NEURAL INFORMATION PROCESSING SYSTEMS

Arxiv links Code links

Lambda: Learning Matchable Prior For Entity Alignment with Unlabeled Dangling Cases

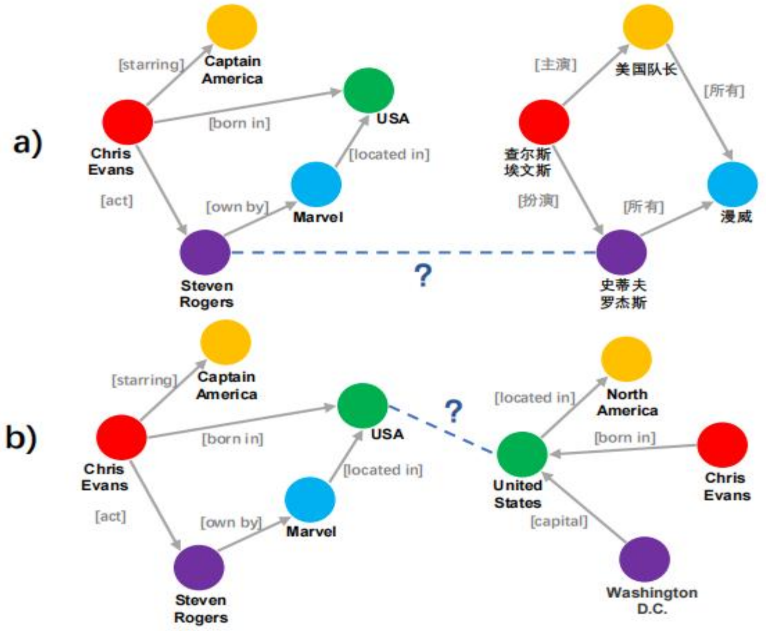
Hang Yin · Liyao Xiang* · Dong Ding · Yuheng He · Yihan Wu · Pengzhi Chu · Xinbing Wang · Chenghu Zhou

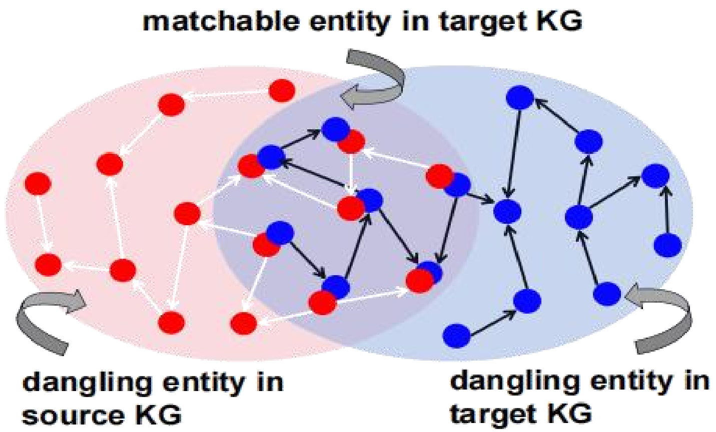
Speaker: Hang Yin

Correspondence to: Liyao Xiang

上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

Task definition: Entity alignment with unlabeled dangling cases





Method	Side Info	Dangling Labels
MTransE	✗	✔
AliNet	✗	✔
UED	✔	✗
SoTead	✔	✗
MHP	✗	✔ + high-order info
Our Work	✗	✗

- Entity alignment (EA) seeks identical entities in different knowledge graphs, which is a long-standing task in the database research.
- Partial entities have no counterparts in the other KG, yet these entities are unlabeled.
- Previous work relies too much on side information and dangling labels.

