representations) for entities via KG encoder,
- pulling two KGs into a unified embedding space through pre-aligned seeds,
- pairing each entity by distance metrics and inference strategies.

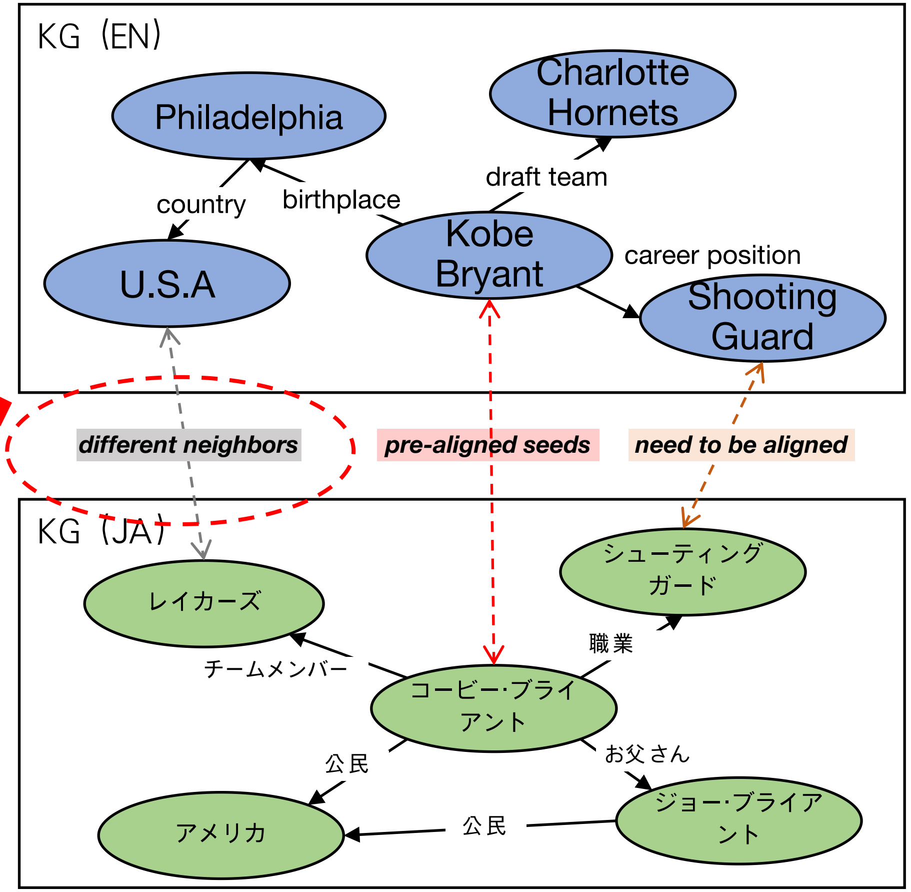


[1] An Experimental Study of State-of-the-Art Entity Alignment Approaches, 2020, TKDE

Embedding-based EA

- GNN-based methods suffer from the structural heterogeneity issue that especially appears in the real KG distributions.
  - For example, <Kobe Bryant, birthplace, Philadelphia> and <コービー・ブライアント, チームメンバー, レイカーズ> are different relational neighbors for central entity *Kobe*.
- Existing methods still ignore the heterogeneous representation learning for vast unseen (unlabeled) entities.
  - Trans-based encoders that can only capture local semantics, while GNN-based encoders only learn from subgraphs with few pre-aligned seeds

