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

Liang Wang[1]* and **Wei Zhao**[2] and **Zhuoyu Wei**[2] and **Jingming Liu**[2]
[1]Microsoft Research Asia
[2]Yuanfudao AI Lab, Beijing, China
wangliang@microsoft.com
{zhaowei01,weizhuoyu,liujm}@yuanfudao.com

arXiv:2203.02167v1 [cs.CL] 4 Mar 2022

Abstract

Knowledge graph completion (KGC) aims to reason over known facts and infer the missing links. Text-based methods such as KG-BERT (Yao et al., 2019) learn entity representations from natural language descriptions, and have the potential for inductive KGC. However, the performance of text-based methods still largely lag behind graph embedding-based methods like TransE (Bordes et al., 2013) and RotatE (Sun et al., 2019b). In this paper, we identify that the key issue is efficient contrastive learning. To improve the learning efficiency, we introduce three types of negatives: in-batch negatives, pre-batch negatives, and self-negatives which act as a simple form of hard negatives. Combined with InfoNCE loss, our proposed model SimKGC can substantially outperform embedding-based methods on several benchmark datasets. In terms of mean reciprocal rank (MRR), we advance the state-of-the-art by +19% on WN18RR, +6.8% on the Wikidata5M transductive setting, and +22% on the Wikidata5M inductive setting. Thorough analyses are conducted to gain insights into each component. Our code is available at https://github.com/intfloat/SimKGC.

1 Introduction

Large-scale knowledge graphs (KGs) are important components for knowledge-intensive applications, such as question answering (Sun et al., 2019a), recommender systems (Huang et al., 2018), and intelligent conversational agents (Dinan et al., 2019) etc. KGs usually consist of a set of triples (h, r, t), where h is the head entity, r is the relation, and t is the tail entity. Popular public KGs include Freebase (Bollacker et al., 2008), Wikidata (Vrandečić and Krötzsch, 2014), YAGO (Suchanek et al., 2007), ConceptNet (Speer et al., 2017), and WordNet (Miller, 1992) etc. Despite their usefulness in practice, they are often incomplete. Knowledge graph completion (KGC) techniques are necessary for the automatic construction and verification of knowledge graphs.

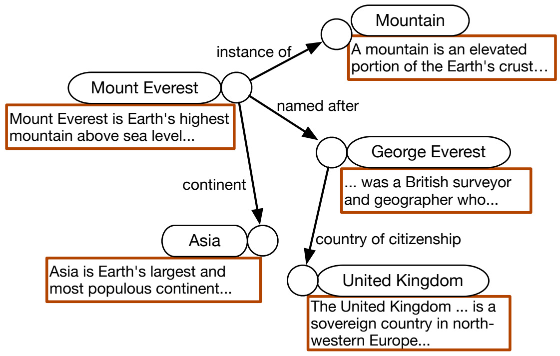

Figure 1: An example of knowledge graph. Each entity has its name and textual descriptions.

Existing KGC methods can be categorized into two families: embedding-based and text-based methods. Embedding-based methods map each entity and relation into a low-dimensional vector, without using any side information such as entity descriptions. This family includes TransE (Bordes et al., 2013), TransH (Wang et al., 2014), RotatE (Sun et al., 2019b), and TuckER (Balazevic et al., 2019) etc. By comparison, text-based methods (Yao et al., 2019; Xie et al., 2016; Wang et al., 2021c) incorporate available texts for entity representation learning, as shown in Figure 1. Intuitively, text-based methods should outperform embedding-based counterparts since they have access to additional input signals. However, results on popular benchmarks (e.g., WN18RR, FB15k-237, Wikidata5M) tell a different story: text-based methods still lag behind even with pre-trained language models.

We hypothesize that the key issue for such performance degradation is the inefficiency in contrastive learning. Embedding-based methods do not involve the expensive computation of text en-

*Work done while at Yuanfudao AI Lab.

coders and thus can be extremely efficient to train with a large negative sample size. For example, the default configuration of RotatE [1] trains 1000 epochs with a negative sample size of 64 on the Wikidata5M dataset. While the text-based method KEPLER (Wang et al., 2021c) can only train 30 epochs with a negative sample size of 1 due to the high computational cost incurred by RoBERTa.

In this paper, inspired by the recent progress on contrastive learning, we introduce three types of negatives to improve the text-based KGC method: in-batch negatives, pre-batch negatives, and self-negatives. By adopting bi-encoder instead of cross-encoder (Yao et al., 2019) architecture, the number of in-batch negatives can be increased by using a larger batch size. Vectors from previous batches are cached and act as pre-batch negatives (Karpukhin et al., 2020). Additionally, mining hard negatives can be beneficial for improving contrastive learning. We find that the head entity itself can serve as hard negatives, which we call "self-negatives". As a result, the negative sample size can be increased to the scale of thousands. We also propose to change the loss function from margin-based ranking loss to InfoNCE, which can make the model focus on hard negatives.

One advantage of text-based methods is that they enable inductive entity representation learning. Entities that are not seen during training can still be appropriately modeled, while embedding-based methods like TransE can only reason under the transductive setting [2]. Inductive knowledge graph completion is important in the real world as new entities are coming out every day. Moreover, text-based methods can leverage state-of-the-art pre-trained language models to learn better representations. A line of recent work (Shin et al., 2020; Petroni et al., 2019) attempts to elicit the implicitly stored knowledge from BERT. The task of KGC can also be regarded as a way to retrieve such knowledge.

Two entities are more likely to be related if connected by a short path in the graph. Empirically, we find that text-based models heavily rely on the semantic match and ignore such topological bias to some degree. We propose a simple re-ranking strategy by boosting the scores of the head entity's k-hop neighbors.