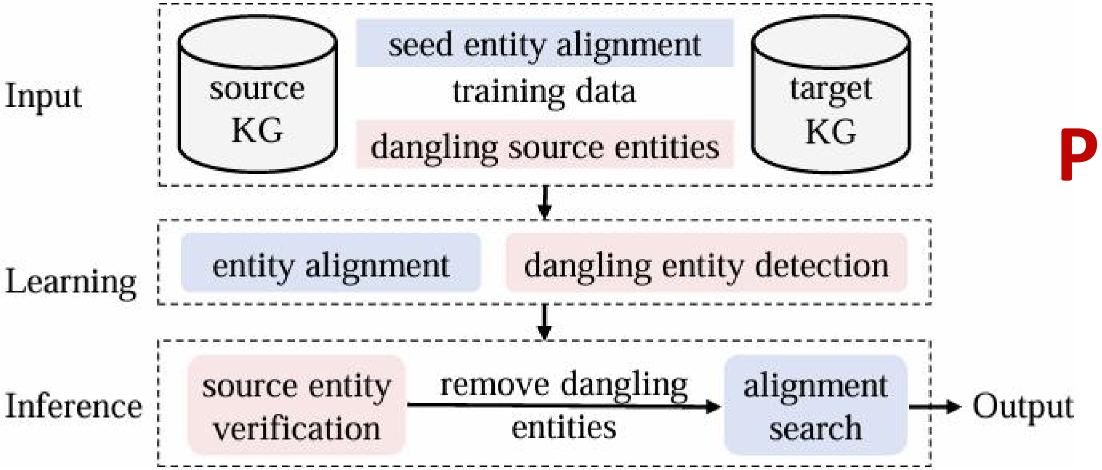
Motivation

Method	DBP15K$_{\text{ZH-EN}}$ H@1	DBP15K$_{\text{ZH-EN}}$ H@10	DBP15K$_{\text{ZH-EN}}$ H@50	DBP15K$_{\text{JA-EN}}$ H@1	DBP15K$_{\text{JA-EN}}$ H@10	DBP15K$_{\text{JA-EN}}$ H@50	DBP15K$_{\text{FR-EN}}$ H@1	DBP15K$_{\text{FR-EN}}$ H@10	DBP15K$_{\text{FR-EN}}$ H@50
BootEA	31.30↓ 20.96	59.70↓ 16.18	71.51↓ 12.91	33.77↓ 15.27	62.66↓ 11.64	73.09↓ 10.29	23.11↓ 26.72	58.39↓ 18.77	71.54↓ 14.00
TransEdge	49.91↓ 15.21	76.62↓ 9.79	83.44↓ 7.16	54.07↓ 13.42	78.01↓ 8.25	84.00↓ 6.21	48.23↓ 17.34	79.32↓ 9.70	86.69↓ 6.24
MRAEA	59.45↓ 5.62	83.04↓ 2.53	88.68↓ 1.56	61.60↓ 4.45	83.48↓ 2.21	88.65↓ 1.50	61.55↓ 6.62	85.85↓ 2.61	90.79↓ 1.69
GCN-Align	31.99↓ 10.70	62.21↓ 6.45	71.93↓ 4.31	32.08↓ 10.08	61.04↓ 5.86	70.34↓ 3.52	30.71↓ 10.50	61.64↓ 7.07	72.45↓ 5.55
RSNs	43.00↓ 8.50	62.90↓ 8.00	69.70↓ 7.00	20.60↓ 31.60	44.60↓ 26.60	53.20↓ 23.60	36.30↓ 15.30	63.30↓ 10.10	71.70↓ 7.80
MuGNN	34.66↓ 14.75	68.48↓ 9.32	80.53↓ 5.69	32.93↓ 14.68	66.68↓ 8.82	78.63↓ 5.67	34.93↓ 14.02	68.88↓ 9.69	81.67↓ 5.32
KECG	35.92↓ 12.87	65.70↓ 10.35	76.44↓ 8.06	32.31↓ 15.48	63.19↓ 11.96	74.42↓ 9.29	32.84↓ 15.47	64.78↓ 11.98	76.70↓ 8.35
AliNet	53.84↓ 0.66	73.73↓ 3.16	80.30↓ 1.59	52.69↓ 1.30	74.01↓ 2.60	80.91↓ 1.90	54.01↓ 0.58	76.19↓ 2.74	83.25↓ 1.40
Dual-AMN	60.72↓ 12.20	83.93↓ 5.22	89.45↓ 3.54	62.29↓ 10.62	83.38↓ 5.35	88.80↓ 3.21	65.33↓ 10.48	87.76↓ 4.17	92.47↓ 2.24

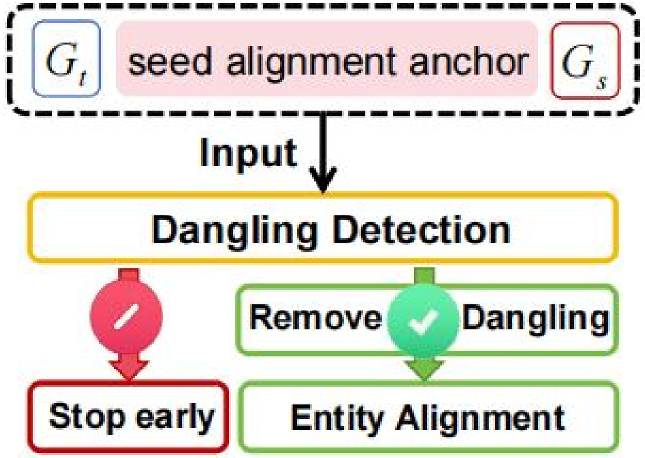
Table 8: Network alignment performance on DBP15K in the consolidated setting. The blue numbers suggest the drop from the relaxed setting (as with their original implementation).

