Contrastive Learning with Graph Context Modeling for Sparse Knowledge Graph Completion

Anonymous ACL submission

Abstract

Knowledge Graph Embeddings (KGE) aim to map entities and relations to high dimensional spaces and have become the *de-facto* standard for knowledge graph completion. Most existing KGE methods suffer from the sparsity challenge, where it is harder to predict entities that appear less frequently in knowledge graphs. In this work, we propose a novel framework KRACL to alleviate the widespread sparsity in KGs with graph context and contrastive learning. Firstly, we propose the Knowledge Relational Attention Network (KRAT) to leverage the graph context by jointly aggregating neighbors and relations with the attention mechanism. KRAT is capable of capturing the subtle importance of different context triples and leveraging multi-hop information in knowledge graphs. Secondly, we propose the knowledge contrastive loss by combining the contrastive loss with cross entropy loss, which introduces more negative samples and thus enriches the feedback to sparse entities. Our experiments demonstrate that KRACL achieves superior results across various standard knowledge graph benchmarks, especially on WN18RR and NELL-995 which have many low in-degree entities. Extensive experiments also bear out KRACL's effectiveness of handling sparse knowledge graphs and robustness against noisy triples.

1 Introduction

Knowledge graphs (KGs) are collections of large-scale facts in the form of structural triples *(subject, relation, object)*, denoted as (s, r, o), e.g., *(Christopher Nolan, Born-in, London)*. These KGs reveal the relations between entities and play an important role in many applications such as natural language processing (Wu et al., 2021b; Cheng et al., 2021; Yasunaga et al., 2021; Zhang et al., 2022a), computer vision (Fang et al., 2017; Gao et al., 2019), and recommender systems (Zhou et al., 2020, 2021). Although KGs already contain millions of facts, they are still far from complete, e.g., 71% of people in the Freebase knowledge graph have no birthplace and 75% have no nationality (Dong et al., 2014), which leads to poor performance on downstream applications. Therefore, knowledge graph completion (KGC) is an important task to predict whether a given triple is valid or not and further expands existing knowledge graphs.

Most existing KGs are stored in symbolic form while downstream applications always involve numerical computation in continuous spaces. To address this issue, researchers proposed to map entities and relations to high dimensional embeddings dubbed knowledge graph embedding (KGE) and these models yield state-of-the-art performance for KGC. TransE (Bordes et al., 2013) is the pioneering work that maps both entities and relations to the latent space by forcing $s + r \approx o$. DistMult (Yang et al., 2014) then proposes to deal with triples using tensor decomposition and score them with a bilinear function. Due to their simple operations and limited parameters, these non-neural models usually produce low-quality embeddings. Recently, neural network-based models greatly improve the performance of KGE (Vashishth et al., 2020; Saxena et al., 2022). For instance, ConvE (Dettmers et al., 2018) reshapes entity embeddings and feeds them into a 2D convolution network for scoring. However, such approaches can only process triples independently and ignore the vast structural and context information in KGs. Graph Neural Network (GNN) is then employed to encode graph structure in KGs. Specifically, CompGCN (Vashishth et al., 2019) introduces a message passing scheme that equally aggregates entity and relation embedding and scores triples in the encoder-decoder framework. These GNN-based KGE models incorporate KGs' structural and semantic information and have achieved state-of-the-art results.

Although much research progress has been made

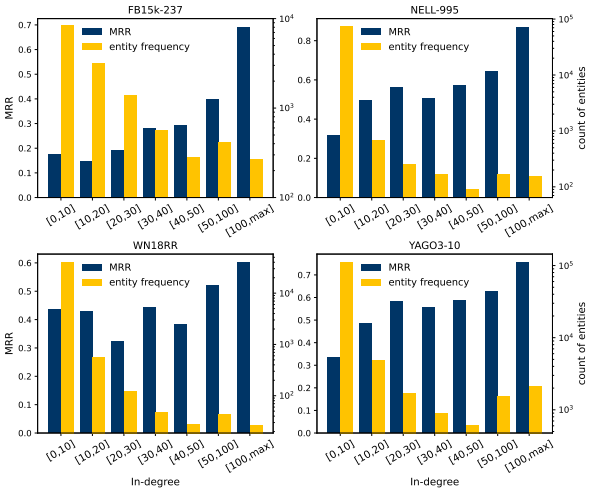

Figure 1: The number and mean reciprocal rank (MRR) of different frequency entities based on RotatE results on FB15k-237, WN18RR, NELL-995, and YAGO3-10 benchmark datasets. This figure reveals the common existence of sparse entities and their poor prediction performance in knowledge graphs.

by recent KGE models, predicting entities that rarely appear in knowledge graphs remains challenging. We investigate the in-degree (using entity frequency) and link prediction performance (using MRR) on several widely acknowledged knowledge graphs, including FB15k-237 (Toutanova and Chen, 2015), WN18RR (Dettmers et al., 2018), NELL-995 (Xiong et al., 2017), and YAGO3-10 (Suchanek et al., 2007) (shown in Figure 1). The yellow bars show that a large portion of entities rarely appear in knowledge graph triples, leading to the limited facts for knowledge graph completion. Moreover, it also reveals the common existence of sparse entities across various datasets. The blue bars show the link prediction performance for entities of different in-degree with RotatE (Sun et al., 2019). We observe that the prediction results are strongly relevant to the entity in-degree, and the prediction performance of sparse entities is much worse than those of frequent entities.

In this work, we propose KRACL (**K**nowledge **R**elational **A**ttention Network with **C**ontrastive **L**earning) to alleviate the sparsity issue in KGs. First, we employ **K**nowledge **R**elational **AT**tention Network (KRAT) to fully leverage the graph context in KG. Specifically, we calculate the attention score for each context triple to capture its importance, and then jointly aggregate relation and neighbor with attention score to enrich the sparse entity's embedding. Second, we project subject entity embedding to object embedding with knowledge projection head, e.g. ConvE, RotatE, DistMult, and TransE. Finally, we optimize the model with proposed knowledge contrastive loss, i.e. combining the contrastive loss and cross entropy loss. We empirically find that contrastive loss can provide more feedback to sparse entities and is more robust against sparsity when compared to explicit negative sampling. Extensive experiments on various standard benchmarks show the superiority of our proposed KRACL model over competitive peer models, especially on WN18RR and NELL-995 with many low in-degree nodes. Our key contributions are summarized as follows:

- We propose the **K**nowledge **R**elational **AT**tention Network (KRAT) to integrate knowledge graph context by jointly fusing relation and entity context with the attention mechanism. After stacking several layers of KRAT, we fuse the multi-hop context information into the entity embeddings and take advantage of context from neighboring entities and relations.

- We propose a knowledge contrastive loss to alleviate the sparsity of knowledge graphs. We incorporate contrastive loss with cross entropy loss to introduce more negative samples, which can enrich the limited positive triples in knowledge graphs and enhance prediction performance for sparse entities.

- Experimental results demonstrate that our proposed KRACL framework achieves superior performance on five standard benchmarks, especially on WN18RR and NELL-995 with many low in-degree entities.