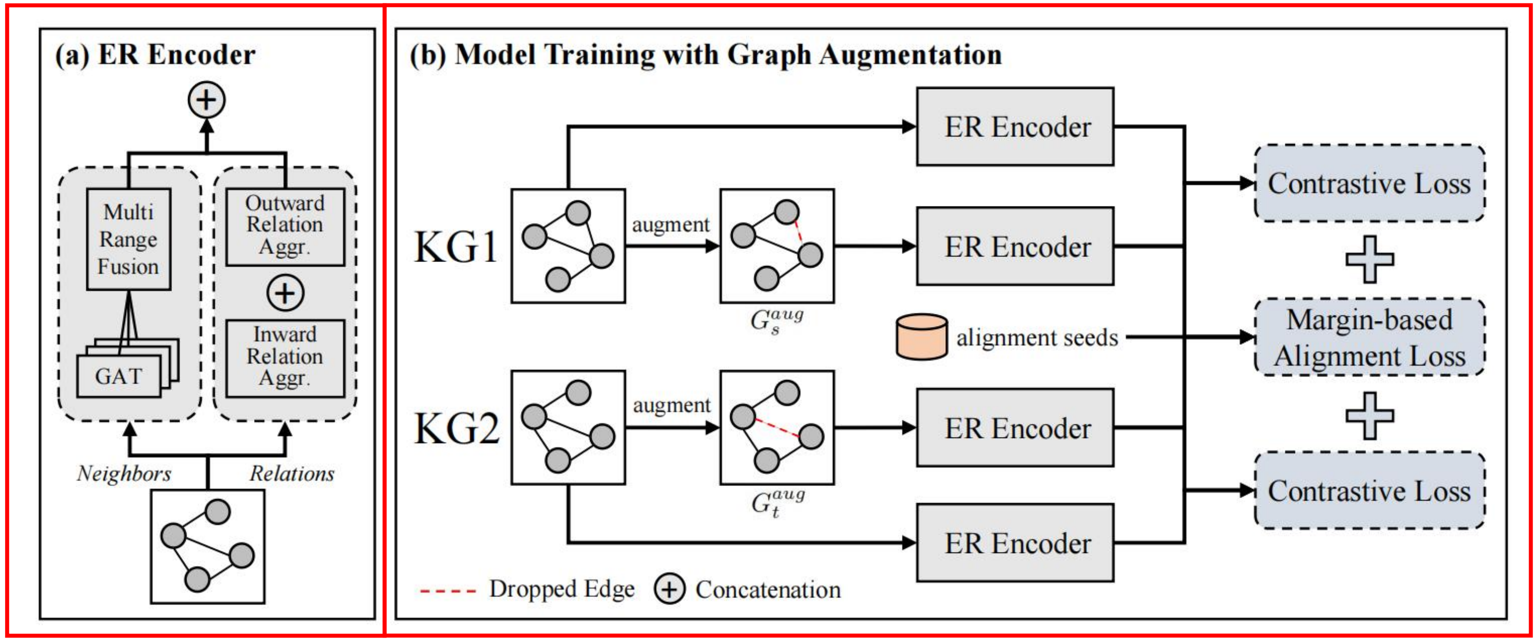
How to design a novel model to mitigate the negative influence caused by structural heterogeneity and sparse seeds?



Outline

- Introduction
- Methodology
  - Entity-Relation Encoder
  - Model Training with Graph Augmentation
- Experiments
- Conclusions

Overview of the proposed GAEA



- Entity-Relation (ER) Encoder: generating entity representations
- Model Training with Graph Augmentation: performing representation learning
- Alignment Inference: applying Faiss[1] to accelerate the inference process