- **We investigated the performance degradation of various existing EA methods in the face of the dangling problem, which shows that this problem is worth considering.**
- **Our work addressed EA problems without side information and dangling labels for better practical application.**

A New Framework: PU learning as a dangling detection classifier



PU learning



- **We expect to have more choices before performing the second phase EA. Cause calculation consumption of training and inference for EA should be avoided when no more potential matchable entities exist.**
- **Given partial pre-aligned matchable entities as positive samples, how to jointly predict the proportion of matchable entities in the unlabeled nodes and identify them?**

Iterative Positive-Unlabeled Learning for Dangling Detection

· We provide a theoretical analysis of PU learning on ***Unbiasedness***, ***Uniform Deviation Bound*** and ***Convergence***:

Theorem 1. $\widehat{R}_{\mathrm{pu}}(g)$ is the **Unbiased risk estimator** of $R(g)$.

$$\widehat{R}_{\mathrm{pu}}(g) = \pi_{\mathrm{p}} \widehat{R}_{\mathrm{p}}^{+}(g) + \frac{\pi_{\mathrm{n}}}{\pi_{\mathrm{n}}^{\mathrm{u}}} \cdot \left[\widehat{R}_{\mathrm{u}}^{-}(g) - \pi_{\mathrm{p}}^{\mathrm{u}} \widehat{R}_{\mathrm{p}}^{-}(g)\right]$$

★ ***Unbiasedness.***

Theorem 2. $\widehat{R}_{\mathrm{pu}}(g)$ gets a tighter **uniform deviation bound** than the classic *Non-negative Risk Estimator.*

★ ***Uniform Deviation Bound.***

It depends on accurate class prior estimation!

To find dangling entities. → **To find its class prior estimation.**

Theorem 3. The iterative process of iPULE based on $\widehat{R}_{\mathrm{pu}}(g)$ is a special case of the EM algorithm thus convergent.

★ ***Convergence.***

(Devation in our appendix)

Loss Function

Positive Unlabeled Loss Function. Since it is evident that all negative samples exist in unlabeled data. Therefore $\frac{\pi_{\mathrm{n}}}{\pi_{\mathrm{n}}^{\mathrm{u}}} < 1$, we apply a hyper-parameter $\alpha = \frac{\pi_{\mathrm{n}}^{\mathrm{u}}}{\pi_{\mathrm{n}}} > 1$ to scale $\pi_{\mathrm{p}} \widehat{R}_{\mathrm{p}}^{+}(g)$ equivalently and $\max(\cdot)$ to restrict the estimated $\pi_{\mathrm{n}} R_{\mathrm{n}}^{-}(g) \geq 0$. The PU learning loss function is formulated as:

$$\mathcal{L}_{\mathrm{pu}} = \alpha \pi_{\mathrm{p}} \widehat{R}_{\mathrm{p}}^{+}(g) + \max\{0, \widehat{R}_{\mathrm{u}}^{-}(g) - \pi_{\mathrm{p}}^{\mathrm{u}} \widehat{R}_{\mathrm{p}}^{-}(g)\},$$

We specify the corresponding risk function using cross-entropy loss as below respectively:

$$\widehat{R}_{\mathrm{p}}^{+}(g) = \frac{1}{|\mathcal{X}_p|} \sum_{e_i \in \mathcal{X}_p} \log \hat{y}_i(+1), \widehat{R}_{\mathrm{u}}^{-}(g) = \frac{1}{|\mathcal{X}_u|} \sum_{e_i \in \mathcal{X}_u} \log \hat{y}_i(-1), \widehat{R}_{\mathrm{p}}^{-}(g) = \frac{1}{|\mathcal{X}_p|} \sum_{e_i \in \mathcal{X}_p} \log \hat{y}_i(-1)$$

Positive as Positive	Unlabeled as Nagetive	Positive as Nagetive
×	×	×
(Alpha) Positive Class Prior	Unlabeled Class Prior	Unlabeled Positive Class Prior

Just need to find its Positive class prior estimation !

Algorithm

Theorem 3. The iterative process of iPULE based on $\widehat{R}_{\mathrm{pu}}(g)$ is a special case of the EM algorithm thus convergent.

Convergence.

