Knowledge Graph Completion with Pre-trained Multimodal Transformer and Twins Negative Sampling

Yichi Zhang
3180103772@zju.edu.cn
College of Computer Science and Technology, Zhejiang University
Hangzhou, Zhejiang, China

Wen Zhang*
wenzhang2015@zju.edu.cn
College of Software Technology, Zhejiang University
Hangzhou, Zhejiang, China

ABSTRACT

Knowledge graphs (KGs) that modeling the world knowledge as structural triples are inevitably incomplete. Such problems still exist for multimodal knowledge graphs (MMKGs). Thus, knowledge graph completion (KGC) is of great importance to predict the missing triples in the existing KGs. As for the existing KGC methods, embedding-based methods rely on manual design to leverage multimodal information while finetune-based approaches are not superior to embedding-based methods in link prediction. To address these problems, we propose a **V**isual**B**ERT-enhanced **K**nowledge **G**raph **C**ompletion model (VBKGC for short). VBKGC could capture deeply fused multimodal information for entities and integrate them into the KGC model. Besides, we achieve the co-design of the KGC model and negative sampling by designing a new negative sampling strategy called twins negative sampling. Twins negative sampling is suitable for multimodal scenarios and could align different embeddings for entities. We conduct extensive experiments to show the outstanding performance of VBKGC on the link prediction task and make further exploration of VBKGC.

CCS CONCEPTS

• **Computing methodologies** → **Knowledge representation and reasoning**.

KEYWORDS

Knowledge Graph, Multimodal Knowledge Graph, Knowledge Graph Embedding, Negative Sampling

ACM Reference Format:
Yichi Zhang and Wen Zhang. 2022. Knowledge Graph Completion with Pre-trained Multimodal Transformer and Twins Negative Sampling. In *Proceedings of August 14–18, 2022 (KDD-UC '22).* ACM, New York, NY, USA, 8 pages. https://doi.org/XXXXXXX.XXXXXXX

*Corresponding Author.

KDD-UC '22, August 14–18, 2022, Washington, DC, USA

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Knowledge graphs (KGs) represent world knowledge as structured triples in the form of (head entity, relation, tail entity), (h, r, t) for short, which means entity h and t have a relation r. KGs contain a wealth of structural information of world knowledge and have become the infrastructure of AI research and can benefit a lot of tasks like recommendation systems [36], language modeling [13] and question answering [33].

Multimodal knowledge graphs (MMKGs) [14] are KGs containing a wealth of modal information (images and text), which greatly enhances the expressiveness of the KGs. How to leverage the modal information in multimodal knowledge graphs is a current research hotspot in KG-related research.

However, KGs and of course MMKGs are far from complete. They contain only the knowledge we have observed while plenty of triples are not discovered among the entities and relations. Therefore, knowledge graph completion (KGC) is an important task in KG research which aims to discover the missing triples in KGs. As for MMKGs, the approaches to achieve multimodal knowledge graph completion (MMKGC) can be divided into two main categories: embedding-based approaches [15, 18, 30] and finetune-based approaches. Embedding-based approaches follow the paradigm of knowledge graph embedding (KGE) which embed the entities and relations into a low-dimensional vector space and define a score function to estimate the plausibility of triples. We could call the embedding-based approaches multimodal knowledge graph embedding (MMKGE) as multimodal information in the KGs is also considered in the embedding model. finetune-based approaches [10, 25, 32] usually employ pre-trained language models like BERT [5] and encode the triples with their textual descriptions. Then the models are finetuned with the KG-related tasks like triple classification [32] and relation prediction [10]. The triple scores are based on the output of the BERT model.

Although existing methods mentioned above have made strides in MMKGC, these methods face the following problems. (1) For the embedding-based methods, their ability to utilize multimodal information about entities is insufficient. The extraction and fusion of multimodal information are highly dependent on the manual design and need more work to find the most suitable method for each dataset. For example, [28] propose three different methods to achieve modal fusion and search for the best strategy on two datasets. (2) For the finetune-based approaches, though they leverage the textual information by fine-tuning the pre-trained model, the inference speed on the test set is unbearably slow [25] due to the deep architecture of the pre-trained model and rank-based evaluation protocol for KGs. The evaluation results of them are still

not significantly better than embedding-based methods either. (3) Design of negative sampling is ignored in all of the approaches. Existing methods just apply the normal negative sampling for model training, which might not be suitable for the multimodal scenario. For the multiple embeddings in the multimodal scenario, aligning them is also important. Normal negative sampling is entity-level and has no such ability.

