SimKGC: Simple Contrastive Knowledge Graph Completion with Pre-trained Language Models

Liang Wang, Wei Zhao, Zhuoyu Wei and Jingming Liu

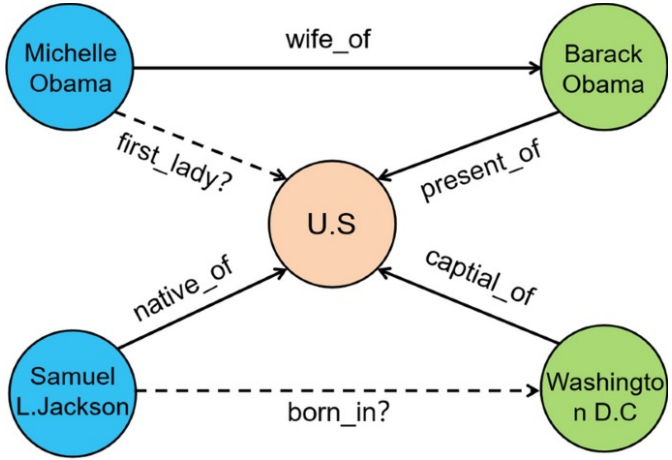
ACL 2022

2025-01-23
HoonUi Lee

- **Knowledge Graph Completion**
- **Text-based KGC**
  - vs. Embedding based KGC
  - Pros & Cons
- **SimKGC**
  - Bi-Encoder
  - Negative sampling
  - Re-rank
  - Training and inference
- **Experiment**
  - dataset & metrics
  - results
- **Ablation study**
- **Conclusion**

Knowledge Graph Completion

- **Why KGC is needed**
  - Collect data from web when constructing KG
  - Facts missing on the web can't be collected
    - or errors occur during extraction process
  - To reason over known facts and infer the missing links



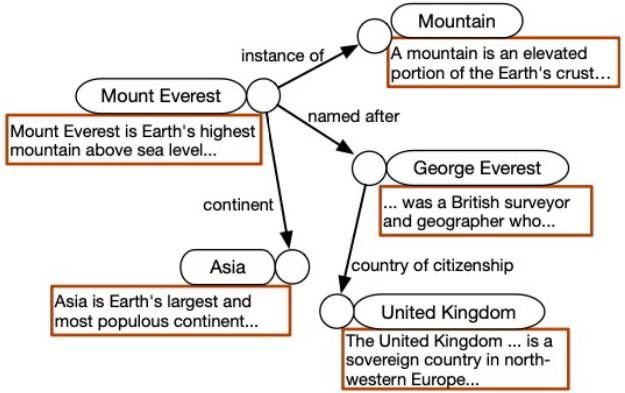
- **Two categories of KGC method**
  - Embedding-based KGC
  - Text-based KGC

Text-based KGC

- VS. Embedding-based KGC

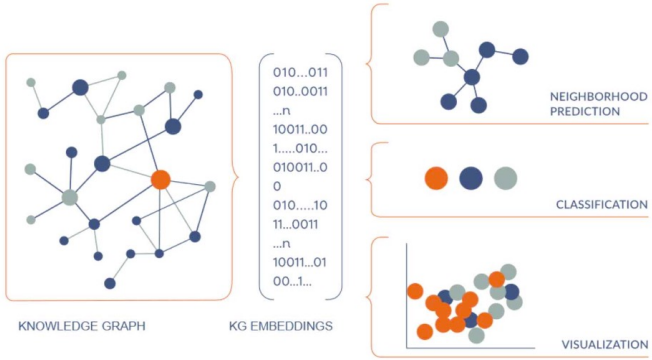
❖ Text-based KGC

- ❑ Use provided texts to construct entity representations
  - ▸ stored in the form of dictionary in this model



❖ Embedding-based KGC

- ❑ Map each entity and relation into a low-dimensional vector without using any side information
  - ▸ no use entity descriptions
- ❑ Perform completion through operations on the embeddings of entities