We evaluate our proposed model SimKGC by conducting experiments on three popular benchmarks: WN18RR, FB15k-237, and Wikidata5M (both transductive and inductive settings). According to the automatic evaluation metrics (MRR, Hits@{1,3,10}), SimKGC outperforms state-of-the-art methods by a large margin on the WN18RR (MRR $47.6 \rightarrow 66.6$), Wikidata5M transductive setting (MRR $29.0 \rightarrow 35.8$), and inductive setting (MRR $49.3 \rightarrow 71.4$). On the FB15k-237 dataset, our results are also competitive. To help better understand our proposed method, we carry out a series of analyses and report human evaluation results. Hopefully, SimKGC will facilitate the future development of better KGC systems.

2 Related Work

Knowledge Graph Completion involves modeling multi-relational data to aid automatic construction of large-scale KGs. In translation-based methods such as TransE (Bordes et al., 2013) and TransH (Wang et al., 2014), a triple (h, r, t) is a relation-specific translation from the head entity h to tail entity t. Complex number embeddings are introduced by Trouillon et al. (2016) to increase the model's expressiveness. RotatE (Sun et al., 2019b) models a triple as relational rotation in complex space. Nickel et al. (2011); Balazevic et al. (2019) treat KGC as a 3-D binary tensor factorization problem and investigate the effectiveness of several factorization techniques. Some methods attempt to incorporate entity descriptions. DKRL (Xie et al., 2016) uses a CNN to encode texts, while KG-BERT (Yao et al., 2019), StAR (Wang et al., 2021a), and BLP (Daza et al., 2021) both adopt pre-trained language models to compute entity embeddings. GraIL (Teru et al., 2020) and BERTRL (Zha et al., 2021) conduct inductive relation prediction by utilizing subgraph or path information. In terms of benchmark performance (Wang et al., 2021c), text-based methods still underperform methods like RotatE.

Pre-trained Language Models including BERT (Devlin et al., 2019), GPT (Radford et al., 2018), and T5 (Raffel et al., 2019) have led to a learning paradigm shift in NLP. Models are first pre-trained on large amounts of unlabeled text corpora with language modeling objectives, and then fine-tuned on downstream tasks. Considering their good performance in few-shot and even zero-shot

[1] https://github.com/DeepGraphLearning/graphvite

[2] All entities in the test set also appear in the training set.

scenarios (Brown et al., 2020), one interesting question is: "Can pre-trained language models be used as knowledge bases?" Petroni et al. (2019) proposed to probe language models with manually designed prompts. A series of following work (Shin et al., 2020; Zhong et al., 2021; Jiang et al., 2020) focus on finding better prompts to elicit the knowledge implicitly stored in the model parameters. Another line of work (Zhang et al., 2019; Liu et al., 2020; Wang et al., 2021c) injects symbolic knowledge into language model pre-training, and shows some performance boost on several knowledge-intensive tasks.

Contrastive Learning learns useful representations by contrasting between positives and negatives (Le-Khac et al., 2020). The definitions of positives and negatives are task-specific. In self-supervised vision representation learning (Chen et al., 2020; He et al., 2020; Grill et al., 2020), a positive pair is two augmented views of the same image, while a negative pair is two augmented views of different images. Recently, contrastive learning paradigm has witnessed great successes in many different fields, including multi-modal pre-training (Radford et al., 2021), video-text retrieval (Liu et al., 2021), and natural language understanding (Gunel et al., 2021) etc. In the NLP community, by leveraging the supervision signals from natural language inference data (Gao et al., 2021), QA pairs (Ni et al., 2021), and parallel corpora (Wang et al., 2021b), these methods have surpassed non-contrastive methods (Reimers and Gurevych, 2019) on semantic similarity benchmarks. Karpukhin et al. (2020); Qu et al. (2021); Xiong et al. (2021) adopt contrastive learning to improve dense passage retrieval for open-domain question answering, where the positive passages are the ones containing the correct answer.

3 Methodology

3.1 Notations

A knowledge graph $\mathcal{G}$ is a directed graph, where the vertices are entities $\mathcal{E}$, and each edge can be represented as a triple (h,r,t), where h, r, and t correspond to head entity, relation, and tail entity, respectively. The link prediction task of KGC is to infer the missing triples given an incomplete $\mathcal{G}$. Under the widely adopted entity ranking evaluation protocol, tail entity prediction (h, r, ?) requires ranking all entities given h and r, similarly for head entity prediction (?, r, t). In this paper, for each triple (h,r,t), we add an inverse triple (t,r^{-1},h), where r^{-1} is the inverse relation of r. Based on such reformulation, we only need to deal with the tail entity prediction problem (Malaviya et al., 2020).

3.2 Model Architecture

Our proposed model SimKGC adopts a bi-encoder architecture. Two encoders are initialized with the same pre-trained language model but do not share parameters.

Given a triple (h,r,t), the first encoder BERT_{hr} is used to compute the relation-aware embedding for the head entity h. We first concatenate the textual descriptions of entity h and relation r with a special symbol [SEP] in between. BERT_{hr} is applied to get the last-layer hidden states. Instead of directly using the hidden state of the first token, we use mean pooling followed by L_2 normalization to get the relation-aware embedding $\mathbf{e}_{hr}$, as mean pooling has been shown to result in better sentence embeddings (Gao et al., 2021; Reimers and Gurevych, 2019). $\mathbf{e}_{hr}$ is relation-aware since different relations will have different inputs and thus have different embeddings, even though the head entity is the same.

Similarly, the second encoder BERT_t is used to compute the L_2-normalized embedding $\mathbf{e}_t$ for the tail entity t. The input for BERT_t only consists of the textual description for entity t.