To address the problems mentioned above, we propose a multimodal knowledge graph completion model called **V**isual**B**ERT-enhanced **K**nowledge **G**raph Completion model (VBKGC for short). VBKGC is an embedding-based model which employs a pre-trained multimodal transformer model (VisualBERT [11] for example) to extract deeply fused multimodal feature which is free of finetuning and have a fast inference speed like many other embedding-based methods. Besides, we achieve the co-design of the MMKGE model and negative sample strategy. We propose a negative sample strategy called twins negative sampling for MMKGE. Twins negative sampling could align the different embeddings of each entity during training and achieve better performance on link prediction tasks.

In general, our contributions in this paper can be summarized as follows:

- We propose an MMKGC model called VBKGC, which is an embedding-based model and employs VisualBERT as a multimodal encoder to capture the deeply fused multimodal features of entities. It is a universal approach and needs no more manual design for modal feature extraction and fusion. Besides, VBKGC has fast inference speed.
- We achieve the co-design of the model and negative sampling for KGC. We propose a new negative sampling method called twins negative sampling for multimodal scenarios. Twins negative sampling could align the structural and multimodal embeddings for entities to perform better on the link prediction task.
- We conduct comprehensive experiments on link prediction tasks with two benchmark datasets. We make further exploration of VBKGC model and twins negative sampling.

2 RELATED WORKS

2.1 Knowledge Graph Embedding

Knowledge graph embedding (KGE) [26] aims to embed entities and relations of KGs into a low-dimensional continuous vector space and measure the plausibility of triples by a well-defined score function, which is a popular topic in KG-related research.

Existing KGE models are diversified. Translation-based methods like TransE [2] and TransH [29] modeling the relation in each triple as a translation from head entity to tail entity. Sematic-based methods like DistMult [31] and ComplEx [22] apply similarity-based score function to modeling the triples. Other method such as RotatE [21] and ConE [38] also modeling triples with various mathematical structures. Convolutional neural networks and graph neural networks are also employed in some KGE models [4, 9, 19, 23], which play a role as feature encoders. Rule-enhanced methods [34, 35] integrate rule learning in KGE for better performance and explainability.

Meanwhile, negative sampling (NS) [2, 29] is a key technology for KGE. It would generate negative triples and teach the KGE model to distinguish between positive and negative triples. Many researchers propose better negative sampling strategies. GAN-based methods [3, 27] apply GAN [7] to generate hard negative triples. NSCaching [37] simply stores the high-quality negative triples with cache during training. Other methods like SANS [1] and CAKE [16] leverage information from original KGs and sample high-quality negative triples.

However, existing NS methods are usually designed for general KGE. In this context, general KGE means KGE with no extra information outside the triplet structure. In the paragraphs that follow, we still use such a concept. In the multimodal scenario, each entity in a KG might have multiple embeddings (structural embedding and multimodal embedding for example) rather than only one structural embedding, which means the aligning the multiple embeddings is also of great significance. Unfortunately, the existing NS methods do not have this feature and a new NS method urgently needs to be proposed.

2.2 Multimodal Knowledge Graph Completion

Multimodal knowledge graph completion (MMKGC) is an important task for MMKGs, which would predict the missing triples in MMKGs with multimodal information of entities and relations. Previous methods of MMKG could be roughly divided into two categories: embedding-based approaches and finetune-based approaches.

Embedding-based approaches [15, 18, 30] can be called multimodal knowledge graph embedding (MMKGE) as well. It follows the paradigm of general KGE and represent each entity and relation with several embeddings. To leverage the multimodal information, these approaches extract multimodal features with pre-trained models like VGG [20] and GloVe [17], then the multimodal features would be fused into the multimodal embeddings of entities. These methods are backward in extracting multimodal information, rely on a lot of manual design and have poor ability to represent the extracted modal information.