Description

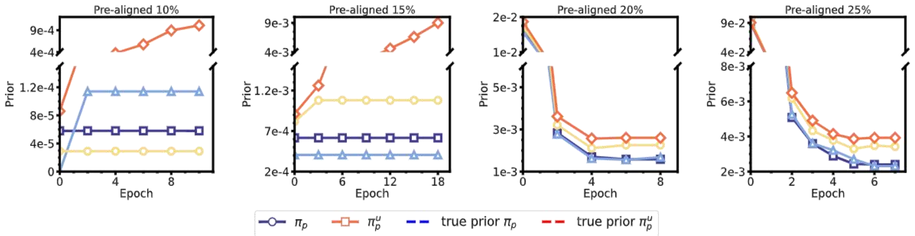
Algorithm 1 iPULE (**i**terative **PU** **L**earning with Prior **E**stimator)

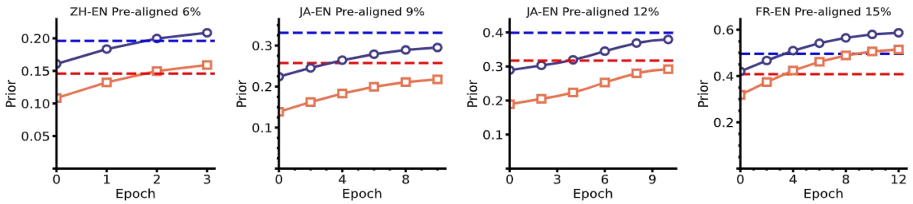
Require: G_s and G_t are treated as one input graph $G = (\mathcal{V}, \mathcal{E})$, set $\mathcal{P} = \mathcal{X}_p$ of positive nodes, set $\mathcal{U} = \mathcal{X}_u$ of unlabeled nodes, classifier f with initial parameters θ^{new}, KEESA $\mathrm{Enc}(G, \psi)$ with initial parameters ψ^{new} and cold start epoch N. $\mathcal{L}$ represents loss function during training.

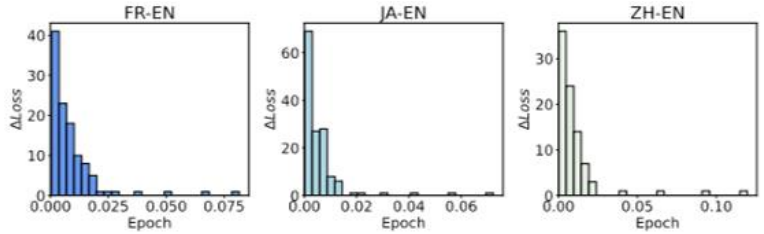
Ensure: Best parameters θ^{new}, ψ^{new} and estimated prior $\hat{\pi}_{\mathrm{p}}$ and $\hat{\pi}_{\mathrm{p}}^{\mathrm{u}}$

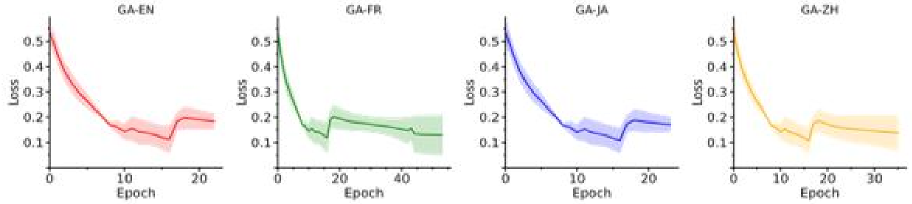
```
l^new ← ∞,   π̂_p^u ← π̂_p ← |P| / (|P|+|U|),   i ← 0,   β = β_0;          //Initial value
L ← β · L_info + (1 − β) · L_pu;                                         //Loss function of cold start
repeat
    X ← Enc(G, ψ^new);                                                     //Entity embedding matrix X
    θ^new, ψ^new ← arg min_{θ,ψ} L(θ; X, y, P, U);                        //Optimize Enc(·) and f jointly
    l^new ← L(θ^new; X, y, P, U);
until N epochs is over                                                     //Cold start
L ← L_pu;
repeat
    X ← Enc(G, ψ^new),   ŷ_i ← f(X, i; θ^new) for all i ∈ V;
    π̂_p^u ← |U|^{-1} Σ_{i∈U} I[ŷ_i(+1) > 0.5],   π̂_p ← (|P|+|U|·π̂_p^u) / (|P|+|U|);   //E step
    l ← l^new,   l^new ← L(θ^new; X, y, P, U);
    θ^new, ψ^new ← arg max_{θ,ψ} −L(θ; X, y, P, U);                      //M step
until |l − l^new| converge OR π̂_p converge
return
```