Text-based KGC

- VS. Embedding-based KGC

❖ Lower than expected performance

- ❑ Text-based methods were expected to outperform embedding-based methods due to utilizing more inputs, but they did not
- ❑ Because of **high computational cost, large negative sample size**
  - ▸ RotatE (emb. based) trains 1000 epochs with negative sample size of 64
  - ▸ KEPLER (text based) trains 30 epochs with negative sample size of 1
- ❑ Hypothesizing the key issue is the inefficiencies in contrastive learning, proposed work aim to improve by using **bi-encoder and 3 types of negatives**
  - ▸ **Bi-Encoder** allow learning more samples
  - ▸ **3 types of negatives** allow learning diverse samples

Text-based KGC

- Pros & Cons

❖ Pros

- ❑ Inductive entity representation learning
  - ‣ inference is possible for entities not present in the training data
  - ‣ generate embedding vector by **entity's descriptions** with a **pre-trained language model**

❖ Cons

- ❑ Heavily rely on the semantic match and ignore topological bias
  - ‣ two entities are more likely to related if they are connected by a short path in graph
  - ‣ text-based tends to not reflect this

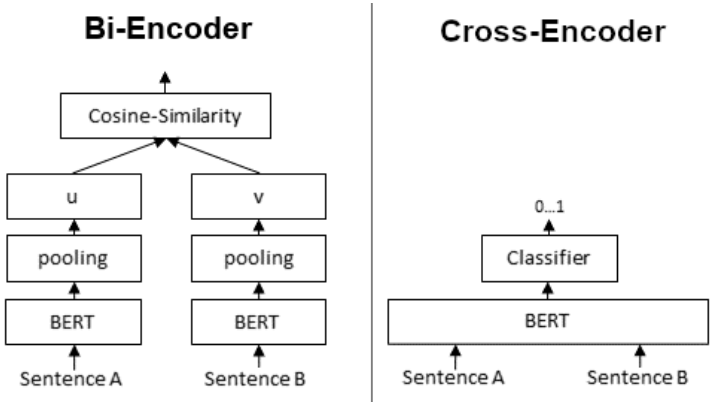
➔ To mitigate this, **simple re-ranking** is proposed

SimKGC

- Bi-Encoder

❖ Bi-Encoder architecture

- ❑ Use two encoders
  - ▸ Each encoder generates embeddings for the sentences
  - ▸ Computes cosine similarity between the generated embeddings
  - ▸ Fast computation speed



❖ Cross-Encoder architecture

- ❑ Uses a single encoder
  - ▸ Generates embeddings by inputting both sentences simultaneously
  - ▸ Analyzes relationships between sentences without transforming them
  - ▸ High accuracy

SimKGC

- Bi-Encoder

❖ Two BERT encoder

❑ $BERT_{hr}$

- get relation-aware embedding e_{hr}
- text descriptions of h (head) + [SEP] + text descriptions of r (relation)
- mean pooling + L_2 normalization

❑ $BERT_t$

- get tail entity embedding e_t
- only text description of t
- mean pooling + L_2 normalization

SimKGC

- Bi-Encoder

❖ Scoring function

- ❑ Using cosine similarity
  - e_{hr} and e_t are both L_2 normalized

$$\cos(\mathbf{e}_{hr}, \mathbf{e}_t) = \frac{\mathbf{e}_{hr} \cdot \mathbf{e}_t}{\|\mathbf{e}_{hr}\| \|\mathbf{e}_t\|} = \mathbf{e}_{hr} \cdot \mathbf{e}_t$$

- ❑ Compute cosine similarity between e_{hr} and all entities in $\mathcal{E}$ for tail entity prediction $(h, r, ?)$

$$\underset{t_i}{\operatorname{argmax}} \cos(\mathbf{e}_{hr}, \mathbf{e}_{t_i}),\ t_i \in \mathcal{E}$$