[1] https://github.com/facebookresearch/faiss

Entity-Relation (ER) Encoder

- **Neighborhood aggregator**
  - Applying GAT to aggregate neighbors

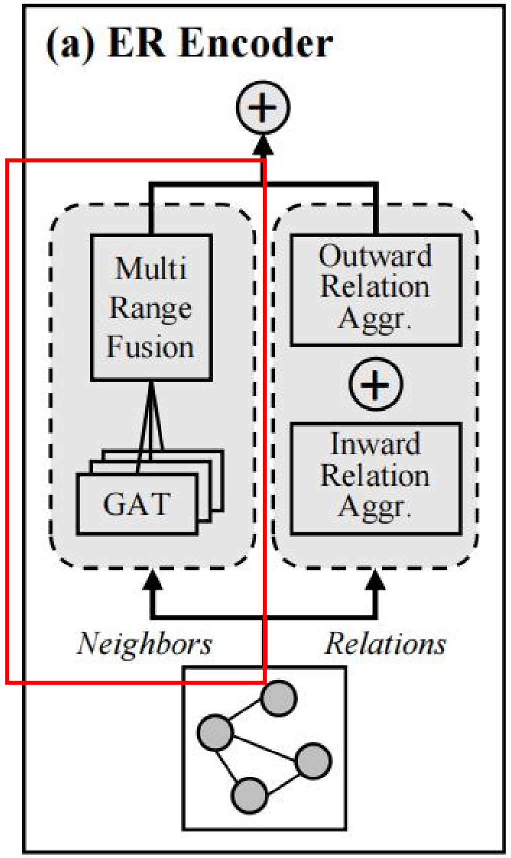
$$\mathbf{h}_{e_i}^{(l)} = \sum_{e_j \in N_{e_i}} \alpha_{ij} \mathbf{h}_{e_j}^{(l-1)},$$

$$\alpha_{ij} = \frac{\exp(\mathrm{LeakyReLU}(\mathbf{a}^\top [\mathbf{W}_g \mathbf{h}_{e_i} \oplus \mathbf{W}_g \mathbf{h}_{e_j}]))}{\sum_{e_k \in N_{e_i}} \exp(\mathrm{LeakyReLU}(\mathbf{a}^\top [\mathbf{W}_g \mathbf{h}_{e_i} \oplus \mathbf{W}_g \mathbf{h}_{e_k}]))},$$

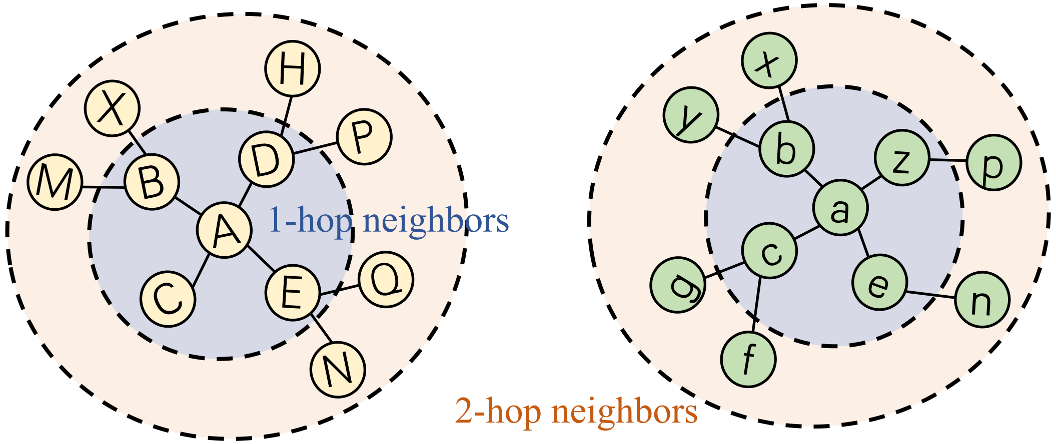
  - Multi-range fusion

$$[\hat{\mathbf{h}}_{e_i}^{(1)}, ..., \hat{\mathbf{h}}_{e_i}^{(L)}] = \mathrm{softmax}(\frac{(\mathbf{H}_{e_i}^m \mathbf{W}_q)(\mathbf{H}_{e_i}^m \mathbf{W}_k)^\top}{\sqrt{d_{ent}}}) \mathbf{H}_{e_i}^m$$

$$\mathbf{h}_{e_i}^n = \frac{1}{L} \sum_{l=1}^{L} \hat{\mathbf{h}}_{e_i}^{(l)},$$



Structure Assumption: equivalent entities tend to have similar neighbor structures. [1]