Experiments









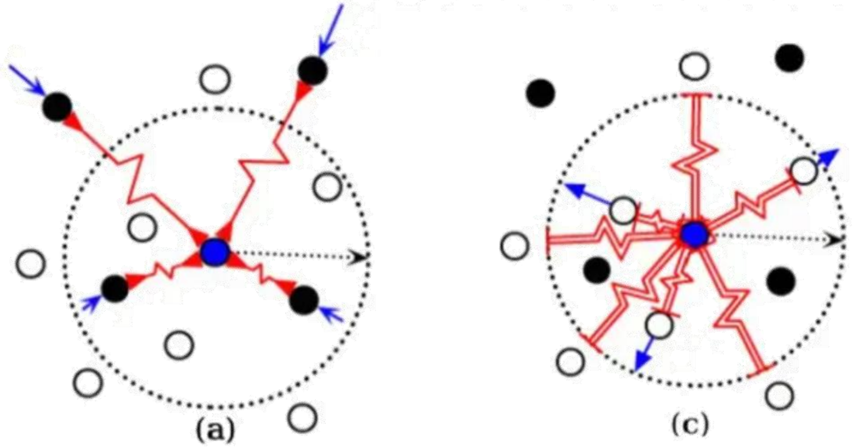
Selective Aggregation with Spectral Contrastive Learning

a. PU learning depends on Classification Discriminative.

b. Entity alignment depends on Entity-to-entity Unified Embedding Space.

$$\mathcal{L}_{\text{info}} = \sum_{e_i \in \mathcal{X}_p} \log \left[1 + \sum_{j}^{N} \exp(\lambda\, H(e_i, e_+^i, e_j^i)) \right].$$

$$H(e_i, e_+^i, e_j^i) = [\mathrm{sim}(e_i, e_j^i) - \mathrm{sim}(e_i, e_+^i) + \gamma]_+,$$



a. could be achieved by Spectral Clustering and b. could be achieved by Contrastive Learning

Contrastive learning is spectral clustering!

——(*) Zhiquan Tan, Yifan Zhang, Jingqin Yang, and Yang Yuan. Contrastive learning is spectral clustering on similarity graph, 2023.

We achieve both by Spectral Contrastive Learning and the encoder KEESA

KEESA (KG Entity Encoder with Selective Aggregation)

a. Adaptive Dangling indicator & Relation Projection Attention.

$$\boldsymbol{h}_{e_i}^{l+1} = \sigma\left(\sum_{e_j \in \mathcal{N}_{e_i} \cup \{e_i\}} \underbrace{\tanh(r_{e_j})}_{\text{adaptive dangling indicator}} \alpha_{i,j} W^{l+1} \boldsymbol{h}_{e_j}^{l}\right)$$

$$\boldsymbol{h}_{r_k}^{\rightarrow e_j} = r_{e_j} W_r \boldsymbol{h}_{r_k} \quad \text{and} \quad L_o = \left\| W_r^\top W_r - I_{d\times d} \right\|_2^2 .$$

$$\alpha_{ijk}^{l} = \frac{\exp(\boldsymbol{v}^\top \boldsymbol{h}_{r_k}^{\rightarrow e_j})}{\sum_{e_m \in \mathcal{N}_{e_i}, <e_i, r_n, e_m> \in T_s \cup T_t} \exp(\boldsymbol{v}^\top \boldsymbol{h}_{r_n}^{\rightarrow e_m})}$$

b. Intra- & Cross-Graph Representation Learning.

$$\boldsymbol{h}_{e_i} = [\boldsymbol{h}_{e_i}^0 \| \boldsymbol{h}_{e_i}^1 \| ... \| \boldsymbol{h}_{e_i}^l] \quad \text{and} \quad \boldsymbol{h}_{e_i}^{proxy} = \sum_{\boldsymbol{q}_j \in S_p} \frac{\exp(\mathrm{sim}(\boldsymbol{h}_{e_i}, \boldsymbol{q}_j))}{\sum_{\boldsymbol{q}_k \in S_p} \exp(\mathrm{sim}(\boldsymbol{h}_{e_i}, \boldsymbol{q}_k))} (\boldsymbol{h}_{e_i} - \boldsymbol{q}_j).$$