- ❑ Predict the one with the largest score for the answer

SimKGC

- Negative sampling

❖ Combine 3 types of negatives

❑ Most existing methods randomly corrupt h or t
then filter out false negatives that appear in the training graph G

- typical number of negatives are
  ~64 for embedding-based methods / ~5 for text-based methods

❑ Use three types of negatives to improve the training efficiency
without incurring significant computational and memory overhead

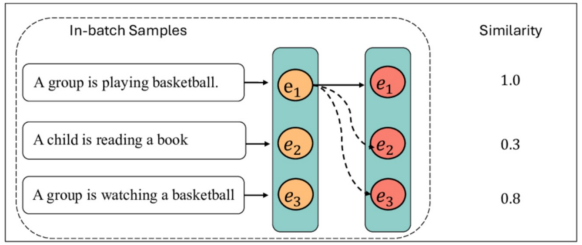
- In-batch Negatives
- Pre-batch Negatives
- Self-Negatives

SimKGC

- Negative sampling: In-batch

❖ In-batch Negatives (IB)

- ❑ Widely adopted strategy in visual representation learning
- ❑ Use Entities within the same batch as the negatives
- ❑ Efficient reuse of entity embeddings for bi-encoder models
  - change e_t when calculate $\cos(e_{hr}, e_t)$
  - by using bi-encoder, the embedding of tail is separated from e_{hr}
  - while cross-encoder has to calculate all combination of (h, r, t')
- ❑ Negative samples are generated equal to the batch size



SimKGC

- Negative sampling: Pre-batch

❖ Pre-batch Negatives (PB)

- ❑ Use entity embeddings from previous batches
  - ▸ the embedding of tail
- ❑ Since these embeddings are computed with an earlier version of model parameters, they are not consistent with in-batch negatives
  - ▸ the consistency problem can occur
- ❑ To minimize this issues, only 1-2 pre-batches are used

SimKGC

- Negative sampling: Self

❖ Self-Negatives (SN)

❑ To mining hard negatives

- that is very close to the Positive Sample (correct answer)
- Although it is harder for the model to learn, it is crucial for enhancing discrimination ability

❑ Text-based methods tend to assign a high score
to the head entity h, because of the high text overlap

- to mitigate this issue, use the head entity as hard negatives
- (h, r, h)
- can make the model rely less on the meaningless text match

SimKGC

- Negative sampling

❖ Filtering out false negatives

- ❑ There may exist some false negatives
  - ▸ the correct entity happens to appear in another triple within the same batch
- ❑ filter out such entities with a binary mask
- ❑ $\{t' | t' \in \mathcal{N}_{\mathrm{IB}} \cup \mathcal{N}_{\mathrm{PB}} \cup \mathcal{N}_{\mathrm{SN}}, (h, r, t') \notin \mathcal{G}\}$
  - ▸ false negatives that do not appear in the training data will not be filtered

SimKGC

- Negative sampling

Bi-Encoder ➔ By separating e_{hr}, e_t , allows learning more samples

Negative sampling

➔ By reusing entities, there is no need for large extra storage space for negative samples

➔ Easy-to-create hard negatives improve learning efficiency

SimKGC

- Re-rank

❖ Graph-based Re-ranking

- ❑ Nearby entities are more likely to be related than entities that are far apart
- ❑ Text-based KGC methods are good at capturing semantic relatedness but may not fully capture such bias

- ❑ simple graph-based re-ranking strategy
  - ▸ increase the score of candidate tail entity t_i
  - ▸ t_i is in k-hop neighbors $\mathcal{E}_k(h)$ of the head entity h

- ❑ $$\underset{t_i}{\operatorname{argmax}} \cos(\mathbf{e}_{hr}, \mathbf{e}_{t_i}) + \alpha \mathbb{1}(t_i \in \mathcal{E}_k(h))$$

SimKGC

- Training and Inference

❖ InfoNCE loss (Information Noise-Contrastive Estimation loss)