Finetune-based approaches [10, 25, 32] would employ pre-trained models like BERT [5] to score the triples directly instead of training entity and relation embeddings. KG-BERT [32] extend the triples into text sequences as inputs of BERT and then finetune BERT with the triple classification task. MTL-KGC [10] is a multi-task version of KG-BERT. StAR [25] apply siamese-style textual encoder to speed up the inference stage. These approaches would are slower and less accurate than the traditional methods of inference due to the rank-based evaluation of KGs.

3 DEFINITION

A MMKG can be denoted as $\mathcal{G}_M = (\mathcal{E}, \mathcal{R}, \mathcal{I}, \mathcal{D}, \mathcal{T})$, where $\mathcal{E}$ is the entity set,$\mathcal{R}$ is the relation set, $\mathcal{I}$ is the image set, $\mathcal{D}$ is the text description set, $\mathcal{T} = \{(h, r, t) \mid h, t \in \mathcal{E}, r \in \mathcal{R}\}$ is the triple set where h, r, t are the head entity, relation, and tail entitiy of a triple. For each entity $e_i \in \mathcal{E}$, it has a textual description $d_i \in \mathcal{D}$. Each textual description d_i consists of several words, denoted as $d_i = (w_1, w_2, \ldots, w_n)$, where $w_j \in \mathcal{V}$ and $\mathcal{V}$ is the vocabulary of the MMKG $\mathcal{G}_M$. Besides, each entity e_i might have 0 to any numbers of images in $\mathcal{I}$, the images of e_i is denoted as set I_i.

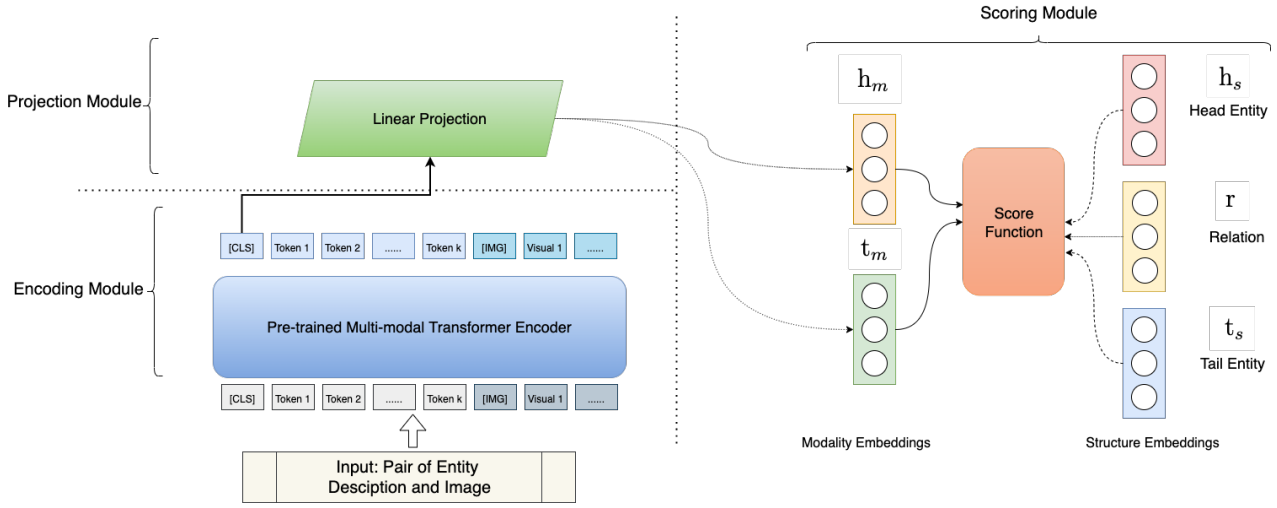

Figure 1: Model architecture of VBKGE, our model consists of three modules: encoding module, projection module, and scoring module.

We denote $\mathbf{e}_s$ and $\mathbf{e}_m$ as the structural embedding and multimodal embedding for an entity e, respectively. Therefore, the entity e can be represented by two embedding vectors $\mathbf{e}_s, \mathbf{e}_m$. Besides, we denote $\mathbf{r}$ as the embedding of relation r.

4 METHOD

The model architecture of our model VBKGE is shown in Figure 1. VBKGE has three modules: encoding module, projection module, and scoring module. The detailed design of each module would be shown below.