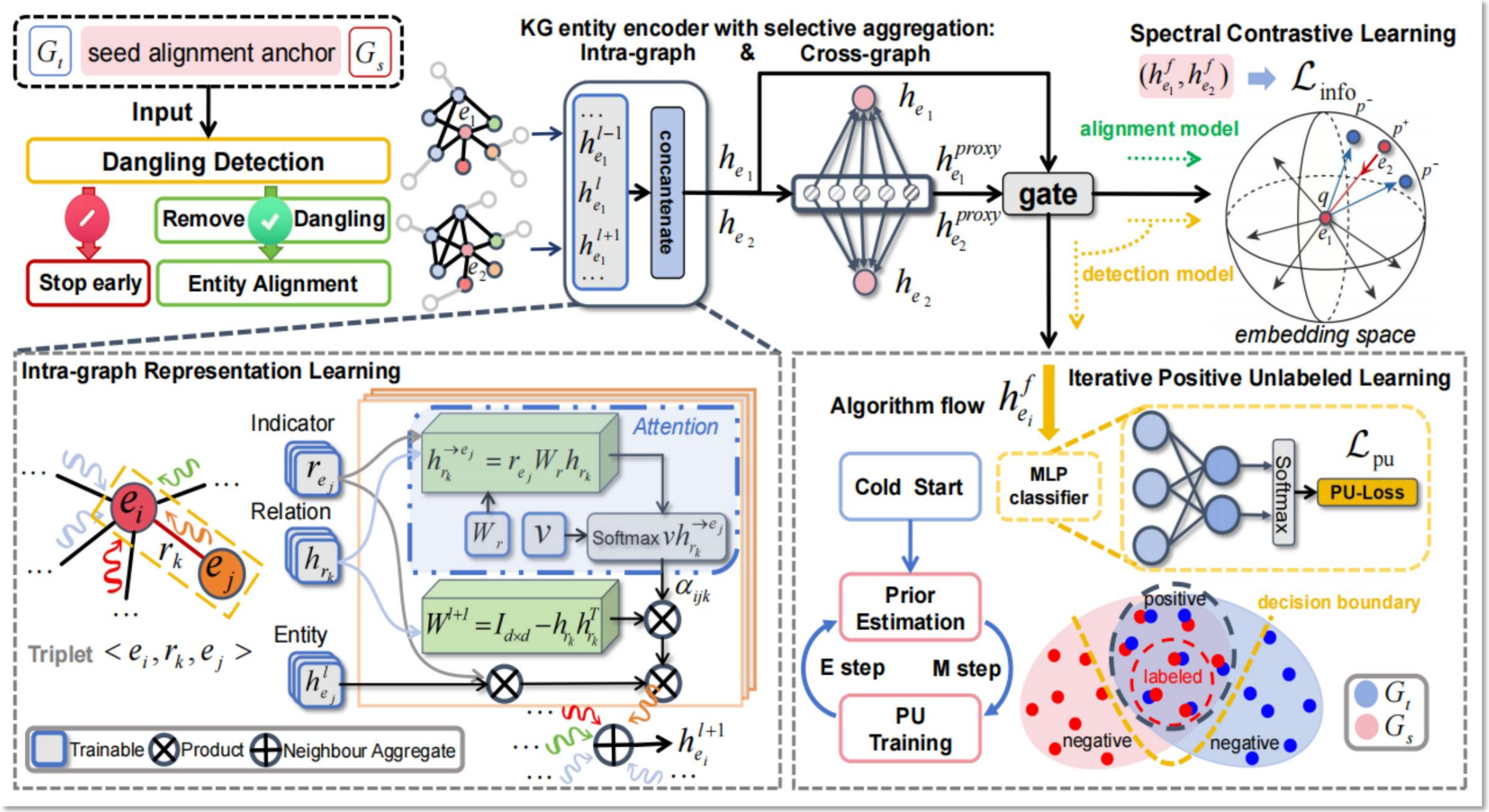
Final Embeddings:

encoded by one shared KEESA with below spectral contrastive learning

$$\boldsymbol{\theta}_{e_i} = \mathrm{sigmoid}(\boldsymbol{W}_g \boldsymbol{h}_{e_i}^{proxy} + \boldsymbol{b}), \qquad \boldsymbol{h}_{e_i}^{f} = [(\boldsymbol{\theta}_{e_i} \cdot \boldsymbol{h}_{e_i} + (1 - \boldsymbol{\theta}_{e_i}) \cdot \boldsymbol{h}_{e_i}^{proxy}) \| r_{e_i}],$$