[1] Cross-lingual Knowledge Graph Alignment via Graph Convolutional Networks, 2018, EMNLP

Entity-Relation (ER) Encoder

- **Neighborhood aggregator**
  - Applying GAT to aggregate neighbors

$$\mathbf{h}_{e_i}^{(l)} = \sum_{e_j \in N_{e_i}} \alpha_{ij} \mathbf{h}_{e_j}^{(l-1)},$$

$$\alpha_{ij} = \frac{\exp(\mathrm{LeakyReLU}(\mathbf{a}^\top [\mathbf{W}_g \mathbf{h}_{e_i} \oplus \mathbf{W}_g \mathbf{h}_{e_j}]))}{\sum_{e_k \in N_{e_i}} \exp(\mathrm{LeakyReLU}(\mathbf{a}^\top [\mathbf{W}_g \mathbf{h}_{e_i} \oplus \mathbf{W}_g \mathbf{h}_{e_k}]))},$$

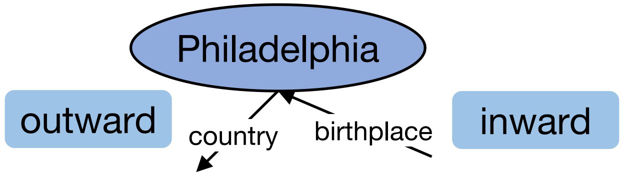
  - Multi-range fusion

$$[\hat{\mathbf{h}}_{e_i}^{(1)}, ..., \hat{\mathbf{h}}_{e_i}^{(L)}] = \mathrm{softmax}(\frac{(\mathbf{H}_{e_i}^m \mathbf{W}_q)(\mathbf{H}_{e_i}^m \mathbf{W}_k)^\top}{\sqrt{d_{ent}}}) \mathbf{H}_{e_i}^m$$

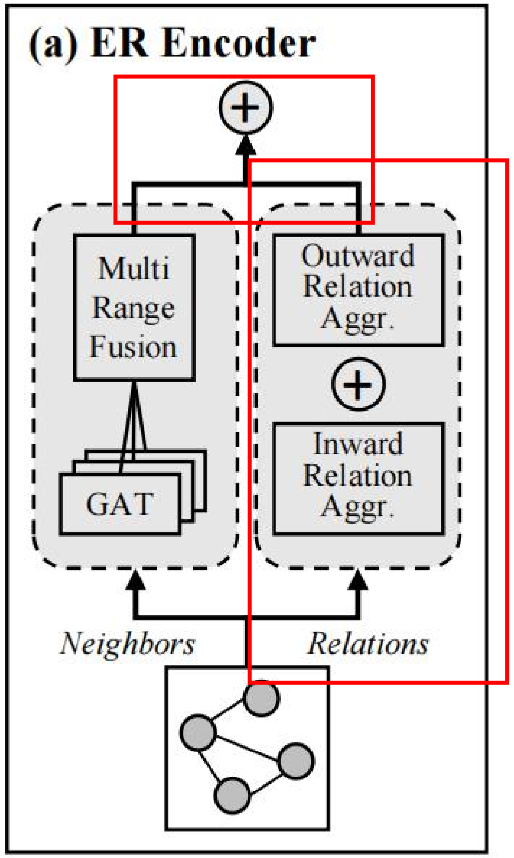
$$\mathbf{h}_{e_i}^n = \frac{1}{L} \sum_{l=1}^{L} \hat{\mathbf{h}}_{e_i}^{(l)},$$

- **Relation aggregator**
  - gather outward relation semantics and inward relation semantics separately to provide supplementary alignment signals for heterogeneous KGs

$$\mathbf{h}_{e_i}^r = \frac{1}{|N_{e_i}^{r+}|} \sum_{r \in N_{e_i}^{r+}} \mathbf{h}_r^{rel} \oplus \frac{1}{|N_{e_i}^{r-}|} \sum_{r \in N_{e_i}^{r-}} \mathbf{h}_r^{rel},$$