4.1 Encoding Module

The main role of the encoding module is to encode entities based on their textual and image information. In this paper, we use VisualBERT [11], a pre-trained multimodal transformer [24] model as our multimodal encoder. VisualBERT learned the knowledge about how to achieve modal alignment and fusion, which is implicitly stored in the parameters. Thus, we could use pre-trained VisualBERT to obtain deeply integrated multimodal features.

For each entity $e_i \in \mathcal{E}$, the textual description of e_i is $d_i = (w_1, w_2, \ldots, w_n)$ and the image set of e_i is I_i. We use VGG [20] to extract the visual feature and process the images of e_i into several visual tokens $(v_1, v_2, \ldots, v_m)$. The inputs $\hat{d}_i$ of VisualBERT model includes both word tokens and visual tokens:

$$\hat{d}_i = ([\text{CLS}], w_1, \ldots, w_n, [\text{SEP}], v_1, \ldots, v_m, [\text{SEP}]) \tag{1}$$

We add several special tokens like [CLS] and [SEP] following the original VisualBERT paper. The output of VisualBERT is:

$$\textbf{VisualBERT}(\hat{d}_i) = (\mathbf{h}_{\text{CLS}}, \mathbf{h}_{w1}, \ldots, \mathbf{h}_{\text{SEP}}, \mathbf{h}_{v1}, \ldots, \mathbf{h}_{\text{SEP}}) \tag{2}$$

The hidden state of [CLS] is employed as the initial multimodal feature of each entity e_i, denoted as $\mathbf{h}_i$.

4.2 Projection Module

The main function of the projection module is to project the modal features of entities into the same representation space of structural embeddings. As the modal features and structured embeddings are heterogeneous, they could not participate in triple scoring together. Thus, we apply a projection matrix $\mathbf{W}$ and obtain the multimodal embedding $\mathbf{e}_{im}$ of each entity e_i by linear projection:

$$\mathbf{e}_{im} = \mathbf{W}\mathbf{h}_{\text{CLS}} \tag{3}$$

4.3 Scoring Module

Scoring module would define a score function and estimate the plausibility of each triple. The general priciple of score function is to give higher scores for positive triples and lower scores for negative ones.

For each triple $(h, r, t) \in \mathcal{T}$, the score function $\mathcal{F}$ of VBKGE can be divided into five different parts:

$$\mathcal{F}_{ss} = f(\mathbf{h}_s, \mathbf{r}, \mathbf{t}_s) \tag{4}$$

$$\mathcal{F}_{mm} = f(\mathbf{h}_m, \mathbf{r}, \mathbf{t}_m) \tag{5}$$

$$\mathcal{F}_{sm} = f(\mathbf{h}_s, \mathbf{r}, \mathbf{t}_m) \tag{6}$$

$$\mathcal{F}_{ms} = f(\mathbf{h}_m, \mathbf{r}, \mathbf{t}_s) \tag{7}$$

$$\mathcal{F}_{all} = f(\mathbf{h}_s + \mathbf{h}_m, \mathbf{r}, \mathbf{t}_s + \mathbf{t}_m) \tag{8}$$

$$\mathcal{F}(h, r, t) = \mathcal{F}_{ss} + \mathcal{F}_{mm} + \mathcal{F}_{sm} + \mathcal{F}_{ms} + \mathcal{F}_{all} \tag{9}$$

In VBKGE, we apply TransE as function f, which can be denoted as:

$$f(\mathbf{h}, \mathbf{r}, \mathbf{t}) = -||\mathbf{h} + \mathbf{r} - \mathbf{t}||_p \tag{10}$$

In our scoring function $\mathcal{F}$, the multimodal embeddings and structure embeddings of entities could interact fully with each other as we define multiple score functions. The five score functions could be divided into two parts, unimodal scores, and multimodal scores.