2 Related Work

2.1 Knowledge Graph Embedding

Non-Neural Non-neural models embed entities and relations into latent space with linear operations. Starting from TransE (Bordes et al., 2013), the pioneering and most representative translational model, a series of models are proposed in this line, such as TransH (Wang et al., 2014), TransR (Lin et al., 2015), and TransD (Ji et al., 2015). RotatE (Sun et al., 2019) extends the translational model to complex space and OTE (Tang et al., 2020) further extends RotatE to high dimensional space. There

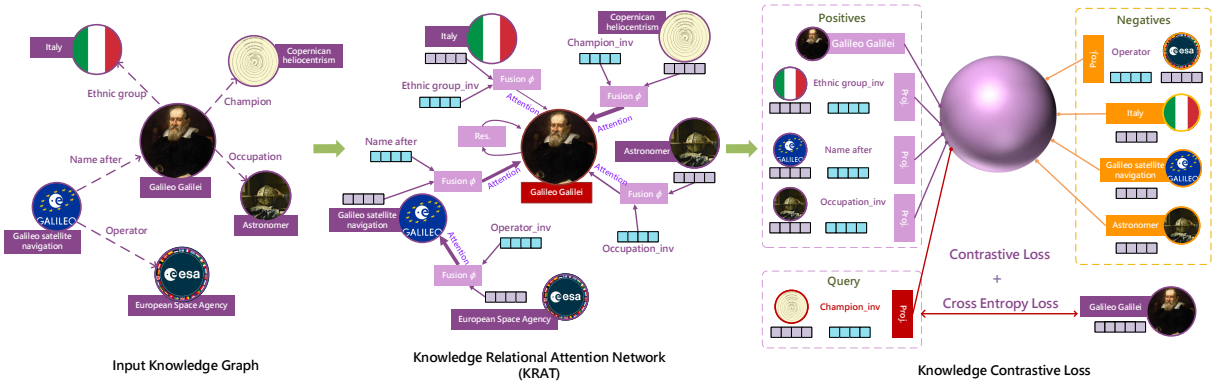

Figure 2: Overview of our proposed KRACL framework to alleviate the sparsity problem in knowledge graphs.

is another line of work that takes tensor decomposition to compute the plausibility of triples. For instance, RESCAL (Nickel et al., 2011) and DistMult (Yang et al., 2014) represent each relation with a full rank matrix and diagonal matrix, respectively. ComplEx (Trouillon et al., 2016) generalizes DistMult to complex space to enhance the expressiveness of complex relations. Furthermore, the non-neural model can also project to Gaussian distribution (He et al., 2015b), manifold (Xiao et al., 2016a), and Lie group (Xiao et al., 2016b).

Neural Network-based Neural network-based KGE models are introduced for KGC due to their inherent strong learning ability. Convolutional neural networks are employed to extract the semantic features from KGE. Specifically, ConvE (Dettmers et al., 2018) utilizes 2D convolution to learn deep features of entities and relations. ConvKB (Nguyen et al., 2017) adopts 1D convolution and feeds the whole triple into the convolutional neural network. HypER (Balažević et al., 2019) employs hypernetwork to generate relation-special filters. Graph neural networks also show strong potential in learning knowledge graphs embedding by incorporating graph structure in KGs. R-GCN (Schlichtkrull et al., 2018) is an extension of the graph convolution neural network (Kipf and Welling, 2016) for relational data. SACN (Shang et al., 2019) encodes node structure and relation types with weighted GCN. CompGCN (Vashishth et al., 2019) jointly embeds both entities and relations in KG through a compositional operator. KBAT (Nathani et al., 2019) proposes to distinguish the weight of neighboring nodes with the attention mechanism.

2.2 Contrastive Learning

Contrastive learning has been a popular approach for self-supervised learning by pulling semantically close neighbors together while pushing apart non-neighbors away (Hadsell et al., 2006). As is first introduced in the computer vision domain, a large collection of works (Hadsell et al., 2006; He et al., 2020; Chen et al., 2020; Tian et al., 2020) learn self-supervised image representations by minimizing the distance between two augmented views of the same image. Khosla et al. (2020) further extends contrastive learning to the supervised setting by considering the representations from the same class as positive samples. Contrastive learning also achieves great success in natural language processing (Gao et al., 2021; Zhang et al., 2022b; Das et al., 2022) and graph representation learning (You et al., 2020; Zhu et al., 2021). However, contrastive learning has not been widely applied to knowledge graphs and we explore its potential to alleviate knowledge graphs' sparsity in this work.

3 Methodology

We consider a knowledge graph as a collection of factual triples $\mathcal{D} = \{(s, r, o)\}$ with $\mathcal{E}$ as entity set and $\mathcal{R}$ as relation set. Each triple has a subject entity s and object entity o, where $s, o \in \mathcal{E}$. Relation $r \in \mathcal{R}$ connects two entities with direction from subject to object. Next, we introduce a novel framework–**K**nowledge **R**elational **A**ttention Network with **C**ontrastive **L**earning (KRACL) for knowledge graph completion. KRACL is twofold, we first introduce the **K**nowledge **R**elational **AT**tention Network (KRAT) that aggregates the graph context information in KG, then we describe **K**nowledge **C**ontrastive **L**earning (KCL) to alleviate the sparsity problem in details.

3.1 Knowledge Relational Attention Network

To fully leverage the limited context information in sparse KGs, we propose KRAT to jointly ag-

gregate neighbor entities and relation context with the attention mechanism. Inspired by Velickovic et al. (2018); Brody et al. (2021), we calculate the attention score $w_{s,r,o}$ for each context triple as

$$w_{sro} = \boldsymbol{a}^{(l)} LeakyReLU(\boldsymbol{W}^{(l)}[\boldsymbol{h}_s||\boldsymbol{h}_r||\boldsymbol{h}_o]), \tag{1}$$

where w_{sro} denotes the attention score for triple (s, r, o), $\boldsymbol{a}^{(l)} \in \mathbb{R}^{1\times d}$ and $\boldsymbol{W}^{(l)} \in \mathbb{R}^{d_e\times(2d_e+d_r)}$ are learnable parameters specific for the l-th layer of KRAT, $\boldsymbol{h}_s, \boldsymbol{h}_r, \boldsymbol{h}_o$ denote the hidden representations of subject entity, relation, and object entity in the $l-1$ layer. Then the attention score of each triple is normalized with softmax as

$$\begin{aligned}\alpha_{sro} &= softmax_{sr}(w_{sro}) \\ &= \frac{exp(w_{sro})}{\sum_{n\in\mathcal{N}_o}\sum_{p\in\mathcal{R}_{no}} exp(w_{npo})},\end{aligned} \tag{2}$$

where $\mathcal{N}_o$ denotes the neighbor entities of o, $\mathcal{R}_{no}$ denotes the relation that connects with n and o, α_{sro} is the normalized attention weight for triple (s, r, o). We then aggregate the context information to obtain entity representation, i.e.,

$$\boldsymbol{h}_o^{(l)} = \sigma(\alpha_{sro}\boldsymbol{W}_{\lambda(r)}^{(l)}\phi(\boldsymbol{h}_s, \boldsymbol{h}_r) + \boldsymbol{W}_{res}^{(l)}\boldsymbol{h}_o^{(l-1)}), \tag{3}$$

where σ denotes Tanh activation function, $\boldsymbol{W}_{\lambda(r)}^{(l)} \in \mathbb{R}^{d_l\times d_{l-1}}$ denotes learnable weight specific to edge type $\lambda(r)$, e.g., incoming and outgoing edge. $\phi(\cdot)$ denotes the fusion operator that combines relation and entity context. Inspired by CompGCN (Vashishth et al., 2019), we take circular-correlation as the default operator while more operators are discussed in section 4.7. We also add a pre-activation residual connection to prevent over-smoothing (Li et al., 2020). The relation representations are updated through a linear transformation

$$\boldsymbol{h}_r^{(l)} = \boldsymbol{W}_{rel}^{(l)} \cdot \boldsymbol{h}_r^{(l-1)}, \tag{4}$$

where $\boldsymbol{W}_{rel}^{(l)} \in \mathbb{R}^{d_r\times d_r}$ is a trainable matrix for relation embeddings under the l-th layer.