Since the embeddings $\mathbf{e}_{hr}$ and $\mathbf{e}_t$ are both L_2 normalized, the cosine similarity $\cos(\mathbf{e}_{hr}, \mathbf{e}_t)$ is simply the dot product between two embeddings:

$$\cos(\mathbf{e}_{hr}, \mathbf{e}_t) = \frac{\mathbf{e}_{hr} \cdot \mathbf{e}_t}{\|\mathbf{e}_{hr}\|\|\mathbf{e}_t\|} = \mathbf{e}_{hr} \cdot \mathbf{e}_t \quad (1)$$

For tail entity prediction (h, r, ?), we compute the cosine similarity between $\mathbf{e}_{hr}$ and all entities in $\mathcal{E}$, and predict the one with the largest score:

$$\underset{t_i}{\text{argmax}} \ \cos(\mathbf{e}_{hr}, \mathbf{e}_{t_i}), \ t_i \in \mathcal{E} \quad (2)$$

3.3 Negative Sampling

For knowledge graph completion, the training data only consists of positive triples. Given a positive triple (h, r, t), "negative sampling" needs to sample one or more negative triples to train discriminative models. Most existing methods randomly corrupt h or t and then filter out false negatives that appear in the training graph $\mathcal{G}$. The negatives for different triples are not shared and therefore independent. The typical number of negatives are ~ 64 for embedding-based methods (Sun et al., 2019b), and ~ 5 for text-based methods (Wang et al., 2021a). We combine three types of negatives to improve the training efficiency without incurring significant computational and memory overhead.

In-batch Negatives (IB) This is a widely adopted strategy in visual representation learning (Chen et al., 2020) and dense passage retrieval (Karpukhin et al., 2020) etc. Entities within the same batch can be used as negatives. Such in-batch negatives allow the efficient reuse of entity embeddings for bi-encoder models.

Pre-batch Negatives (PB) The disadvantage of in-batch negatives is that the number of negatives is coupled with batch size. Pre-batch negatives (Lee et al., 2021) use entity embeddings from previous batches. Since these embeddings are computed with an earlier version of model parameters, they are not consistent with in-batch negatives. Usually, only 1 or 2 pre-batches are used. Other methods like MoCo (He et al., 2020) can also provide more negatives. We leave the investigation of MoCo as future work.

Self-Negatives (SN) Besides increasing the number of negatives, mining hard negatives (Gao et al., 2021; Xiong et al., 2021) is also important for improving contrastive representation learning. For tail entity prediction (h, r, ?), text-based methods tend to assign a high score to the head entity h, likely due to the high text overlap. To mitigate this issue, we propose self-negatives that use the head entity h as hard negatives. Including self-negatives can make the model rely less on the spurious text match.

We use $\mathcal{N}_{\text{IB}}$, $\mathcal{N}_{\text{PB}}$, and $\mathcal{N}_{\text{SN}}$ to denote the aforementioned three types of negatives. During training, there may exist some false negatives. For example, the correct entity happens to appear in another triple within the same batch. We filter out such entities with a binary mask [3]. Combining them all, the collection of negatives $\mathcal{N}(h, r)$ is:

$$\{t'|t' \in \mathcal{N}_{\text{IB}} \cup \mathcal{N}_{\text{PB}} \cup \mathcal{N}_{\text{SN}}, (h, r, t') \notin \mathcal{G}\} \quad (3)$$

[3]False negatives that do not appear in the training data will not be filtered.

Assume the batch size is 1024, and 2 pre-batches are used, we would have $|\mathcal{N}_{\text{IB}}| = 1024 - 1$, $|\mathcal{N}_{\text{PB}}| = 2 \times 1024$, $|\mathcal{N}_{\text{SN}}| = 1$, and $|\mathcal{N}(h, r)| = 3072$ negatives in total.

3.4 Graph-based Re-ranking

Knowledge graphs often exhibit spatial locality. 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 inductive bias. We propose a simple graph-based re-ranking strategy: increase the score of candidate tail entity t_i by $\alpha \geq 0$ if t_i is in k-hop neighbors $\mathcal{E}_k(h)$ of the head entity h based on the graph from training set:

$$\underset{t_i}{\text{argmax}} \ \cos(\mathbf{e}_{hr}, \mathbf{e}_{t_i}) + \alpha \mathbb{1}(t_i \in \mathcal{E}_k(h)) \quad (4)$$

3.5 Training and Inference

During training, we use InfoNCE loss with additive margin (Chen et al., 2020; Yang et al., 2019):

$$\mathcal{L} = -\log \frac{e^{(\phi(h,r,t)-\gamma)/\tau}}{e^{(\phi(h,r,t)-\gamma)/\tau} + \sum_{i=1}^{|\mathcal{N}|} e^{\phi(h,r,t'_i)/\tau}} \quad (5)$$

The additive margin $\gamma > 0$ encourages the model to increase the score of the correct triple (h,r,t). $\phi(h, r, t)$ is the score function for a candidate triple, here we define $\phi(h, r, t) = \cos(\mathbf{e}_{hr}, \mathbf{e}_t) \in [-1, 1]$ as in Equation 1. The temperature τ can adjust the relative importance of negatives, smaller τ makes the loss put more emphasis on hard negatives, but also risks over-fitting label noise. To avoid tuning τ as a hyperparameter, we re-parameterize $\log \frac{1}{\tau}$ as a learnable parameter.

For inference, the most time-consuming part is $O(|\mathcal{E}|)$ BERT forward pass computation of entity embeddings. Assume there are $|\mathcal{T}|$ test triples. For each triple (h, r, ?) and (t, r^{-1}, ?), we need to compute the relation-aware head entity embedding and use a dot product to get the ranking score for all entities. In total, SimKGC needs $|\mathcal{E}| + 2 \times |\mathcal{T}|$ BERT forward passes, while cross-encoder models like KG-BERT (Yao et al., 2019) needs $|\mathcal{E}| \times 2 \times |\mathcal{T}|$. Being able to scale to large datasets is important for practical usage. For bi-encoder models, we can precompute the entity embeddings and retrieve top-k entities efficiently with the help of fast similarity search tools like Faiss (Johnson et al., 2021).