Unimodal scores are calculated by one kind of embeddings (structural or multimodal) while multimodal scores need both. Thus, the overall score function $\mathcal{F}$ can be expressed in another way:

$$\mathcal{F}_{unimodal} = \mathcal{F}_{ss} + \mathcal{F}_{mm} \tag{11}$$

$$\mathcal{F}_{multimodal} = \mathcal{F}_{sm} + \mathcal{F}_{ms} + \mathcal{F}_{all} \tag{12}$$

$$\mathcal{F}(h,r,t) = \mathcal{F}_{unimodal} + \mathcal{F}_{multimodal} \tag{13}$$

4.4 Training Objective And Negative Sampling Strategy

4.4.1 Contrastive Training Objective. As a general paradigm, KGE models would give higher scores for positive triples and lower scores for negative ones. We first generate a negative triple set by randomly replacing the head or tail entity in each positive triple, which is called negative sampling. The negative triple set can be denoted as:

$$\begin{aligned}\mathcal{T}' = &\left\{(h,r,t') \mid t' \in \mathcal{E} \wedge (h,r,t') \notin \mathcal{T}\right\} \\ &\cup \left\{(h',r,t) \mid h' \in \mathcal{E} \wedge (h,r,t') \notin \mathcal{T}\right\}\end{aligned} \tag{14}$$

where (h,r,t) is a positive triple. We genearate k negative samples for each postive triple.

Therefore, we apply a margin-rank loss for postive-negative contrast during training:

$$\mathcal{L} = \sum_{(h,r,t)\in\mathcal{T}} \max\left(\gamma - \mathcal{F}(h,r,t) + \frac{1}{k}\sum_{(h'_i,r'_i,t'_i)\in\mathcal{T}'} \mathcal{F}(h'_i,r'_i,t'_i), 0\right) \tag{15}$$

where γ is the margin.

4.4.2 Twins Negaitve Sampling. In this paper, we also propose a negative sampling called twins for better MKGE model training.

Traditional negative sampling in KGE is entity-level which would replace the whole head or tail entity to generate a negative triple. It works in general KGE as general KGE usually defines only one embedding for each entity. However, in MKGE models, each entity would have multiple embeddings such as structural and multimodal embeddings. In the multimodal scenario, entity-level negative sampling would replace all the embeddings of the selected entity, which implicitly assumes that different embeddings of this entity have been aligned. Therefore, such an assumption might be too strong for model training.

Hence, we propose a more fine-grained negative sampling strategy called twins to solve the problem. Twins negative sampling employs different negative sampling strategies for unimodal and multimodal parts of the model. That's why we call it twins. With twin negative sampling, the model could not only learn to discriminate the plausibility of triples (just like traditional negative sampling) but also align the different embeddings for each entity.

As for the unimodal scores, twins employs just normal negative sampling. The head or tail entity e in the positive triple is randomly replaced by another entity e'. For multimodal scores, however, we only sample negative multimodal features for the replaced entity. We still denote the sampled entity as e', but the structural and multimodal embeddings of e' are $\mathbf{e}_s, \mathbf{e}'_m$ while they are $\mathbf{e}'_s, \mathbf{e}'_m$ in normal negative sampling. By contrasting with the negative modal features, the model can further align the two kinds of embeddings. With such a fine-grained and modal-level negative sampling strategy for multimodal scores, the model could learn to align the embeddings for each entity during training.

5 EXPERIMENTS

Table 1: Statistical information of datasets.

Dataset	Entities	Relations	Train	Valid	Test
WN9	6555	9	11741	1337	1319
FB15K-237	14541	237	272115	17535	20466

In this section, we will report the experiment details including datasets, evaluation protocols, parameter settings, and the results. In addition to the conventional link prediction experiments, we have three exploratory questions:

(1) **Question 1** (Q1): Does twins negative sampling works?
(2) **Question 2** (Q2): Does our methods inference faster than finetune-based approaches?
(3) **Question 3** (Q3): Whether the design of each part of the model is valid?

Following the three questions, we would expolore more about VBKGC next.

5.1 Datasets

In our experiments, we employ two public benchmarks WN9 [30] and FB15K-237 [32]. The image resources of FB15K-237 is collected from [14]. The detailed information about the datasets is shown in Table 1.

5.2 Evaluation Protocols

Following classic KG research, we apply link prediction task and rank-based evaluation protocol. Given a correct triple, we rank it against all candidate triples with their scores. Both head and tail entity prediction would be applied in link prediction. The whole entity set $\mathcal{E}$ would be the candidate entity set.