3.2 Knowledge Contrastive Learning

After passing T layers of KRAT, the entity representations are enriched with T hops context. Taking the idea of supervised contrastive learning (Gunel et al., 2021) that pulls embeddings from the same entities close and pushes embeddings from different entity further away, we calculate the contrastive loss as

$$\mathcal{L}_{CL} = \sum_{o\in\mathcal{T}} \frac{-1}{|\mathcal{T}_o|} \sum_{\boldsymbol{z}_{(s,r)}\in\mathcal{T}_o} log\frac{exp(\boldsymbol{z}_{(s,r)}\cdot\boldsymbol{h}_o/\tau)}{\sum_{k\notin\mathcal{T}_o} exp(\boldsymbol{z}_k\cdot\boldsymbol{h}_o/\tau)}, \tag{5}$$

where $\mathcal{T}$ denotes a batch of normalized entity embeddings, $\mathcal{T}_o$ denotes the set of representations corresponding to entity o, τ is an adjustable temperature hyperparameter that controls the balance between uniformity and tolerance (Wang and Liu, 2021). The contrastive loss introduces more negative samples, therefore enriching the feedback to the limited positive triples. $z_{(s,r)}$ is a knowledge projection head such as TransE, DistMult, RotatE, and ConvE to transform embeddings from subject to object. Here we take ConvE as an example[1]

$$\boldsymbol{z}_{(s,r)} = f(vec(f(\left[\overline{\boldsymbol{h}_s}||\overline{\boldsymbol{h}_r}\right] * \omega))\boldsymbol{W}_p), \tag{6}$$

where $\overline{\boldsymbol{h}_s} \in \mathbb{R}^{d_w\times d_h}$ and $\overline{\boldsymbol{h}_r} \in \mathbb{R}^{d_w\times d_h}$ denote 2D reshaping of $\boldsymbol{h}_s \in \mathbb{R}^{d_w d_h\times 1}$ and $\boldsymbol{h}_r \in \mathbb{R}^{d_w d_h\times 1}$ respectively, $*$ denotes the convolution operation, f denotes non-linearity (PReLU (He et al., 2015a) by default), vec denotes vectorization, and $\boldsymbol{W}_p$ is a linear transformation matrix. The whole formula represents the predicted object representation given the subject s and relation r. We then calculate the cross entropy loss as follows

$$\mathcal{L}_{CE} = -\frac{1}{|\mathcal{T}|}\sum_{(s,r)\in\mathcal{T}}\sum_{o\in\mathcal{E}} y_{(s,r)}^o \cdot \log \hat{y}_{(s,r)}^o, \tag{7}$$

where $\mathcal{T}$ denotes training triples in a batch, $\mathcal{E}$ denotes all entities that exist in the KG, $y_{(s,r)}^o$ denotes the ground-truth labels, i.e., $y_{(s,r)}^o = 1$ if triple (s, r, o) is valid and $y_{(s,r)}^o = 0$ otherwise. $z_{(s,r)}$ is 1-N scoring function taken from ConvE (Dettmers et al., 2018), which scores all candidate entities with dot product

$$\hat{y}_{(s,r)}^o = \boldsymbol{z}_{(s,r)} \cdot \boldsymbol{h}_o^T, \tag{8}$$

where $\hat{y}_{(s,r)}^o$ denotes the predicted plausibility for triple (s, r, o), $\boldsymbol{h}_o \in \mathbb{R}^{d\times|\mathcal{E}|}$ denotes the representations of all entities. Finally, we demonstrate the final objective by incorporating the contrastive loss and cross entropy loss through summation,

$$\mathcal{L} = \mathcal{L}_{CL} + \mathcal{L}_{CE} \tag{9}$$

[1]For more combination of knowledge projection head, please refer to Section 4.7.

By jointly optimizing the two objectives, we capture the similarity of the same entity embeddings and contrast them with other entities, while keeping the performance for entity prediction.

4 Experiment

4.1 Experiment Settings

Dataset To evaluate KRACL, we consider five widely acknowledged datasets: FB15k-237 (Toutanova and Chen, 2015), WN18RR (Dettmers et al., 2018), NELL-995 (Xiong et al., 2017), Kinship (Lin et al., 2018), and UMLS (Kok and Domingos, 2007), following the standard train/test split. Statistics of these benchmarks are listed in Appendix 6, we further investigate the average and medium entity in-degree to demonstrate their sparsity. FB15k-237 and WN18RR are obtained by removing the inverse and equal relations from FB15k and WN18 respectively, making them more difficult. NELL-995 is extracted from the 995-th iteration of NELL system (Mitchell et al., 2018). WN18RR and NELL-995 are much sparser than FB15k-237, Kinship, and UMLS.

Evaluation Protocol Following Bordes et al. (2013), we use the filtered setting for link prediction, i.e., while evaluating test triples, all valid triples are filtered out from the candidate set. We report mean reciprocal rank (MRR), mean rank (MR), and Hits@N. MRR is the average inverse of obtained ranks of correct entities among all candidate entities. MR means the average obtained ranks of correct entities among all candidate entities. Hits@N measures the proportion of correct entities ranked in the top N among all candidate entities. We take N=1,3,10 in this work.

Baselines We compare our model with state-of-the-art KGE models[2] , which can be categorized into: (1) translational-based TransE (Bordes et al., 2013), RotatE (Sun et al., 2019); (2) tensor decomposition-based DistMult (Yang et al., 2014), ComplEx (Trouillon et al., 2016); (3) CNN-based ConvE (Dettmers et al., 2018), ConvKB (Nguyen et al., 2017), HypER (Balažević et al., 2019); (4) GNN-based R-GCN (Schlichtkrull et al., 2018), KBAT (Nathani et al., 2019), CompGCN (Vashishth et al., 2019), and DisenKGAT (Wu et al., 2021a).

Implementation We implement KRACL[3] on a RTX 3090 GPU with 24GB memory using PyTorch (Paszke et al., 2017), Pytorch lightning (Falcon and The PyTorch Lightning team, 2019), and Pytorch Geometric (Fey and Lenssen, 2019). Following Vashishth et al. (2019), each triple (s, r, o) is augmented with a flipped triple (o, r^{-1}, s). We present our hyperparameter settings in Table 9 to facilitate reproducibility. We also use OpenKE (Han et al., 2018) and Pykeen (Ali et al., 2021) library to reproduce the baseline models.

4.2 Main Results

Table 1 and 2 show the link prediction performance on the test set on standard benchmarks including FB15k-237, WN18RR, NELL-995, and Kinship[4]. From the experimental results, we observe that: 1) on sparse knowledge graphs, i.e., WN18RR and NELL-995, KRACL outperforms all other baseline models on most of the metrics. Particularly, MRR is improved from 0.481 and 0.534 in CompGCN to 0.523 and 0.554, about 8.7% and 3.7% relative performance improvement; 2) on dense knowledge graphs, i.e., FB15k-237 and Kinship, KRACL also achieves competitive results compared to baseline models, with significant improvement on Kinship dataset. Overall, these results show the effectiveness of the proposed KRACL for the task of predicting missing links in knowledge graphs and its superior performance on both sparse and dense knowledge graphs.