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.

4 Experiments

4.1 Experimental Setup

Datasets We use three datasets for evaluation: WN18RR, FB15k-237, and Wikidata5M (Wang et al., 2021c). The statistics are shown in Table 1. Bordes et al. (2013) proposed the WN18 and FB15k datasets. Later work (Toutanova et al., 2015; Dettmers et al., 2018) showed that these two datasets suffer from test set leakage and released WN18RR and FB15k-237 datasets by removing the inverse relations. The WN18RR dataset consists of $\sim 41k$ synsets and 11 relations from WordNet (Miller, 1992), and the FB15k-237 dataset consists of $\sim 15k$ entities and 237 relations from Freebase. The Wikidata5M dataset is much larger in scale with ~ 5 million entities and ~ 20 million triples. It provides two settings: transductive and inductive. For the transductive setting, all entities in the test set also appear in the training set, while for the inductive setting, there is no entity overlap between train and test set. We use "Wikidata5M-Trans" and "Wikidata5M-Ind" to indicate these two settings.

For textual descriptions, we use the data provided by KG-BERT (Yao et al., 2019) for WN18RR and FB15k-237 datasets. The Wikidata5M dataset already contains descriptions for all entities and relations.

Evaluation Metrics Following previous work, our proposed KGC model is evaluated with entity ranking task: for each test triple (h, r, t), tail entity prediction ranks all entities to predict t given h and r, similarly for head entity prediction. We use four automatic evaluation metrics: mean reciprocal rank (MRR), and Hits@k($k \in$\{1,3,10\}) (H@k for short). MRR is the average reciprocal rank of all test triples. H@k calculates the proportion of correct entities ranked among the top-k. MRR and H@k are reported under the *filtered setting* (Bordes et al., 2013), The *filtered setting* ignores the scores of all known true triples in the training, validation, and test set. All metrics are computed by averaging over two directions: head entity prediction and tail entity prediction.

We also conduct a human evaluation on the Wikidata5M dataset to provide a more accurate estimate of the model's performance.

Hyperparameters The encoders are initialized with *bert-base-uncased* (English). Using better pre-trained language models is expected to improve performance further. Most hyperparameters except learning rate and training epochs are shared across all datasets to avoid dataset-specific tuning. We conduct grid search on learning rate with ranges $\{10^{-5}, 3\times10^{-5}, 5\times10^{-5}\}$. Entity descriptions are truncated to a maximum of 50 tokens. Temperature τ is initialized to 0.05, and the additive margin for InfoNCE loss is 0.02. For re-ranking, we set $\alpha = 0.05$. 2 pre-batches are used with logit weight 0.5. We use AdamW optimizer with linear learning rate decay. Models are trained with batch size 1024 on 4 V100 GPUs. For the WN18RR, FB15k-237, and Wikidata5M (both settings) datasets, we train for 50, 10, and 1 epochs, respectively. Please see Appendix A for more details.

4.2 Main Results

We reuse the numbers reported by Wang et al. (2021c) for TransE and DKRL, and the results for RotatE are from the official GraphVite [4] benchmark. In Table 2 and 3, our proposed model SimKGC$_{\text{IB+PB+SN}}$ outperforms state-of-the-art methods by a large margin on the WN18RR, Wikidata5M-Trans, and Wikidata5M-Ind datasets, but slightly lags behind on the FB15k-237 dataset (MRR 33.6% vs 35.8%). To the best of our knowledge, SimKGC is the first text-based KGC method that achieves better results than embedding-based counterparts.

[4] https://graphvite.io/docs/latest/benchmark

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
$\text{SimKGC}_{\text{IB}}$	35.3	30.1	37.4	**44.8**	60.3	39.5	77.8	92.3
$\text{SimKGC}_{\text{IB+PB}}$	35.4	30.2	37.3	**44.8**	60.2	39.4	77.7	**92.4**
$\text{SimKGC}_{\text{IB+SN}}$	35.6	31.0	37.3	43.9	71.3	60.7	**78.7**	91.3
$\text{SimKGC}_{\text{IB+PB+SN}}$	**35.8**	**31.3**	**37.6**	44.1	**71.4**	**60.9**	78.5	91.7

Table 2: Main results for the Wikidata5M dataset. "IB", "PB", and "SN" refer to in-batch negatives, pre-batch negatives, and self-negatives respectively. Embedding-based methods are inherently unable to perform inductive KGC. According to the evaluation protocol by Wang et al. (2021c), the inductive setting only ranks $7,475$ entities in the test set, while the transductive setting ranks ~ 4.6 million entities, so the reported metrics for the inductive setting are much higher. Results are statistically significant under paired student's t-test with p-value 0.05.

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
$\text{SimKGC}_{\text{IB}}$	**67.1**	58.5	**73.1**	**81.7**	33.3	24.6	36.2	51.0
$\text{SimKGC}_{\text{IB+PB}}$	66.6	57.8	72.3	**81.7**	33.4	24.6	**36.5**	**51.1**
$\text{SimKGC}_{\text{IB+SN}}$	66.7	**58.8**	72.1	80.5	33.4	24.7	36.3	50.9
$\text{SimKGC}_{\text{IB+PB+SN}}$	66.6	58.7	71.7	80.0	**33.6**	**24.9**	36.2	**51.1**

Table 3: Main results for WN18RR and FB15k-237 datasets. †: numbers are from Wang et al. (2021a).

We report results for various combinations of negatives. With in-batch negatives only, the performance of $\text{SimKGC}_{\text{IB}}$ is already quite strong thanks to the large batch size (1024) we use. Adding self-negatives tends to improve H@1 but hurt H@10. We hypothesize that self-negatives make the model rely less on simple text match. Thus they have negative impacts on metrics that emphasize recall, such as H@10. Combining all three types of negatives generally has the best results but not always.