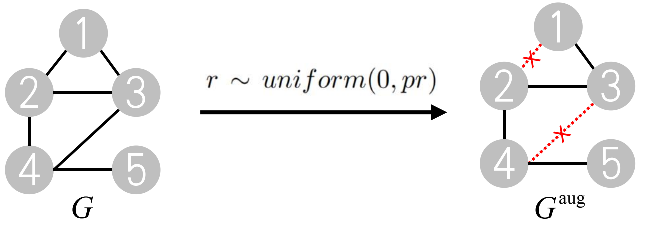


- **Feature fusion** $\tilde{\mathbf{h}}_{e_i} = \mathbf{h}_{e_i}^n \oplus \mathbf{h}_{e_i}^r,$



Model Training with Graph Augmentation

- **Augmented graph generation**



edge dropping:
✓ Do not bring logic errors
✓ Do not consider long-tail entities

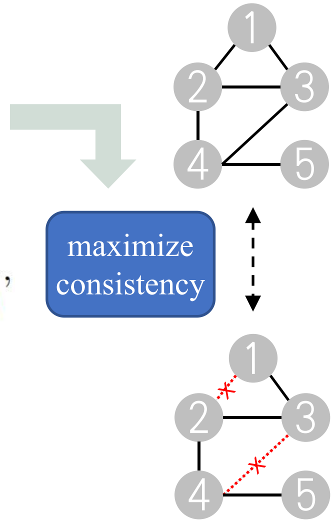
- **Margin-based alignment loss**

$$\mathcal{L}_a = \sum_{(e_i,e_j)\in S} \sum_{(\bar{e}_i,\bar{e}_j)\in \bar{S}_{(e_i,e_j)}} \left[||\tilde{\mathbf{h}}^{aug}_{e_i} - \tilde{\mathbf{h}}^{aug}_{e_j}||_{L2} + \rho - ||\tilde{\mathbf{h}}^{aug}_{\bar{e}_i} - \tilde{\mathbf{h}}^{aug}_{\bar{e}_j}||_{L2} \right]_+$$

- **Contrastive loss**

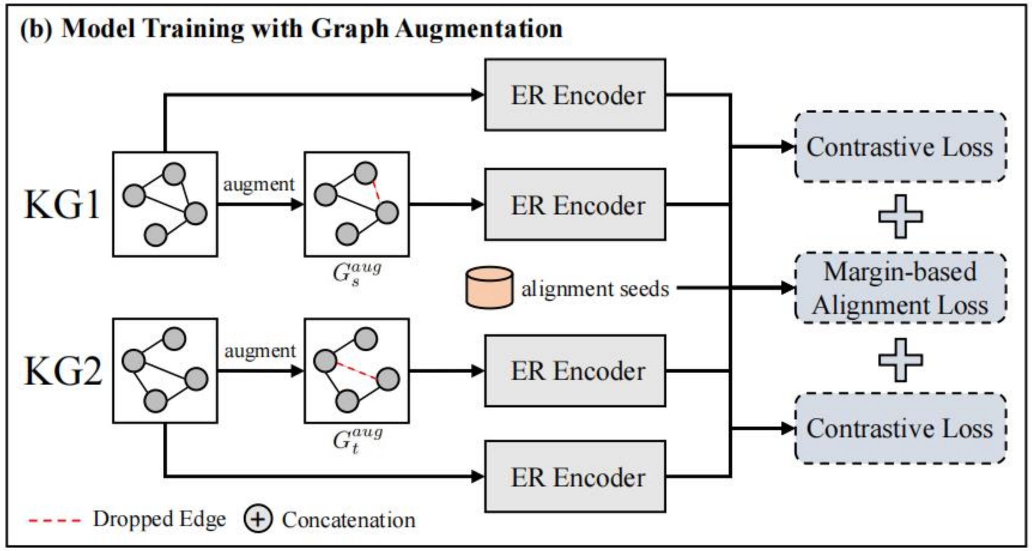
$$\mathcal{L}_c = \sum_{z=\{s,t\}} \frac{1}{2|E_z|} \sum_{e_i \in E_z} \left(\mathcal{L}^{(G_z,G_z^{aug})}_{c,e_i} + \mathcal{L}^{(G_z^{aug},G_z)}_{c,e_i}\right),$$