4.3 Entity In-degree Analysis

Since the sparsity in KGs will lead to entities with low in-degree and thus lack information to conduct link prediction, we follow Shang et al. (2019) and analyze link prediction performance on entities with different in-degree. In the following experiments, we choose FB15k-237 dataset as our object due to its abundant relation types and dense graph structure. As shown in Table 3, we present Hits@10 and MRR metrics on 7 sets of entities within different in-degree scopes and compare the performance of KRACL with TransE, DistMult, ConvE, and CompGCN. Firstly, for entities with low in-degree, GNN-based models such as KRACL and CompGCN outperform ConvE and RotatE, because they get extra information by aggregating neighboring entities. However, we find that simply aggregating neighbors equally is not enough. By varying the importance of every entity's neighborhood and introducing more feedback

[2]More details of baselines can be found in Appendix B.

[3]The code is available at https://anonymous.4open.science/r/KRACL-3448/

[4]Please see main results on UMLS dataset in Appendix 8.

Model	WN18RR					NELL-995				
	MRR	MR	H@10	H@3	H@1	MRR	MR	H@10	H@3	H@1
TransE	.243	2300	.532	.441	.043	.401	2100	.501	.472	.344
DistMult	.444	7000	.504	.47	.412	.485	4213	.61	.524	.401
ComplEx	.449	7882	.53	.469	.409	.482	4600	.606	.528	.399
RotatE	.494	4046	.571	.510	.455	.483	2582	.565	.514	.435
ConvE	.456	4464	.531	.47	.419	.491	3560	.613	.531	.403
ConvKB	.265	**1295**	.558	.445	.058	.43	600	.545	.47	.37
HypER	.493	4687	.549	.503	.464	.540	1763	.657	.580	.471
R-GCN	.123	6700	.207	.137	.08	.12	7600	.188	.126	.082
KBAT	.412	1921	.554	-	-	.319	3683	.474	.370	.233
CompGCN	.481	3113	.548	.492	.448	.534	1246	.644	**.607**	.466
DisenKGAT	.506	4135	.590	.522	.462	.547	882	.666	.598	.474
KRACL (Ours)	**.523**	1754	**.606**	**.539**	**.481**	**.554**	**546**	**.670**	.597	**.484**

Table 1: Link prediction performance on sparse knowledge graphs, i.e., WN18RR and NELL-995. The best score is in **bold** and the second best score is underlined, '-' indicates the result is not reported in previous work.

Model	FB15k-237					Kinship				
	MRR	MR	H@10	H@3	H@1	MRR	MR	H@10	H@3	H@1
TransE	.294	357	.465	-	-	.211	38.9	.470	.252	.093
DistMult	.241	254	.419	.263	.155	.48	7.9	.708	.491	.377
ComplEx	.247	339	.428	.275	.158	.823	2.48	.971	.899	.733
RotatE	.338	177	.533	.375	.241	.738	2.9	.954	.827	.617
ConvE	.325	244	.501	.356	.237	.772	3.0	.950	.858	.665
ConvKB	.243	311	.421	.371	.155	.614	3.3	.953	.755	.436
HypER	.341	250	.520	.376	.252	.868	1.96	.981	.935	.790
R-GCN	.248	339	.428	.275	.158	.109	25.9	.239	.088	.03
KBAT	.156	392	.305	.167	.085	.637	3.41	.955	.757	.470
CompGCN	.355	197	.535	.390	.264	.810	2.26	.977	.892	.709
DisenKGAT	**.368**	179	**.553**	**.407**	**.275**	.832	1.96	.986	.914	.737
KRACL (Ours)	.360	**150**	.548	.395	.266	**.895**	**1.48**	**.991**	**.970**	**.817**

Table 2: Link prediction performance on denser knowledge graphs, i.e., FB15k-237 and Kinship. The best score is in **bold** and the second best score is underlined. '-' indicates the result is not reported in previous work.

with KCL loss, KRACL achieves significant improvement over all baselines for entities with in-degree $[0, 100]$. For entities with higher in-degree, the performance of KRACL is close to ConvE and RotatE, while the performance of CompGCN is the worst, because entity embedding is substantially smoothed by too much neighboring information (Liang et al., 2021). To sum up, these results show the strong capability of KRACL to predict sparse entities and it is also effective for dense entities.

4.4 Knowledge Sparsity Study

To verify KRACL's sensitivity against sparsity, we randomly remove triples from the training set of FB15k-237 and evaluate the models on the full test set. Figure 3 shows the MRR and Hits@10 of 7 competitive models including TransE, DistMult, ComplEx, RotatE, ConvE, HypER, CompGCN, and our proposed KRACL. Performance of all models universally decreases as the training set diminishes. However, the results show that KRACL consistently outperforms all baseline models, and as the corruption ratio increases, the improvement of KRACL against baseline models increases as well. Overall, these experiment results indicate our models' superior robustness against sparsity across a variety of baseline models.

4.5 Ablation Study

As KRACL outperforms various baselines across all selected benchmark datasets, we investigate the impact of each module in KRACL to verify their

In-degree	RotatE		ConvE		CompGCN		KRACL	
	MRR	H@10	MRR	H@10	MRR	H@10	MRR	H@10
$[0, 10]$	.178	.309	.186	.338	.198	.348	**.232**	**.394**
$[10, 20]$	.149	.294	.154	.299	.156	.296	**.181**	**.335**
$[20, 30]$	.194	.381	.199	.386	.198	.370	**.218**	**.405**
$[30, 40]$	.282	.497	.287	.485	.280	.476	**.307**	**.501**
$[40, 50]$	.294	.547	.297	.516	.298	.520	**.328**	**.552**
$[50, 100]$	.399	.681	.403	.675	.400	.663	**.434**	**.702**
$[100, max]$	.691	.929	.714	**.936**	.674	.905	**.716**	.932

Table 3: Link prediction performance categorized by different entity in-degree on the FB15k-237 dataset. The best score is in **bold** and the second best score is underlined.

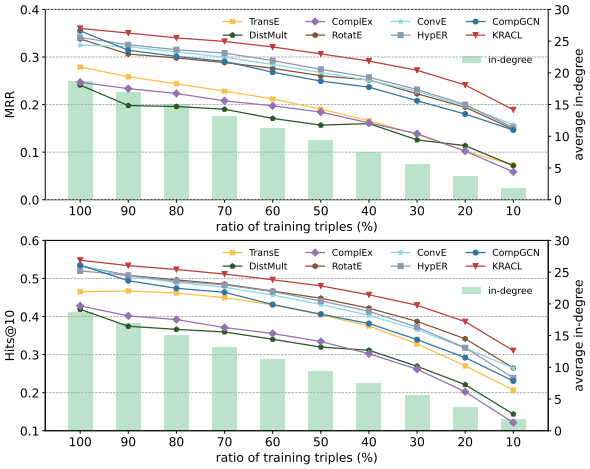

Figure 3: Link prediction performance on sparse knowledge graph of KRACL and competitive peer models on the FB15k-237 datasets.