Compared to other datasets, the graph for the FB15k-237 dataset is much denser (average degree is ~ 37 per entity), and contains fewer entities ($\sim 15k$). To perform well, models need to learn generalizable inference rules instead of just modeling textual relatedness. Embedding-based methods are likely to hold an advantage for this scenario. It is possible to ensemble our method with embedding-based ones, as done by Wang et al. (2021a). Since this is not the main focus of this paper, we leave it as future work. Also, Cao et al. (2021) points out that many links in the FB15k-237 dataset are not predictable based on the available information. These two reasons help explain the unsatisfactory performance of SimKGC.

Adding self-negatives is particularly helpful for the inductive setting of Wikidata5M dataset, with MRR rising from 60.3% to 71.3%. For inductive KGC, text-based models rely more heavily on text match than the transductive setting. Self negatives can prevent the model from simply predicting the given head entity.

In terms of inference time, the most expensive part is the forward pass with BERT. For the Wikidata5M-Trans dataset, SimKGC requires ~ 40 minutes to compute ~ 4.6 million embeddings with 2 GPUs, while cross-encoder models such as KG-BERT (Yao et al., 2019) would require an estimated time of 3000 hours. We are not the first work that enables fast inference, models such as ConvE (Dettmers et al., 2018) and StAR (Wang et al., 2021a) also share similar advantages. Here we just want to re-emphasize the importance of inference efficiency and scalability when designing new models.

5 Analysis

We conduct a series of analyses to gain further insights into our proposed model and the KGC task.

5.1 What Makes SimKGC Excel?

Compared to existing text-based methods, SimKGC makes two major changes: using more negatives, and switching from margin-based ranking loss to InfoNCE loss. To guide the future work on knowledge graph completion, it is crucial to understand which factor contributes most to the superior performance of SimKGC.

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
margin-τ	255	57.8	48.5	63.7	74.9

Table 4: Analysis of loss function and the number of negatives on the WN18RR dataset.

In Table 4, we use SimKGC$_{\text{IB}}$ with batch size 256 as a baseline. By reducing the number of negatives from 255 to 5, MRR drops from 64.4 to 48.8. Changing the loss function from InfoNCE to the following margin loss makes MRR drop to 39.5:

$$\frac{1}{|\mathcal{N}|}\sum_{i=1}^{|\mathcal{N}|}\max(0, \lambda + \phi(h,r,t'_i) - \phi(h,r,t)) \quad (6)$$

Consistent with Equation 5, $\phi(h,r,t'_i)$ is cosine similarity score for a candidate triple, and $\lambda = 0.8$.

To summarize, both InfoNCE loss and a large number of negatives are important factors, while the loss function seems to have bigger impacts. For InfoNCE loss, the hard negatives naturally contribute larger gradients, and adding more negatives can lead to more robust representations. Wang and Liu (2021) also draws a similar conclusion: such hardness-aware property is vital for the success of contrastive loss.

We also propose a variant "margin-τ" loss by changing the weight in Equation 6 from $\frac{1}{|\mathcal{N}|}$ to $\frac{\exp(s(t'_i)/\tau)}{\sum_{j=1}^{|\mathcal{N}|}\exp(s(t'_j)/\tau)}$, where $s(t'_i) = \max(0, \lambda + \phi(h,r,t'_i) - \phi(h,r,t))$ and $\tau = 0.05$. Similar to InfoNCE loss, "margin-τ" loss makes the model pay more attention to hard negatives and leads to better performance as shown in Table 4. It is similar to the "self-adversarial negative sampling" proposed by Sun et al. (2019b). Most hyperparameters are tuned based on InfoNCE loss. We expect the margin-τ loss to achieve better results with a bit more hyperparameter optimization.

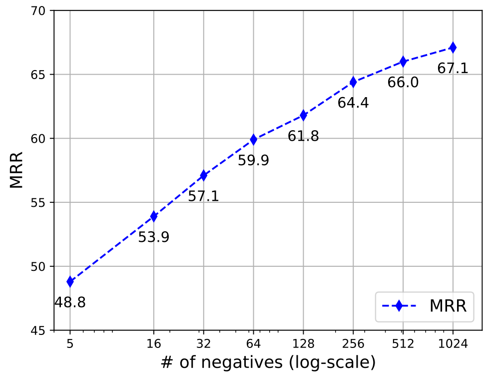

Figure 2: MRR on the WN18RR dataset w.r.t the number of negatives with SimKGC$_{\text{IB}}$. We use a batch size of 1024 for all experiments, and change the number of negatives with a binary mask over the softmax logits.

In Figure 2, we quantitatively illustrate how MRR changes as more negatives are added. There is a clear trend that the performance steadily improves from 48.8 to 67.1. However, adding more negatives requires more GPU memory and may cause optimization difficulties (You et al., 2020; Chen et al., 2020). We do not experiment with batch size larger than 1024.

5.2 Ablation on Re-ranking

Our proposed re-ranking strategy is a simple way to incorporate topological information in the knowledge graph. For graphs whose connectivity patterns exhibit spatial locality, re-ranking is likely to help.

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
triple	(1997 IBF World Championships, followed by, **1999 IBF World Championships**)
evidence	The 10th IBF World Championships (Badminton) were held in Glasgow, Scotland, between 24 May and 1 June 1997...
SimKGC	2000 IBF World Junior Championships

Table 5: Examples of SimKGC prediction results on the test set of the Wikidata5M-Trans dataset. The entity to predict is in bold font. We only show a snippet of relevant texts in the row of "evidence" for space reason.

	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.

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 $\text{SimKGC}_{\text{IB+PB+SN}}$.