$$\mathcal{L}^{(G_z,G_z^{aug})}_{c,e_i} = -\log \frac{\exp(\langle \mathrm{proj}(\tilde{\mathbf{h}}_{e_i}), \mathrm{proj}(\tilde{\mathbf{h}}^{aug}_{e_i})\rangle)}{\sum_{e_k \in E_z} \exp(\langle \mathrm{proj}(\tilde{\mathbf{h}}_{e_i}), \mathrm{proj}(\tilde{\mathbf{h}}^{aug}_{e_k})\rangle)},$$



- **Model Training**

$$\boxed{\mathcal{L} = \mathcal{L}_a + \lambda \mathcal{L}_c,} \quad \triangleright \text{ Adam step}$$



Outline

- Introduction
- Methodology
  - Entity-Relation Encoder
  - Model Training with Graph Augmentation
- Experiments
- Conclusions

Experimental setups

- **Datasets**

  We use the 15K benchmark dataset (version 1.0) in OpenEA for evaluation

  The KGs in V1 are sparse and the entities thereof follow the real-world degree distribution

- **Metrics**

  Entity alignment is a typical ranking problem

  We use Hit@k (k=1, 5) and MRR (Mean Reciprocal Rank) as the evaluation metrics

- **Baselines**

GNNs	GCN (ICLR2017), GAT (ICLR 2018)
Trans-based	MTransE (IJCAI2017), IPTransE (IJCAI2017), SEA (WWW2019)
GNN-based	GCN-Align (EMNLP2018), AliNet (AAAI2019), HyperKA (EMNLP2020), KE-GCN (WWW2021)
others	RSNs (ICML2019), IMEA (WSDM2022)

Overall performance

Models	EN-FR-15K			EN-DE-15K			D-W-15K			D-Y-15K		
	Hit@1	Hit@5	MRR	Hit@1	Hit@5	MRR	Hit@1	Hit@5	MRR	Hit@1	Hit@5	MRR
GCN*	.210	.414	.304	.304	.497	.394	.208	.367	.284	.343	.503	.416
GAT*	.297	.585	.426	.542	.737	.630	.383	.622	.489	.468	.707	.573
MTrasnE†	.247	.467	.351	.307	.518	.407	.259	.461	.354	.463	.675	.559
SEA†	.280	.530	.397	.530	.718	.617	.360	.572	.458	.500	.706	.591
IPTransE†	.169	.320	.243	.350	.515	.430	.232	.380	.303	.313	.456	.378
RSNs†	.393	.595	.487	.587	.752	.662	.441	.615	.521	.514	.655	.580
GCN-Align†	.338	.589	.451	.481	.679	.571	.364	.580	.461	.465	.626	.536
AliNet‡	.364	.597	.467	.604	.759	.673	.440	.628	.522	.559	.690	.617
HyperKA‡	.353	.630	.477	.560	.780	.656	.440	.686	.548	.568	.777	.659
KE-GCN‡	.408	.670	.524	.658	.822	.730	.519	.727	.608	.560	.750	.644
IMEA‡	.458	.720	.574	.639	.827	.724	.527	.753	.626	**.639**	**.804**	**.712**
GAEA	**.486**	**.746**	**.602**	**.684**	**.854**	**.760**	**.562**	**.768**	**.654**	.608	.791	.688
w/o *rel.*	.324	.626	.458	.593	.785	.678	.409	.666	.521	.502	.743	.605

- Experimental results show that our proposed GAEA outperforms other models in most tasks, especially in cross-lingual settings.
- The performance of models utilizing knowledge representation learning as the encoder are inferior compared with the models applying GNNs as the encoder.