Model	WN18RR		NELL-995	
	MRR	H@3	MRR	H@3
w/o KRAT	.509	.522	.543	.589
w/o attention	.504	.521	.543	.583
w/o res.	.518	.532	.551	.593
w/o $\mathcal{L}_{CL}$	.502	.514	.496	.541
w/o $\mathcal{L}_{CE}$	.495	.531	.542	.586
$BCELoss$	.469	.478	.507	.547
KRACL	**.523**	**.539**	**.554**	**.597**

Table 4: Results of ablation study of the proposed KRACL on the WN18RR and NELL-995 dataset. $BCELoss$ denotes replacing the KCL loss with binary cross entropy loss.

effectiveness. More specifically, we perform ablation studies on the proposed KRAT and its attention mechanism, residual connection, and test the effectiveness of proposed KCL and its two components on WN18RR and NELL-995 datasets, as is shown in Table 4. First, it is illustrated that full KRACL model outperforms 6 ablated models, which proves the effectiveness of our design choice. Second, we observe a significant drop when replacing the proposed KCL loss with binary cross entropy loss, which is probably resulted from the its poor generalization performance with limited labels (Liu et al., 2016; Cao et al., 2019).

4.6 Performance by Relation Category

In this part, we follow Wang et al. (2014) and further investigate the performance of KRACL in different relation categories (shown in Table 5). We report MRR and Hits@10 of KRACL and compare with TransE, DistMult, ConvE, and CompGCN. We can see that KRACL almost outperforms all baselines for all relation types. Furthermore, it is demonstrated that KRACL achieves significant improvement on 1-1, 1-N, and N-1 relations while the prediction performance on N-N relations is close to CompGCN. We speculate that KRACL is good at learning the relative simple relations and predicting the $N-N$ relation is still challenging to KRACL. We leave the research of a more expressive scheme to model complex N-N relations as our future work.

4.7 Combination of Different GNN Encoder and Projection Head

Borrowing from CompGCN, we evaluate the effect of different GNN methods combined with different knowledge projection heads such as TransE, DistMult, RotatE, and ConvE. The results are shown in Table 7. We evaluate KRAT on four fusion operators taken from Bordes et al. (2013); Yang et al. (2014); Sun et al. (2019); Nickel et al. (2016) ,[5]

- **Subtraction(Sub)**: $\phi(\boldsymbol{h}_s, \boldsymbol{h}_r) = \boldsymbol{h}_s - \boldsymbol{h}_r$
- **Multiplication(Mult)**: $\phi(\boldsymbol{h}_s, \boldsymbol{h}_r) = \boldsymbol{h}_s * \boldsymbol{h}_r$

[5]Please see details of rotation and circular-correlation operator in Appendix F.

		TransE		DistMult		ConvE		CompGCN		KRACL	
		MRR	H@10	MRR	H@10	MRR	H@10	MRR	H@10	MRR	H@10
Head	1-1	.484	.593	.255	.307	.374	.505	.457	.604	**.500**	**.609**
	1-N	.080	.152	.038	.071	.091	.170	.112	.190	**.118**	**.215**
	N-1	.329	.589	.322	.558	.444	.644	.471	.656	**.485**	**.675**
	N-N	.219	.436	.131	.255	.261	.459	.275	.474	**.276**	**.481**
Tail	1-1	.476	.588	.257	.312	.366	.510	.453	.589	**.515**	**.635**
	1-N	.536	.846	.575	.750	.762	.878	.779	.885	**.796**	**.894**
	N-1	.060	.118	.032	.067	.069	.150	.076	.151	**.093**	**.180**
	N-N	.287	.553	.184	.376	.375	.603	**.395**	.616	.394	**.620**

Table 5: Link prediction performance by relation category on FB15k-237 dataset for TransE, DistMult, ConvE, CompGCN, and proposed KRACL. Following Wang et al. (2014), the relations are categorized into one-to-one (1-1), one-to-many (1-N), many-to-one (N-1), and many-to-many (N-N).

- **Rotation(Rot)**: $\phi(\boldsymbol{h}_s, \boldsymbol{h}_r) = \boldsymbol{h}_s \circ \boldsymbol{h}_r$
- **Circular-correlation(Corr)**: $\phi(\boldsymbol{h}_s, \boldsymbol{h}_r) = \boldsymbol{h}_s \star \boldsymbol{h}_r$.

From experimental results in Table 7, we have the following observations. First, by utilizing graph neural networks (GNNs), the model can further incorporate graph structure and context information in the knowledge graph and boost model's performance. The lack of fusing relation and entity embeddings leads to poor performance of R-GCN and W-GCN, while CompGCN and KRACL integrate relation and entity context and outperform other baselines. Second, KRACL obtains an average of 6.3%, 6.0%, 17.6%, and 3.5% relative improvement on MRR compared with CompGCN, which indicates the strong robustness of KRACL across multi-categories knowledge projection heads. We can also see that KRACL significantly outperforms other baseline encoders when combined with RotatE. It reveals the strong robustness and adaptation of the proposed KRACL framework.

4.8 Robustness against Noisy Triples

Beyond sparsity, facts generated by knowledge extraction approaches can also be unreliable, e.g., NELL facts have a precision ranging from 0.75-0.85 for confident extractions and 0.35-0.45 across the broader set of extractions (Mitchell et al., 2018). In this section, we randomly add unreliable triples in the sparse version of FB15k-237 to test models' robustness against noisy triples. Figure 4 shows how the MRR and Hits@10 suffer as noises increase. We observe that KRACL shows consistent improvement over the baseline models and its performance shows a lower level of volatility.

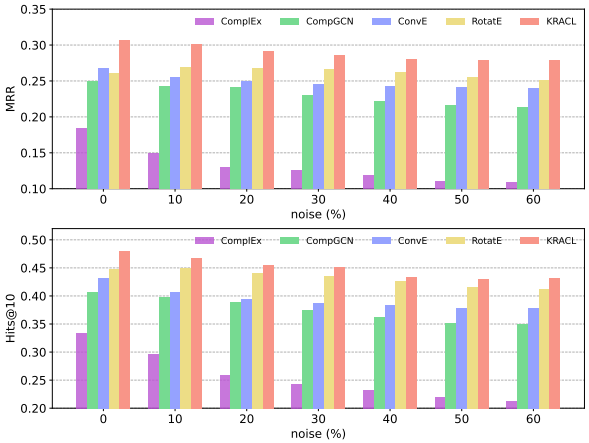

Figure 4: Link prediction performance on noisy knowledge graph of KRACL and some baseline models on the FB15k-237 dataset.

5 Conclusion

In this paper, we present KRACL model to alleviate the widespread sparsity problem in knowledge graphs for knowledge graph completion. First, KRACL leverages graph context by jointly aggregating neighbor entities and relations with the attention mechanism. Second, we propose a knowledge contrastive loss to introduce more negative, hence more feedback is provided to sparse entities.

The proposed KRACL effectively improves prediction performance on sparse entities in KGs. Extensive experiments on standard benchmark FB15k-237, WN18RR, NELL-995, Kinship, and UMLS show that KRACL improves consistently over competitive baseline models, especially on WN18RR and NELL-995 with many low in-degree entities.

6 Limitations

As most of the KGE models, our proposed KRACL model has the following limitations:

- **Scalability.** We take the initial input of KRACL as learnable embeddings, leading to the linear scaling of the model size with the number of entities in KGs. This design choice makes our model not feasible for large-scale knowledge graphs such as Wikidata5M (Wang et al., 2021) and WikiKG90M (Hu et al., 2021).

- **Inductive setting.** By the same token of learnable embeddings, our model is not capable of inductive learning. That is, the proposed KRACL cannot prediction unseen entities in knowledge graphs.