In Table 6, we see a slight but stable increase for all metrics on the Wikidata5M-Trans dataset. Note that this re-ranking strategy does not apply to inductive KGC since entities in the test set never appear in the training data. Exploring more effective ways such as graph neural networks (Wu et al., 2019) instead of simple re-ranking would be a future direction.

5.3 Fine-grained Analysis

1-1	1-n
spouse	child
capital of	has part
lake inflows	notable work
head of government	side effect
n-1	**n-n**
instance of	cast member
place of birth	member of
given name	influenced by
work location	nominated for

Table 7: Examples for different categories of relations on the Wikidata5M-Trans dataset.

We classify all relations into four categories based on the cardinality of head and tail arguments following the rules by Bordes et al. (2013): one-to-one(1-1), one-to-many(1-n), many-to-one(n-1), and many-to-many(n-n). Examples are shown in Table 7. As shown in Table 8, predicting the "n" side is generally more difficult, since there are many seemingly plausible answers that would confuse the model. Another main reason is the incompleteness of the knowledge graph. Some predicted triples might be correct based on human evaluation, especially for 1-n relations in head entity prediction, such as "instance of", "place of birth" etc.

In Table 5, for the first example, "Marbletown", "Ulster County", and "New York" are both correct answers. The second example illustrates the case for relation "place of birth": a lot of people share the same place of birth, and some triples may not exist in the knowledge graph. This helps explain the low performance of "1-n" relations for the Wikidata5M-Trans dataset. In the third example, SimKGC predicts a closely related but incorrect entity "http server".

5.4 Human Evaluation

The analyses above suggest that automatic evaluation metrics such as MRR tend to underestimate the model's performance. To have a more accurate estimation of the performance, we conduct human evaluation and list the results in Table 9. An average of 49% of the wrong predictions according to H@1 are correct according to human annotators. If we take this into account, the H@1 of our proposed model would be much higher. How to accurately

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

Table 9: Human evaluation results on the Wikidata5M-Trans dataset. (h, r, ?) and (?, r, t) denote tail entity and head entity prediction respectively. We randomly sample 100 wrong predictions according to H@1 from test set. The "unknown" category indicates annotators are unable to decide whether the prediction is correct or wrong based on the textual information.

measure the performance of KGC systems is also an interesting future research direction.

5.5 Entity Visualization

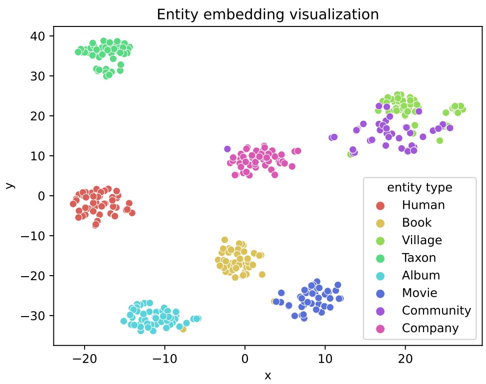

Figure 3: 2-D visualization of the entity embeddings from the Wikidata5M-Trans dataset with t-SNE (Maaten and Hinton, 2008).

To examine our proposed model qualitatively, we visualize the entity embeddings from 8 largest categories [5] with 50 randomly selected entities per category. Entity embeddings are computed with BERT_t in Section 3.2. In Figure 3, different categories are well separated, demonstrating the high quality of the learned embeddings. One interesting phenomenon is that the two categories "Community" and "Village" have some overlap. This is reasonable since these two concepts are not mutually exclusive.

6 Conclusion

This paper proposes a simple method SimKGC to improve text-based knowledge graph completion. We identify that the key issue is how to perform efficient contrastive learning. Leveraging the recent progress in the field of contrastive learning, SimKGC adopts a bi-encoder architecture and combines three types of negatives. Experiments on the WN18RR, FB15k-237, and Wikidata5M datasets show that SimKGC substantially outperforms state-of-the-art methods.

For future work, one direction is to improve the interpretability of SimKGC. In methods like RotatE (Sun et al., 2019b) and TransE (Bordes et al., 2013), a triple can be modeled as rotation in complex space or relational translation, while SimKGC does not enable such easy-to-understand interpretations. Another direction is to explore effective ways to deal with false negatives (Huynh et al., 2020) resulting from the incompleteness of knowledge graphs.

[5] We utilize the "instance of" relation to determine the entity category.

7 Broader Impacts

Future work could use SimKGC as a solid baseline to keep improving text-based knowledge graph completion systems. Our experimental results and analyses also reveal several promising research directions. For example, how to incorporate global graph structure in a more principled way? Are there other loss functions that perform better than the InfoNCE loss? For knowledge-intensive tasks such as knowledge base question answering (KBQA), information retrieval, and knowledge-grounded response generation, etc., it would be interesting to explore the new opportunities brought by the improved knowledge graph completion systems.

Acknowledgements

We would like to thank anonymous reviewers and area chairs for their valuable comments, and ACL Rolling Review organizers for their efforts.

References

Ivana Balazevic, Carl Allen, and Timothy Hospedales. 2019. TuckER: Tensor factorization for knowledge graph completion. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5185–5194, Hong Kong, China. Association for Computational Linguistics.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM*

SIGMOD international conference on Management of data, pages 1247–1250.

Antoine Bordes, Nicolas Usunier, Alberto García-Durán, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pages 2787–2795.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Yixin Cao, Xiang Ji, Xin Lv, Juanzi Li, Yonggang Wen, and Hanwang Zhang. 2021. Are missing links predictable? an inferential benchmark for knowledge graph completion. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6855–6865, Online. Association for Computational Linguistics.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. 2020. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 1597–1607. PMLR.

Daniel Daza, Michael Cochez, and Paul Groth. 2021. Inductive entity representations from text via link prediction. In *Proceedings of the Web Conference 2021*, pages 798–808.

Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. 2018. Convolutional 2d knowledge graph embeddings. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 1811–1818. AAAI Press.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2019. Wizard of wikipedia: Knowledge-powered conversational agents. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. *ArXiv preprint*, abs/2104.08821.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Ávila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. 2020. Bootstrap your own latent - A new approach to self-supervised learning. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Beliz Gunel, Jingfei Du, Alexis Conneau, and Veselin Stoyanov. 2021. Supervised contrastive learning for pre-trained language model fine-tuning. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 9726–9735. IEEE.

Jin Huang, Wayne Xin Zhao, Hongjian Dou, Ji-Rong Wen, and Edward Y. Chang. 2018. Improving sequential recommendation with knowledge-enhanced memory networks. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2018, Ann Arbor, MI, USA, July 08-12, 2018*, pages 505–514. ACM.

Tri Huynh, Simon Kornblith, Matthew R. Walter, Michael Maire, and Maryam Khademi. 2020. Boosting contrastive self-supervised learning with false negative cancellation. *ArXiv preprint*, abs/2011.11765.

Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. 2020. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438.

Jeff Johnson, M. Douze, and H. Jégou. 2021. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, 7:535–547.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Bosung Kim, Taesuk Hong, Youngjoong Ko, and Jungyun Seo. 2020. Multi-task learning for knowledge graph completion with pre-trained language models. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1737–1743, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Phuc H Le-Khac, Graham Healy, and Alan F Smeaton. 2020. Contrastive representation learning: A framework and review. *IEEE Access*.

Jinhyuk Lee, Mujeen Sung, Jaewoo Kang, and Danqi Chen. 2021. Learning dense representations of phrases at scale. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6634–6647, Online. Association for Computational Linguistics.

Song Liu, Haoqi Fan, Shengsheng Qian, Yiru Chen, W. Ding, and Zhongyuan Wang. 2021. Hit: Hierarchical transformer with momentum contrast for video-text retrieval. *ArXiv preprint*, abs/2103.15049.

Weijie Liu, Peng Zhou, Zhe Zhao, Zhiruo Wang, Qi Ju, Haotang Deng, and Ping Wang. 2020. K-BERT: enabling language representation with knowledge graph. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 2901–2908. AAAI Press.

L. V. D. Maaten and Geoffrey E. Hinton. 2008. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605.

Chaitanya Malaviya, Chandra Bhagavatula, Antoine Bosselut, and Yejin Choi. 2020. Commonsense knowledge base completion with structural and semantic context. In *AAAI*.

George A. Miller. 1992. WordNet: A lexical database for English. In *Speech and Natural Language: Proceedings of a Workshop Held at Harriman, New York, February 23-26, 1992*.

Jianmo Ni, Noah Constant, Ji Ma, Keith B Hall, Daniel Cer, Yinfei Yang, et al. 2021. Sentence-t5: Scalable sentence encoders from pre-trained text-to-text models. *ArXiv preprint*, abs/2108.08877.

Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. 2011. A three-way model for collective learning on multi-relational data. In *Proceedings of the 28th International Conference on Machine Learning, ICML 2011, Bellevue, Washington, USA, June 28 - July 2, 2011*, pages 809–816. Omnipress.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Yingqi Qu, Yuchen Ding, Jing Liu, Kai Liu, Ruiyang Ren, Wayne Xin Zhao, Daxiang Dong, Hua Wu, and Haifeng Wang. 2021. RocketQA: An optimized training approach to dense passage retrieval for open-domain question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5835–5847, Online. Association for Computational Linguistics.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Colin Raffel, Noam M. Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, W. Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *ArXiv preprint*, abs/1910.10683.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. 2020. AutoPrompt:

Eliciting Knowledge from Language Models with Automatically Generated Prompts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4222–4235, Online. Association for Computational Linguistics.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, pages 4444–4451. AAAI Press.

Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2007. Yago: a core of semantic knowledge. In *Proceedings of the 16th International Conference on World Wide Web, WWW 2007, Banff, Alberta, Canada, May 8-12, 2007*, pages 697–706. ACM.

Haitian Sun, Tania Bedrax-Weiss, and William Cohen. 2019a. PullNet: Open domain question answering with iterative retrieval on knowledge bases and text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2380–2390, Hong Kong, China. Association for Computational Linguistics.

Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. 2019b. Rotate: Knowledge graph embedding by relational rotation in complex space. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Komal Teru, Etienne Denis, and Will Hamilton. 2020. Inductive relation prediction by subgraph reasoning. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 9448–9457. PMLR.

Kristina Toutanova, Danqi Chen, Patrick Pantel, Hoifung Poon, Pallavi Choudhury, and Michael Gamon. 2015. Representing text for joint embedding of text and knowledge bases. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1499–1509, Lisbon, Portugal. Association for Computational Linguistics.

Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. 2016. Complex embeddings for simple link prediction. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, volume 48 of *JMLR Workshop and Conference Proceedings*, pages 2071–2080. JMLR.org.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85.

Bo Wang, Tao Shen, Guodong Long, Tianyi Zhou, Ying Wang, and Yi Chang. 2021a. Structure-augmented text representation learning for efficient knowledge graph completion. In *Proceedings of the Web Conference 2021*, pages 1737–1748.

Feng Wang and Huaping Liu. 2021. Understanding the behaviour of contrastive loss. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2495–2504.

Liang Wang, Wei Zhao, and Jingming Liu. 2021b. Aligning cross-lingual sentence representations with dual momentum contrast. In *EMNLP*.

Xiaozhi Wang, Tianyu Gao, Zhaocheng Zhu, Zhengyan Zhang, Zhiyuan Liu, Juanzi Li, and Jian Tang. 2021c. Kepler: A unified model for knowledge embedding and pre-trained language representation. *Transactions of the Association for Computational Linguistics*, 9:176–194.

Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014. Knowledge graph embedding by translating on hyperplanes. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, July 27 -31, 2014, Québec City, Québec, Canada*, pages 1112–1119. AAAI Press.

Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, C. Zhang, and Philip S. Yu. 2019. A comprehensive survey on graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 32:4–24.

Ruobing Xie, Zhiyuan Liu, Jia Jia, Huanbo Luan, and Maosong Sun. 2016. Representation learning of knowledge graphs with entity descriptions. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA*, pages 2659–2665. AAAI Press.

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul N. Bennett, Junaid Ahmed, and Arnold Overwijk. 2021. Approximate nearest neighbor negative contrastive learning for dense text retrieval. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. 2015. Embedding entities and relations for learning and inference in knowledge bases. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Yinfei Yang, Gustavo Hernández Ábrego, Steve Yuan, Mandy Guo, Qinlan Shen, Daniel Cer, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2019. Improving multilingual sentence embedding using bidirectional dual encoder with additive margin softmax. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 5370–5378. ijcai.org.

Liang Yao, Chengsheng Mao, and Yuan Luo. 2019. Kg-bert: Bert for knowledge graph completion. *ArXiv preprint*, abs/1909.03193.

Yang You, Jing Li, Sashank J. Reddi, Jonathan Hseu, Sanjiv Kumar, Srinadh Bhojanapalli, Xiaodan Song, James Demmel, Kurt Keutzer, and Cho-Jui Hsieh. 2020. Large batch optimization for deep learning: Training BERT in 76 minutes. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Hanwen Zha, Zhiyu Chen, and Xifeng Yan. 2021. Inductive relation prediction by bert. *ArXiv preprint*, abs/2103.07102.

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. ERNIE: Enhanced language representation with informative entities. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1441–1451, Florence, Italy. Association for Computational Linguistics.

Zexuan Zhong, Dan Friedman, and Danqi Chen. 2021. Factual probing is [MASK]: Learning vs. learning to recall. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5017–5033, Online. Association for Computational Linguistics.

A Details on Hyperparameters

Hyperparameter	value
# of GPUs	4
initial temperature τ	0.05
gradient clip	10
warmup steps	400
batch size	1024
max # of tokens	50
weight α for re-ranking	0.05
dropout	0.1
weight decay	10^{-4}
InfoNCE margin	0.02
pooling	mean

Table 10: Shared hyperparameters for our proposed SimKGC model.

In Table 10, we show the hyperparameters that are shared across all the datasets. For learning rate, we use 5×10^{-5}, 10^{-5}, and 3×10^{-5} for WN18RR, FB15k-237, and Wikidata5M datasets, respectively. For re-ranking, we use 5-hop neighbors for WN18RR and 2-hop neighbors for other datasets. Each epoch takes $\sim$ 3 minutes for WN18RR, $\sim$ 12 minutes for FB15k-237, and $\sim$ 12 hours for Wikidata5M (both settings). Our implementation is based on open-source project *transformers* [6].

For inverse relation r^{-1}, we add a prefix word "inverse" to the description of r. For examples, if r = "instance of", then r^{-1} = "inverse instance of".

Some entities in the WN18RR and FB15k-237 dataset have very short textual descriptions. We concatenate them with the entity names of its neighbors in the training set. To avoid label leakage during training, we dynamically exclude the correct entity in the input text.

B More Analysis Results

batch size	MRR	H@1	H@3	H@10
256	33.8	28.7	35.8	43.1
512	34.6	29.4	36.7	43.7
1024	**35.3**	**30.1**	**37.4**	**44.8**

Table 11: Effects of batch size on the Wikidata5M-Trans dataset with $\text{SimKGC}_{\text{IB}}$.

batch size	MRR	H@1	H@3	H@10
256	32.4	23.3	35.4	50.9
512	32.7	23.7	35.6	**51.0**
1024	**33.3**	**24.6**	**36.2**	**51.0**

Table 12: Effects of batch size on the FB15k-237 dataset with $\text{SimKGC}_{\text{IB}}$.

margin γ	MRR	H@1	H@3	H@10
0	33.4	24.8	36.0	50.9
0.02	**33.6**	24.9	**36.2**	**51.1**
0.05	**33.6**	**25.0**	**36.2**	50.9

Table 13: Ablation for the additive margin γ of InfoNCE loss on the FB15k-237 dataset.

In Table 11 and 12, we show how the batch size affects model performance on the Wikidata5M-Trans and FB15k-237 dataset.

In Equation 5, we use a variant of InfoNCE loss that has an additive margin γ. In our experiments, such a variant performs consistently better than the standard InfoNCE loss, though the improvement is quite marginal, as shown in Table 13.

In Table 14, we show more examples of SimKGC predictions on the Wikidata5M-Trans

[6]https://github.com/huggingface/transformers

triple	(captive state (film), instance of, **movie**)
evidence	Captive State is a 2019 American crime science fiction thriller film directed by Rupert Wyatt and co-written by Wyatt and Erica Beeney.. . .
SimKGC	3-D movies
triple	(Lionel Belasco, occupation, **composer**)
evidence	Lionel Belasco (1881 – c. 24 June 1967) was a prominent pianist, composer and bandleader, best known for his calypso recordings.
SimKGC	bandleaders
triple	(**Johan Nordhagen**, country of citizenship, Norway)
evidence	Waqas Ahmed (born 9 June 1991) is a Norwegian cricketer. . . .
SimKGC	Waqas Ahmed
triple	(**Carlos Peña Romulo**, position held, philippine resident commissioner)
evidence	Francis Burton Harrison was an American-born Filipino statesman who served in the United States House of Representatives and was appointed Governor-General of the Philippines . . .
SimKGC	Francis Burton Harrison

Table 14: More examples of SimKGC prediction results on the test set of Wikidata5M-Trans.

dataset to help better understand our model's behavior. Full model predictions on test datasets are available in our public code repository.