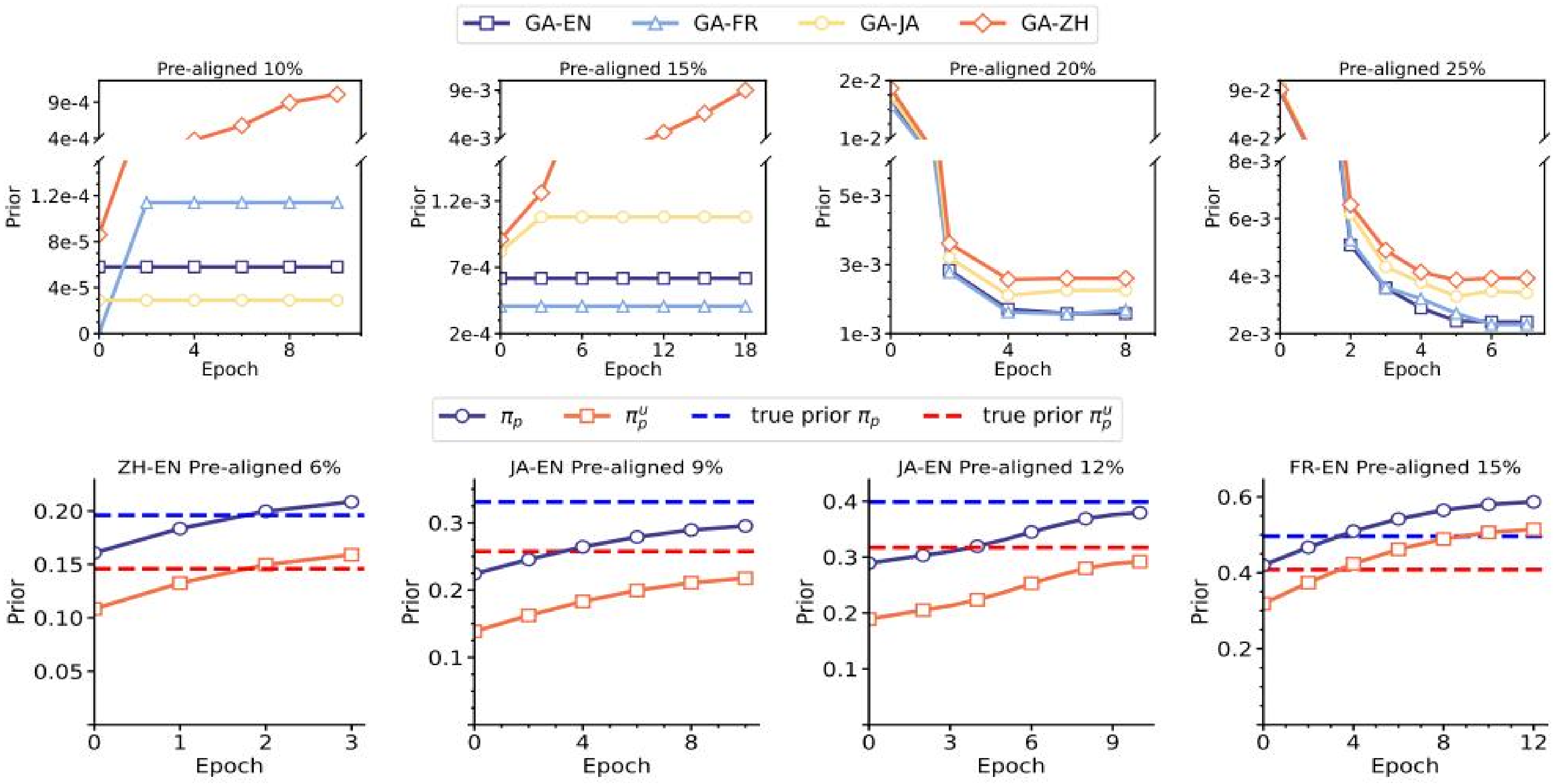
(Comparison with Dual-AMN in our appendix)

The Framework contains all above modules



Experiments:

1. Prior Estimation



2. Dangling-Unaware Comparison

Method	GA16K H@1	GA16K H@10	GA16K H@50
BootEA	13.95	37.25	49.08
TransEdge	0.03	0.12	0.14
MRAEA	63.97	76.64	81.06
GCN-Align	29.48	45.64	57.15
RSNs	9.40	42.70	46.70
MuGNN	62.17	76.25	80.87
KECG	44.18	57.73	63.41
AliNet	48.53	67.72	74.50
Dual-AMN	64.49	**80.55**	**84.67**
Ours	**67.59**	80.33	84.35

Table 2: Performance comparison with dangling-entities-unaware baselines on GA16K.

Experiments:

3. Dangling-Aware:

3.1 Dangling Detection

Methods		ZH-EN			EN-ZH			JA-EN			EN-JA			FR-EN			EN-FR		
		Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
AliNet	NNC	.676	.419	.517	.738	.558	.634	.597	.482	.534	.761	.120	.207	.466	.365	.409	.545	.162	.250
	MR	.752	.538	.627	.828	.505	.627	.779	.580	.665	.854	.543	.664	.552	.570	.561	.686	.549	.609
	BR	.762	.556	.643	.829	.515	.635	.783	.591	.673	.846	.546	.663	.547	.556	.552	.674	.556	.609
MTransE	NNC	.604	.485	.538	.719	.511	.598	.622	.491	.549	.686	.506	.583	.459	.447	.453	.557	.543	.550
	MR	.781	.702	.740	.866	.675	.759	.799	.708	.751	.864	.653	.744	.482	.575	.524	.639	.613	.625
	BR	**.811**	.728	.767	**.892**	.700	.785	**.816**	.733	.772	**.888**	.731	.801	.539	.686	.604	.692	.735	.713
Ours		.763	**.925**	**.836**	.844	**.909**	**.875**	.807	**.836**	**.821**	.880	**.809**	**.843**	**.615**	**.772**	**.685**	**.732**	**.749**	**.740**

Table 3: Dangling detection results on DBP2.0 in the consolidated setting.