However, Galkin et al. (2021) also notice this problem and propose a node level tokenizer dubbed nodepiece. It greatly reduces the parameter of learnable embeddings and can enable inductive learning for KGE models as well. We defer combining our model with nodepiece and evaluating KRACL on large-scale knowledge graphs as our future work.

References

Mehdi Ali, Max Berrendorf, Charles Tapley Hoyt, Laurent Vermue, Sahand Sharifzadeh, Volker Tresp, and Jens Lehmann. 2021. Pykeen 1.0: A python library for training and evaluating knowledge graph embeddings. *J. Mach. Learn. Res.*, 22:82:1–82:6.

Ivana Balažević, Carl Allen, and Timothy M Hospedales. 2019. Hypernetwork knowledge graph embeddings. In *International Conference on Artificial Neural Networks*, pages 553–565. Springer.

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 Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. *Advances in neural information processing systems*, 26.

Shaked Brody, Uri Alon, and Eran Yahav. 2021. How attentive are graph attention networks? *CoRR*, abs/2105.14491.

Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Aréchiga, and Tengyu Ma. 2019. Learning imbalanced datasets with label-distribution-aware margin loss. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 1565–1576.

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.

Kewei Cheng, Ziqing Yang, Ming Zhang, and Yizhou Sun. 2021. Uniker: A unified framework for combining embedding and definite horn rule reasoning for knowledge graph inference. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 9753–9771. Association for Computational Linguistics.

Sarkar Snigdha Sarathi Das, Arzoo Katiyar, Rebecca J. Passonneau, and Rui Zhang. 2022. Container: Few-shot named entity recognition via contrastive learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 6338–6353. Association for Computational Linguistics.

Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. 2018. Convolutional 2d knowledge graph embeddings. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Xin Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmann, Shaohua Sun, and Wei Zhang. 2014. Knowledge vault: a web-scale approach to probabilistic knowledge fusion. In *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014*, pages 601–610. ACM.

William Falcon and The PyTorch Lightning team. 2019. PyTorch Lightning.

Yuan Fang, Kingsley Kuan, Jie Lin, Cheston Tan, and Vijay Chandrasekhar. 2017. Object detection meets knowledge graphs. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, pages 1661–1667. ijcai.org.

Matthias Fey and Jan Eric Lenssen. 2019. Fast graph representation learning with pytorch geometric. *CoRR*, abs/1903.02428.

Mikhail Galkin, Jiapeng Wu, Etienne G. Denis, and William L. Hamilton. 2021. Nodepiece: Compositional and parameter-efficient representations of large knowledge graphs. *CoRR*, abs/2106.12144.

Junyu Gao, Tianzhu Zhang, and Changsheng Xu. 2019. I know the relationships: Zero-shot action recognition via two-stream graph convolutional networks and knowledge graphs. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 8303–8311. AAAI Press.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 6894–6910. Association for Computational Linguistics.

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.

Raia Hadsell, Sumit Chopra, and Yann LeCun. 2006. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2006), 17-22 June 2006, New York, NY, USA*, pages 1735–1742. IEEE Computer Society.

Xu Han, Shulin Cao, Xin Lv, Yankai Lin, Zhiyuan Liu, Maosong Sun, and Juanzi Li. 2018. Openke: An open toolkit for knowledge embedding. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018: System Demonstrations, Brussels, Belgium, October 31 - November 4, 2018*, pages 139–144. Association for Computational Linguistics.

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. Computer Vision Foundation / IEEE.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015a. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pages 1026–1034. IEEE Computer Society.

Shizhu He, Kang Liu, Guoliang Ji, and Jun Zhao. 2015b. Learning to represent knowledge graphs with gaussian embedding. In *Proceedings of the 24th ACM International Conference on Information and Knowledge Management, CIKM 2015, Melbourne, VIC, Australia, October 19 - 23, 2015*, pages 623–632. ACM.

Weihua Hu, Matthias Fey, Hongyu Ren, Maho Nakata, Yuxiao Dong, and Jure Leskovec. 2021. OGB-LSC: A large-scale challenge for machine learning on graphs. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*.

Guoliang Ji, Shizhu He, Liheng Xu, Kang Liu, and Jun Zhao. 2015. Knowledge graph embedding via dynamic mapping matrix. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*, pages 687–696. The Association for Computer Linguistics.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Stanley Kok and Pedro Domingos. 2007. Statistical predicate invention. In *Proceedings of the 24th international conference on Machine learning*, pages 433–440.

Guohao Li, Chenxin Xiong, Ali K. Thabet, and Bernard Ghanem. 2020. Deepergcn: All you need to train deeper gcns. *CoRR*, abs/2006.07739.

Shuang Liang, Jie Shao, Dongyang Zhang, Jiasheng Zhang, and Bin Cui. 2021. Drgi: Deep relational graph infomax for knowledge graph completion. *IEEE Transactions on Knowledge and Data Engineering*.

Xi Victoria Lin, Richard Socher, and Caiming Xiong. 2018. Multi-hop knowledge graph reasoning with reward shaping. *arXiv preprint arXiv:1808.10568*.

Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. 2015. Learning entity and relation embeddings for knowledge graph completion. In *Twenty-ninth AAAI conference on artificial intelligence*.

Weiyang Liu, Yandong Wen, Zhiding Yu, and Meng Yang. 2016. Large-margin softmax loss for convolutional neural networks. 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 507–516. JMLR.org.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Tom Mitchell, William Cohen, Estevam Hruschka, Partha Talukdar, Bishan Yang, Justin Betteridge, Andrew Carlson, Bhavana Dalvi, Matt Gardner, Bryan Kisiel, et al. 2018. Never-ending learning. *Communications of the ACM*, 61(5):103–115.

Deepak Nathani, Jatin Chauhan, Charu Sharma, and Manohar Kaul. 2019. Learning attention-based embeddings for relation prediction in knowledge graphs. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 4710–4723. Association for Computational Linguistics.

Dai Quoc Nguyen, Tu Dinh Nguyen, Dat Quoc Nguyen, and Dinh Phung. 2017. A novel embedding model for knowledge base completion based on convolutional neural network. *arXiv preprint arXiv:1712.02121*.

Maximilian Nickel, Lorenzo Rosasco, and Tomaso A. Poggio. 2016. Holographic embeddings of knowledge graphs. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA*, pages 1955–1961. AAAI Press.

Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. 2011. A three-way model for collective learning on multi-relational data. In *Icml*.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in pytorch.

Apoorv Saxena, Adrian Kochsiek, and Rainer Gemulla. 2022. Sequence-to-sequence knowledge graph completion and question answering. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 2814–2828. Association for Computational Linguistics.

Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In *European semantic web conference*, pages 593–607. Springer.

Chao Shang, Yun Tang, Jing Huang, Jinbo Bi, Xiaodong He, and Bowen Zhou. 2019. End-to-end structure-aware convolutional networks for knowledge base completion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3060–3067.

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.

Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. 2019. Rotate: Knowledge graph embedding by relational rotation in complex space. *arXiv preprint arXiv:1902.10197*.

Yun Tang, Jing Huang, Guangtao Wang, Xiaodong He, and Bowen Zhou. 2020. Orthogonal relation transforms with graph context modeling for knowledge graph embedding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 2713–2722. Association for Computational Linguistics.

Yonglong Tian, Dilip Krishnan, and Phillip Isola. 2020. Contrastive multiview coding. In *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XI*, volume 12356 of *Lecture Notes in Computer Science*, pages 776–794. Springer.