❑ $$\mathcal{L} = -\log \frac{e^{(\phi(h,r,t)-\gamma)/\tau}}{e^{(\phi(h,r,t)-\gamma)/\tau} + \sum_{i=1}^{|\mathcal{N}|} e^{\phi(h,r,t_i')/\tau}}$$

- To increase the score of correct triple and help distinguish it from negative samples
- γ for encouraging the model to have greater confidence in the correct triple
- τ for adjusting the relative importance of negative samples
  - ✓ re-parameterizing $\log 1/\tau$ as a learnable parameter
  - ✓ gradually adjust the importance of Negative Samples during training

❖ Inference time

❑ BERT forward pass computation + ranking score for all entities

❑ $|\mathcal{E}| + |T|$

- number of entities + number of test triples

Experiment

- dataset & metrics

Dataset

dataset	#entity	#relation	#train	#valid	#test
WN18RR	40,943	11	86,835	3034	3134
FB15k-237	14,541	237	272,115	17,535	20,466
Wikidata5M-Trans	4,594,485	822	20,614,279	5,163	5,163
Wikidata5M-Ind	4,579,609	822	20,496,514	6,699	6,894

Table 1: Statistics of the datasets used in this paper. "Wikidata5M-Trans" and "Wikidata5M-Ind" refer to the transductive and inductive settings, respectively.

Metrics

- **MRR**: The average of the reciprocal ranks of the correct answers inferenced by the model
- **Hits@k:** Evaluate whether the answer is included in the top k rankings that model inferenced
  - with filtered setting
  - averaging head entity prediction and tail entity prediction

Experiment

- Wikidata5M

❖ Result

Method	Wikidata5M-Trans				Wikidata5M-Ind			
	MRR	H@1	H@3	H@10	MRR	H@1	H@3	H@10
embedding-based methods								
TransE (Bordes et al., 2013)	25.3	17.0	31.1	39.2	-	-	-	-
RotatE (Sun et al., 2019b)	29.0	23.4	32.2	39.0	-	-	-	-
text-based methods								
DKRL (Xie et al., 2016)	16.0	12.0	18.1	22.9	23.1	5.9	32.0	54.6
KEPLER (Wang et al., 2021c)	21.0	17.3	22.4	27.7	40.2	22.2	51.4	73.0
BLP-ComplEx (Daza et al., 2021)	-	-	-	-	48.9	26.2	66.4	87.7
BLP-SimplE (Daza et al., 2021)	-	-	-	-	49.3	28.9	63.9	86.6
SimKGC$_{\text{IB}}$	35.3	30.1	37.4	**44.8**	60.3	39.5	77.8	**92.3**
SimKGC$_{\text{IB+PB}}$	35.4	30.2	37.3	**44.8**	60.2	39.4	77.7	**92.4**
SimKGC$_{\text{IB+SN}}$	35.6	31.0	37.3	43.9	71.3	60.7	**78.7**	91.3
SimKGC$_{\text{IB+PB+SN}}$	**35.8**	**31.3**	**37.6**	44.1	**71.4**	**60.9**	78.5	91.7

- Metrics for inductive setting are much higher
  - inductive setting ranks 7,475 entities in the test set
    while transductive setting ranks ~ 4.6 million entities
- Combining all three types of negatives not always has the best results