3.2 Entity Alignment

Methods		ZH-EN			EN-ZH			JA-EN			EN-JA			FR-EN			EN-FR		
		Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
AliNet	NNC	.121	.193	.149	.085	.138	.105	.113	.146	.127	.067	.208	.101	.126	.148	.136	.086	.161	.112
	MR	.207	.299	.245	.159	.320	.213	.231	.321	.269	.178	.340	.234	.195	.190	.193	.160	.200	.178
	BR	.203	.286	.238	.155	.308	.207	.223	.306	.258	.170	.321	.222	.183	.181	.182	.164	.200	.180
MTransE	NNC	.164	.215	.186	.118	.207	.150	.180	.238	.205	.101	.167	.125	.185	.189	.187	.135	.140	.138
	MR	.302	.349	.324	.231	362	.282	.313	.367	.338	.227	.366	.280	.260	.220	.238	.213	.224	.218
	BR	**.312**	.362	.335	**.241**	.376	.294	.314	.363	.336	**.251**	.358	.295	**.265**	.208	.233	**.231**	.213	.222
Ours		.279	**.447**	**.344**	.219	**.489**	**.303**	**.324**	**.409**	**.362**	.234	**.460**	**.310**	.234	**.320**	**.271**	.192	**.363**	**.251**

Table 4: Entity alignment results on DBP2.0 in the consolidated setting.

4. Ablation Study:

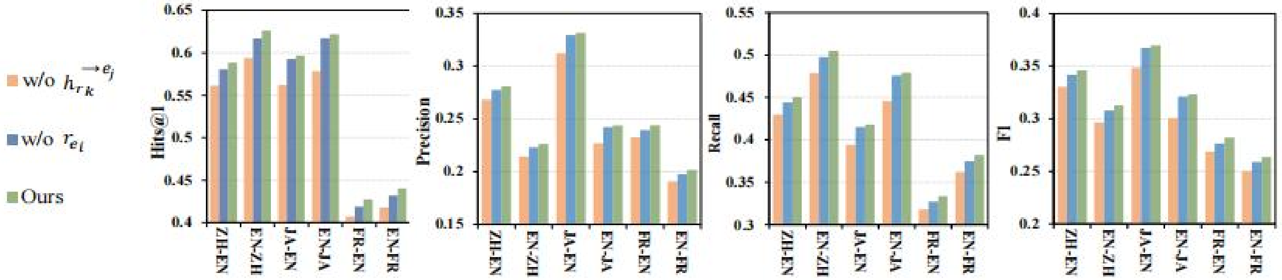


Figure 5: The ablation study of entity alignment performance in the consolidated setting on DBP2.0.

5. Convergence

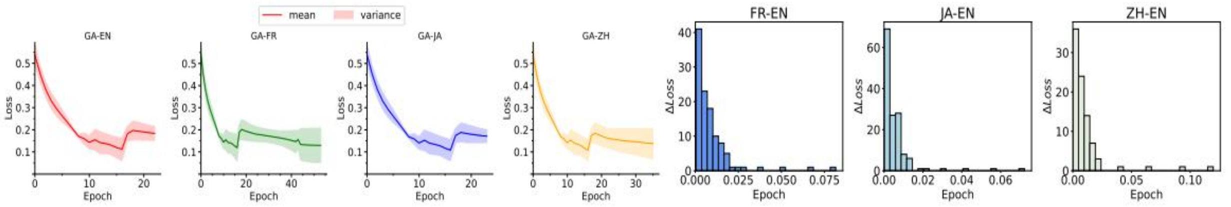


Figure 7: Visualization of loss convergence on DBP2.0 and GA-DBP15K.

6. Efficiency

Datasets	Triples	Inference Time	Average Training Time (from 1 to 45 training epochs)						CPU Memory	GPU Memory
			1-20	21-25	26-30	31-35	36-40	41-45		
DBP2.0$_{ZH-EN}$	872,935	48.78s	11.21s/it	21.16s/it	25.67s/it	28.17s/it	29.21s/it	30.14s/it	10.8GB	32.5GB
DBP2.0$_{JA-EN}$	1,015,545	120.76s	28.14s/it	53.99s/it	63.43s/it	68.27s/it	70.61s/it	72.80s/it	11.9GB	32.6GB
DBP2.0$_{FR-EN}$	2,089,909	382.48s	90.18s/it	158.18s/it	190.65s/it	-	-	-	27.7GB	60.2GB

上海交通大学
SHANGHAI JIAO TONG UNIVERSITY

Thanks!

Arxiv links | Code links | Wechat