Kristina Toutanova and Danqi Chen. 2015. Observed versus latent features for knowledge base and text inference. In *Proceedings of the 3rd workshop on continuous vector space models and their compositionality*, pages 57–66.

Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. 2016. Complex embeddings for simple link prediction. In *International conference on machine learning*, pages 2071–2080. PMLR.

Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9(11).

Shikhar Vashishth, Soumya Sanyal, Vikram Nitin, Nilesh Agrawal, and Partha P. Talukdar. 2020. Interacte: Improving convolution-based knowledge graph embeddings by increasing feature interactions. 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 3009–3016. AAAI Press.

Shikhar Vashishth, Soumya Sanyal, Vikram Nitin, and Partha Talukdar. 2019. Composition-based multi-relational graph convolutional networks. *arXiv preprint arXiv:1911.03082*.

Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph attention networks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Feng Wang and Huaping Liu. 2021. Understanding the behaviour of contrastive loss. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 2495–2504. Computer Vision Foundation / IEEE.

Xiaozhi Wang, Tianyu Gao, Zhaocheng Zhu, Zhengyan Zhang, Zhiyuan Liu, Juanzi Li, and Jian Tang. 2021. KEPLER: A unified model for knowledge embedding and pre-trained language representation. *Trans. Assoc. Comput. 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 AAAI Conference on Artificial Intelligence*, volume 28.

Junkang Wu, Wentao Shi, Xuezhi Cao, Jiawei Chen, Wenqiang Lei, Fuzheng Zhang, Wei Wu, and Xiangnan He. 2021a. Disenkgat: Knowledge graph embedding with disentangled graph attention network. In *CIKM '21: The 30th ACM International Conference on Information and Knowledge Management, Virtual Event, Queensland, Australia, November 1 - 5, 2021*, pages 2140–2149. ACM.

Sixing Wu, Ying Li, Minghui Wang, Dawei Zhang, Yang Zhou, and Zhonghai Wu. 2021b. More is better: Enhancing open-domain dialogue generation via multi-source heterogeneous knowledge. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 2286–2300. Association for Computational Linguistics.

Han Xiao, Minlie Huang, and Xiaoyan Zhu. 2016a. From one point to a manifold: Knowledge graph embedding for precise link prediction. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, pages 1315–1321. IJCAI/AAAI Press.

Han Xiao, Minlie Huang, and Xiaoyan Zhu. 2016b. Transg : A generative model for knowledge graph embedding. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics.

Wenhan Xiong, Thien Hoang, and William Yang Wang. 2017. Deeppath: A reinforcement learning method for knowledge graph reasoning. *arXiv preprint arXiv:1707.06690*.

Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. 2014. Embedding entities and relations for learning and inference in knowledge bases. *arXiv preprint arXiv:1412.6575*.

Michihiro Yasunaga, Hongyu Ren, Antoine Bosselut, Percy Liang, and Jure Leskovec. 2021. QA-GNN: reasoning with language models and knowledge graphs for question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 535–546. Association for Computational Linguistics.

Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. 2020. Graph contrastive learning with augmentations. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Wenqian Zhang, Shangbin Feng, Zilong Chen, Zhenyu Lei, Jundong Li, and Minnan Luo. 2022a. KCD: knowledge walks and textual cues enhanced political perspective detection in news media. *CoRR*, abs/2204.04046.

Yuhao Zhang, Hongji Zhu, Yongliang Wang, Nan Xu, Xiaobo Li, and Binqiang Zhao. 2022b. A contrastive framework for learning sentence representations from pairwise and triple-wise perspective in angular space. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 4892–4903. Association for Computational Linguistics.

Jinfeng Zhou, Bo Wang, Ruifang He, and Yuexian Hou. 2021. CRFR: improving conversational recommender systems via flexible fragments reasoning on knowledge graphs. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 4324–4334. Association for Computational Linguistics.

Sijin Zhou, Xinyi Dai, Haokun Chen, Weinan Zhang, Kan Ren, Ruiming Tang, Xiuqiang He, and Yong Yu. 2020. Interactive recommender system via knowledge graph-enhanced reinforcement learning. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 179–188.

Yanqiao Zhu, Yichen Xu, Feng Yu, Qiang Liu, Shu Wu, and Liang Wang. 2021. Graph contrastive learning with adaptive augmentation. In *WWW '21: The Web Conference 2021, Virtual Event / Ljubljana, Slovenia, April 19-23, 2021*, pages 2069–2080. ACM / IW3C2.

A Dataset Details

In this section, we provide the details of the different datasets used in our experiments.

- **FB15k-237** (Toutanova and Chen, 2015) is a subset of FB15k (Bordes et al., 2013), which contains knowledge base describing facts about the real world and is extracted from FreeBase (Bollacker et al., 2008). Different from FB15k, it removes all the reverse relations to prevent test data leakage.
- **WN18RR** (Dettmers et al., 2018) is a subset of the WordNet (Miller, 1995) containing lecxical relation between words. Similar to FB15k-237, WN18RR also removes the reverse relations to avoid test data leakage.
- **NELL-995** (Xiong et al., 2017) is a subset of the 995-th iteration of NELL system. From Table 6 we can see that it is much sparser than other datasets.
- **Kinship** (Lin et al., 2018) contains a set of triples that explains the kinship relationships among members of the Alyawarra tribe from Central Australia. It is an integral part of aboriginal across Australia with regard to marriages between aboriginal people.
- **UMLS** (Kok and Domingos, 2007) is a knowledge base that brings together many health and biomedical vocabularies and standards to enable the interoperability between computer systems.

B Baseline Details

We compare the proposed KRACL model with the following baseline models and reproduce their results using OpenKE (Han et al., 2018) and Pykeen (Ali et al., 2021) library.

- **TransE** (Bordes et al., 2013) is the most representative KGE model with the assumption that the superposition of head and relation embedding is close to tail embedding.
- **DistMult** (Yang et al., 2014) is a matrix factorization model that uses a bilinear scoring function.
- **ComplEx** (Trouillon et al., 2016) is a matrix factorization model that is embedded in complex space.
- **RotatE** (Sun et al., 2019) is a translational model that maps relations embeddings as rotation operation in complex space.
- **ConvE** (Dettmers et al., 2018) is a CNN-based model that adopts 2D convolution neural network to extract semantic information between entities and relations.
- **ConvKB** (Nguyen et al., 2017) is a CNN-based model that performs 1D convolution on triple embeddings for scoring.
- **HypER** (Balažević et al., 2019) is a CNN-based model that uses hypernetwork to construct relational convolution kernel.
- **R-GCN** (Schlichtkrull et al., 2018) is a GNN-based model that extends GCN to relational data. Specifically, it aggregate message from different relations with different projection matrix.
- **KBAT** (Nathani et al., 2019) is a GNN-based model that introduces attention mechanism to learn the importance of neighboring nodes and takes advantage of multi-hop neighbors.
- **CompGCN** (Vashishth et al., 2019) is a GNN-based model that jointly aggregates entity and relation embeddings and score triples with with a decoder such as TransE, DistMult, and ConvE.
- **DisenKGAT** (Wu et al., 2021a) is a GNN-based model that proposes to leverage micro-disentanglement and macro-disentanglement for representative embeddings.