Experiment

- WN18RR & FB15K-237

❖ Result

Method	WN18RR				FB15k-237			
	MRR	H@1	H@3	H@10	MRR	H@1	H@3	H@10
embedding-based methods								
TransE (Bordes et al., 2013)†	24.3	4.3	44.1	53.2	27.9	19.8	37.6	44.1
DistMult (Yang et al., 2015)†	44.4	41.2	47.0	50.4	28.1	19.9	30.1	44.6
RotatE (Sun et al., 2019b)†	47.6	42.8	49.2	57.1	33.8	24.1	37.5	53.3
TuckER (Balazevic et al., 2019)†	47.0	44.3	48.2	52.6	**35.8**	**26.6**	**39.4**	**54.4**
text-based methods								
KG-BERT (Yao et al., 2019)	21.6	4.1	30.2	52.4	-	-	-	42.0
MTL-KGC (Kim et al., 2020)	33.1	20.3	38.3	59.7	26.7	17.2	29.8	45.8
StAR (Wang et al., 2021a)	40.1	24.3	49.1	70.9	29.6	20.5	32.2	48.2
SimKGC$_{\text{IB}}$	**67.1**	58.5	**73.1**	**81.7**	33.3	24.6	36.2	51.0
SimKGC$_{\text{IB+PB}}$	66.6	57.8	72.3	**81.7**	33.4	24.6	**36.5**	**51.1**
SimKGC$_{\text{IB+SN}}$	66.7	**58.8**	72.1	80.5	33.4	24.7	36.3	50.9
SimKGC$_{\text{IB+PB+SN}}$	66.6	58.7	71.7	80.0	**33.6**	**24.9**	36.2	**51.1**

- FB15k-237 dataset is much denser
  - contains fewer entities
  - generalizable inference rules more important than textual relatedness
- It is possible to ensemble with embedding-based methods

Ablation study

- using more negatives & InfoNCE loss

❖ Result

loss	# of neg	MRR	H@1	H@3	H@10
InfoNCE	255	**64.4**	**53.8**	**71.7**	**82.8**
InfoNCE	5	48.8	31.9	60.2	80.3
margin	255	39.5	28.5	44.4	61.2
margin	5	38.0	27.5	42.8	58.7

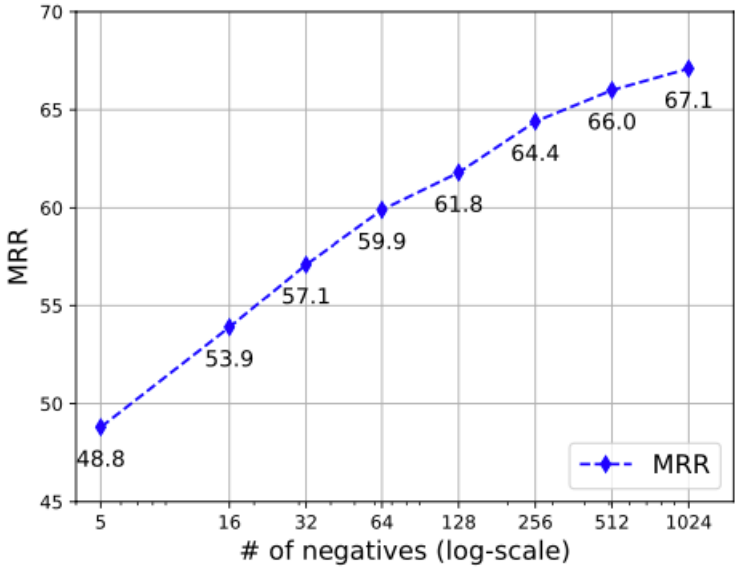
- 255 negatives versus 5 negatives
  - with batch size 256
- InfoNCE versus following margin loss
  - average of the difference from the correct answer score

$$\frac{1}{|\mathcal{N}|}\sum_{i=1}^{|\mathcal{N}|}\max(0, \lambda + \phi(h,r,t'_i) - \phi(h,r,t))$$

Ablation study

- # of negatives

❖ **Result**



❑ More negatives, better performance

Ablation study

- Re-ranking

❖ Result

	MRR	H@1	H@3	H@10
w/ re-rank	**35.8**	**31.3**	**37.6**	**44.1**
w/o re-rank	35.5	31.0	37.3	43.9

Table 6: Ablation of re-ranking on the Wikidata5M-Trans dataset.