Ablation study

Models	EN-DE-15K			D-W-15K		
	Hit@1	Hit@5	MRR	Hit@1	Hit@5	MRR
GAEA	**.684**	**.854**	**.760**	**.562**	**.768**	**.654**
$-gaal.$	.674	.848	.751	.557	.764	.650
$-\mathcal{L}_c$	.665	.841	.744	.544	.755	.639

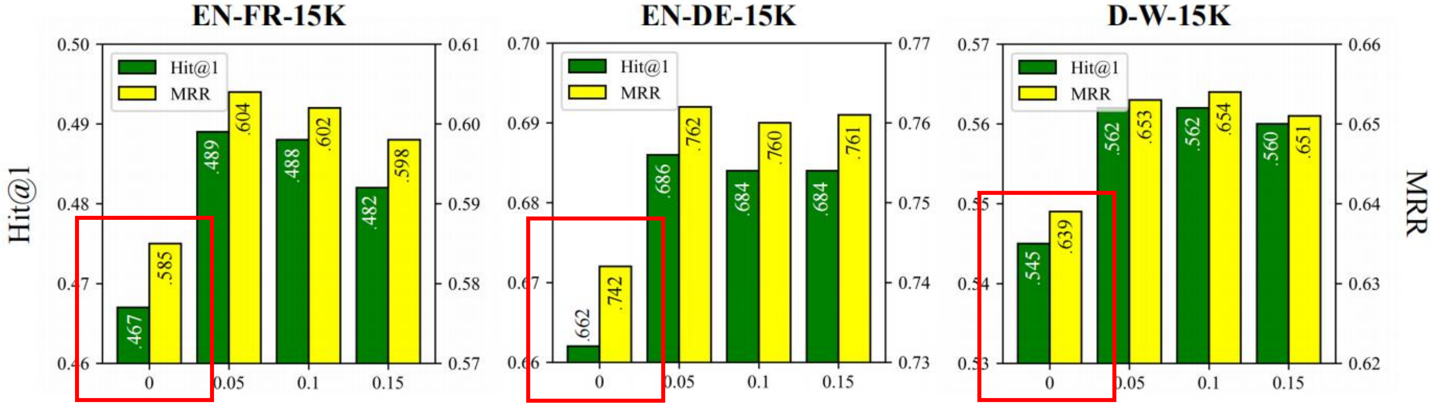
The results show that utilizing graph augmentation can have positive impacts on EA and consistently get better performance.

#Params comparison

Models	#Params (M)
GCN	~7.81M
AliNet	~16.18M
IMEA	~20.44M
GAEA (ours)	~8.10M

GAEA greatly reduces the number of parameters compared to IMEA while acquiring decent alignment performance.

Ablation study



- The performance is worst on all three tasks when pr=0, indicating that graph augmentation can do benefit for alignment learning.
- The alignment effect is best when pr equals 0.05 or 0.1, increasing pr to 0.15 will not further improve the performance, and even bring performance drops.

Future work

- how to amplify the improvement brought by graph augmentation when there no pre-aligned seeds are given (i.e. unsupervised).
- how to conduct graph augmentation learning in a highly structured KG to improve performance without introducing logic errors.

Outline

- Introduction
- Methodology
  - Entity-Relation Encoder
  - Model Training with Graph Augmentation
- Experiments
- Conclusions

Conslusions

We propose GAEA, a novel entity alignment approach based on graph augmentation.

- We design a simple Entity-Relation (ER) Encoder to generate latent representations for entities via jointly capturing neighborhood structures and relation semantics.
- We apply graph augmentation to create two graph views for margin-based alignment learning and contrastive entity representation learning.
- Extensive results on OpenEA dataset verified the effectiveness of our method.

Thanks for your listening!

For more information, please refer to our paper or source codes:

Open source: https://github.com/Xiefeng69/GAEA
website: https://xiefeng69.github.io/