C Relation Category Details

Following Wang et al. (2014), for each relation r, we compute the average number of tails per head and the average number of head per tail, denoted as $tphr$ and $hptr$, respectively. If $tphr < 1.5$ and $hptr < 1.5$, r is treated as one-to-one (1-1); if $tphr < 1.5$ and $hptr \geq 1.5$, r is treated as many-to-one (N-1); if $tphr \geq 1.5$ and $hptr < 1.5$, r is treated as one-to-many (1-N); if $tphr \geq 1.5$ and $hptr \geq 1.5$, r is treated as a many-to-many (N-N).

D Visualization of Entity Representations

To examine the quality of learned representations, we visualize the entity embeddings . Given a link

Dataset	#Ent.	#Rel.	#Edge			#In-degree	
			Train	Valid	Test	Avg.	Med.
FB15k-237	14,541	237	272,115	17,535	20,466	18.76	8
WN18RR	40,943	11	86,835	3,034	3,134	2.14	1
NELL-995	75,492	200	149,678	543	3,992	2.01	0
Kinship	104	25	8,544	1,068	1,074	82.15	82
UMLS	135	46	5,216	652	661	38.63	20

Table 6: Benchmark statistics.

Dec./Proj. (=X) $\rightarrow$	**TransE**		**DistMult**		**RotatE**		**ConvE**	
Methods $\downarrow$	MRR	H@10	MRR	H@10	MRR	H@10	MRR	H@10
X	.279	.441	.241	.419	.338	.533	.325	.501
X+R-GCN	.281	.469	.324	.499	.295	.457	.342	.525
X+W-GCN	.264	.444	.324	.504	.272	.430	.244	.525
X+CompGCN (Sub)	.335	.514	.336	.513	.290	.453	.352	.530
X+CompGCN (Mult)	.337	.515	.338	.518	.296	.456	.353	.532
X+CompGCN (Rot)	.271	.447	.289	.448	.296	.461	.325	.506
X+CompGCN (Corr)	.336	.518	.335	.514	.294	.459	.355	.535
X+KRAT (Sub)	**.341**	.523	.343	.525	.345	.527	.356	.542
X+KRAT (Mult)	.340	.523	**.345**	**.526**	.346	**.528**	.358	.546
X+KRAT (Rot)	.339	.522	**.345**	.524	**.348**	.527	.359	.544
X+KRAT (Corr)	.340	**.524**	.343	**.526**	.345	.526	**.360**	**.548**

Table 7: Performance of link prediction on FB15k-237 dataset. Following Vashishth et al. (2019), X+M (Y) denotes that M is the GNN backbone to obtain entity and relation embeddings and X is the scoring function or projection head in this work, Y denotes the fusion operator between entity and relation embeddings. The best scores across all settings are highlighted by $\boxed{\cdot}$.

Model	**UMLS**			
	MRR	MR	H@10	H@1
TransE	.615	3.6	.945	.391
DistMult	.164	18.8	.403	.061
ComplEx	.844	2.47	.967	<u>.765</u>
RotatE	.822	2.1	.969	.703
ConvE	.836	3.2	.946	.764
ConvKB	.782	<u>1.61</u>	.986	.593
SACN	<u>.856</u>	1.7	.985	.764
R-GCN	.481	7.8	.835	.318
KBAT	.818	1.855	<u>.987</u>	.711
KRACL	**.904**	**1.38**	**.995**	**.831**

Table 8: Link prediction performance of KRACL and baseline models on the UMLS dataset. The best score is in **bold** and the second best score is <u>underlined</u>.

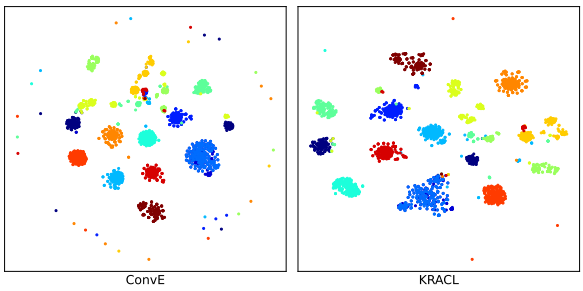

Figure 5: Visualization of tail entities in ConvE and KRACL with T-SNE.

prediction task $(s, r, ?)$, we select queries $(s, r, ?)$ that have the same answers and visualize their predictions with T-SNE (Van der Maaten and Hinton, 2008). As is shown in Figure 5, our model shows higher level of collocation for entities, which indicates that our KRACL framework learns high-quality representations for entities and relations.

E Computation Details

Our proposed KRACL model's learnable parameters and computational budget are listed in Table 10. We train our KRACL model on one RTX 3090 GPU with 24GB memory.

For main results shown in Table 1 and 2, we adjust the hyperparameters based on the performance

Hyperarameter	FB15k-237	WN18RR	NELL-995	Kinship	UMLS
Entity dim d_e	200	200	200	200	200
Relation dim d_r	200	200	200	200	200
Batch size	2048	2048	2048	1024	1024
Learning rate	10^{-3}	10^{-3}	10^{-3}	3×10^{-4}	5×10^{-4}
Epochs	1500	1000	1000	1000	1000
GNN layers	1	2	2	2	2
Encoder dropout	0.1	0.2	0.2	0.2	0.2
Temperature τ	0.07	0.07	0.07	0.1	0.1
Optimizer	AdamW	AdamW	AdamW	AdamW	AdamW

Table 9: Hyperparameter settings of KRACL across various benchmark datasets. We find our hyperparameter settings robust across all datasets and all hyperparameters are chosen by the performance on the validation set.

Dataset	Parameters	GPU hours
FB15k-237	13.3M	9.5
WN18RR	18.6M	4.5
NELL-995	25.9M	10.5
Kinship	10.4M	0.7
UMLS	10.4M	0.5

Table 10: Number of parameters in the KRACL model and GPU hours for training on selected datasets.

on validation set and report the best results on the test set. For other experiments, we present the performance of a single run.

F Fusion Operator Details

- **Rotation**: $\phi(\boldsymbol{h}_s, \boldsymbol{h_r}) = \boldsymbol{h}_s \circ \boldsymbol{h}_r$
  For each dimention i, $e[2i]$ and $e[2i+1]$ are corresponding real and imaginary components. Given the subject embedding e_s and relation transform embedding θ_r, the rotation projection is formulated as

$$\begin{bmatrix} (\boldsymbol{h}_s \circ \boldsymbol{h}_r)[2i] \\ (\boldsymbol{h}_s \circ \boldsymbol{h}_r)[2i+1] \end{bmatrix} = \begin{bmatrix} \cos \boldsymbol{h}_r(i) & -\sin \boldsymbol{h}_r(i) \\ \sin \boldsymbol{h}_r(i) & \cos \boldsymbol{h}_r(i) \end{bmatrix} \begin{bmatrix} \boldsymbol{h}_s[2i] \\ \boldsymbol{h}_s[2i+1] \end{bmatrix}, \tag{10}$$

  where θ_r is learnable parameter corresponding to relation type r, $\hat{h_o}$ denotes the projected object embedding after rotation.

- **Circular-correlation**: $\phi(\boldsymbol{h}_s, \boldsymbol{h_r}) = \boldsymbol{h}_s \star \boldsymbol{h}_r$
  Taken from Nickel et al. (2016), the circular-correlation operator is formulated as

$$(h_s \star h_r)[k] = \sum_{i=0}^{d-1} h_s[i] \cdot h_r[(k+i) \ mod \ d], \tag{11}$$

where d is the dimension of entity and relation embeddings, mod denotes the modulo operation. The circular-correlation operator can discriminate the direction of relation because of its non-commutative property.