- ❑ Slight but stable increase

- ❑ Re-ranking strategy does not apply to inductive KGC
  since entities in the test set never appear in the training data

Ablation study

- “n” side predicting

❖ Result

Dataset	1-1	1-n	n-1	n-n
Wikidata5M-Trans	30.4	8.3	71.1	10.6
Wikidata5M-Ind	83.5	71.1	80.0	54.7

Table 8: MRR for different kinds of relations on the Wikidata5M dataset with $\text{SimKGC}_{\text{IB+PB+SN}}$.

- Predicting the n-side is more difficult
  - classify “possible answer” and “hard negative”
  - incompleteness of the knowledge graph

Ablation study

- "n" side predicting

❖ Examples

triple	(Rest Plaus Historic District, is located in, **New York**)
evidence	. . . a national historic district located at Marbletown in Ulster County, New York. . .
SimKGC	Marbletown
triple	(**Timothy P. Green**, place of birth, St. Louis)
evidence	William Douglas Guthrie (born January 17, 1967 in St. Louis, MO) is a professional boxer. . .
SimKGC	William Douglas Guthrie
triple	(TLS termination proxy, instance of, **networked software**)
evidence	. . . a proxy server that is used by an institution to handle incoming TLS connections. . .
SimKGC	http server

- 1st ex : both correct answers
- 2nd ex : lots of cases (place of birth)
- 3rd ex: fail to classify hard negative

➔ Limitation of text-based KGC

Ablation study

- Human evaluation

❖ Result

	correct	wrong	unknown
$(h, r, ?)$	24%	54%	22%
$(?, r, t)$	74%	14%	12%
Avg	49%	34%	17%

- ❑ Randomly sample 100 wrong predictions according to H@1
  - ▸ Whether it is incorrect because it does not exist in the graph or truly a wrong answer
  - ▸ “unknown” means annotators are unable to decide based on textual information

- ❑ The possibility that the current metric may underestimate the model’s performance

Ablation study

- Entity visualization

❖ **Result**

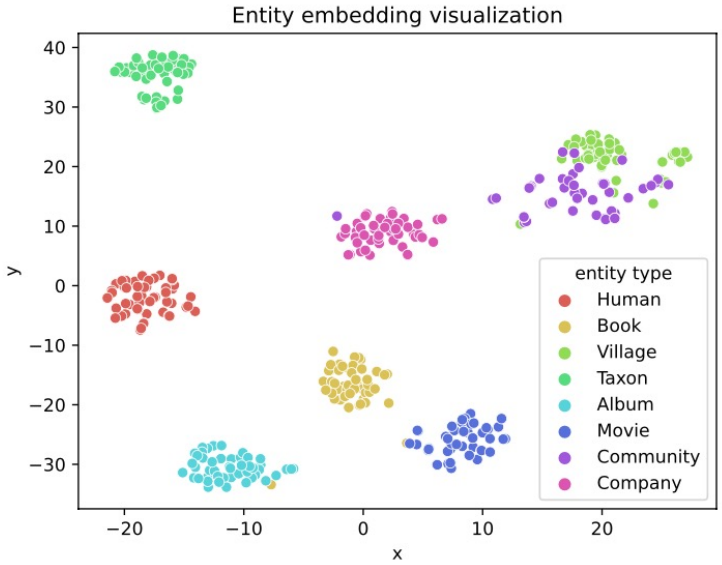


Figure 3: 2-D visualization of the entity embeddings from the Wikidata5M-Trans dataset with t-SNE (Maaten and Hinton, 2008).

Conclusion

- **Previous work**
  - Existing model tends to fall behind embedding-based KGC due to reduced training efficiency
- **SimKGC**
  - Bi-encoder and 3 types of negatives allowed the model to learn more and diverse samples
- **Experiment**
  - Improvement is needed for transductive KGC on large-sized graphs and KGC on sparse graphs
- **Ablation study**
  - InfoNCE & # of negatives are important factors + re-ranking
  - limitation of n-side prediction
  - The need for a new metric