We use mean reciprocal rank (MRR) and Hit@K (K=1,3,10) as evaluation metrics. They can be denoted as:

$$\text{MRR} = \frac{1}{2|\mathcal{T}_{test}|}\sum_{t\in\mathcal{T}_{test}}\left(\frac{1}{rank_{th}} + \frac{1}{rank_{tt}}\right) \tag{16}$$

$$\text{Hit@K} = \frac{1}{2|\mathcal{T}_{test}|}\sum_{t\in\mathcal{T}_{test}} \mathbf{1}(rank_{th} \le K) + \mathbf{1}(rank_{tt} \le K) \tag{17}$$

where $\mathcal{T}_{test}$ is the test triple set and $rank_{th}, rank_{tt}$ are predicted ranks of head/tail entity prediction for each test triple t.

Besides, all the metrics are in the filter setting [2]. It would remove the candidate triples which have already appeared in train and valid data.

5.3 Experiment Settings

For experiments, we set both structural embedding and multimodal embedding size $d_e = 128$ for each model. The dimension of multimodal features captured by the pre-trained VisualBERT model is $d_m = 768$. For those entities which have no image, we employ

Table 2: Expeirment results of the link prediction task. The baselines marked with * are our reproduction based on the original paper. The best results in each metric are bold and the second-best results are underlined. Some results of baselines that are hard for reproduction and have no results in origin paper are marked as -.

Method	WN9				FB15K-237			
	MRR	Hit@10	Hit@3	Hit@1	MRR	Hit@10	Hit@3	Hit@1
TransE*	0.766	0.912	0.885	0.641	0.261	0.437	0.291	0.173
IKRL*	0.433	0.938	0.849	0.011	0.268	0.449	0.301	0.177
TransAE	-	0.942	-	-	-	-	-	-
MTKRL*	0.354	**0.948**	0.651	0.112	-	-	-	-
KG-BERT	-	-	-	-	0.237	0.427	0.260	0.144
StAR(base)	-	-	-	-	0.296	**0.482**	0.322	0.205
VBKGC+Normal	0.749	0.919	0.901	0.592	0.299	0.477	0.331	0.210
VBKGC+Twins	**0.857**	0.922	**0.904**	**0.803**	**0.301**	0.478	**0.332**	**0.213**

Xavier initialization [6] for their visual features. We set the amount of negative sample $k = 16$.

During training, we divide each dataset into mini-batches and apply TransE [2] as base score functions f. We use default Adam optimizer for optimization and tune the hyper-parameters of our model with grid search. The number of batches is tuned in $\{100, 400\}$ The margin γ is tuned in $\{4.0, 6.0, 8.0, 10.0\}$ and learning rate is tuned in $\{2e-5, 1e-4, 5e-4, 1e-3\}$. The parameter settings are based on existing research findings [8, 30]. All the experiments are conducted on Nvidia GeForce 3090 GPUs.

As for baselines, we employ several embedding-based methods (TransE [2], IKRL [30], TransAE[28], MTKRL [15], all of them apply TransE as score function) and finetune-based approaches (KG-BERT [32], StAR [25]). For fair comparisions, we use the same embeddings dim for embedding-based approaches ($d = 128$) and reproduce some of the baselines with the same hyperparamters as original papers.

5.4 Link Prediction Results

The main experiment results on the link prediction task are shown in Table 2. We reproduce some classic baselines on the datasets and those results are marked with *. Some results of baselines that are hard for reproduction and have no results in origin paper are marked as -. We could visualize that our method VBKGC could perform better than baselines on most of the metrics, except Hit@10 on WN9.

Besides, the twins negative sampling could behave better than normal negative sampling in the multimodal scenario. The performance gains it brings are particularly noticeable in the WN9 dataset. Though twins negative sampling gets limited improvement on the FB15K237 dataset, we would make a further exploration in the analysis of Q1.

Another surprising conclusion that can be deduced from the experimental results is that VBKGC with twins negative sampling could perform precise reasoning. Compared with other embedding-based baselines ([2, 15, 30]), VBKGC with twins could achieve outstanding improvement on Hit@1 and MRR metrics. VBKGC with twins obtains nearly 25% (from 0.641 to 0.803) and 20% (from 0.177 to 0.213) on Hit@1 with WN9 and FB15K-237 respectively. Compared with finetune-based approaches, the improvements brought about by our model are also clearly perceptible.

5.5 Analysis of Q1: Negative Sampling

We could conclude that twins negative sampling could improve the link prediction performance from the previous results. In this section, we would dive deeper into the negative sampling in MMKGC and try to answer Q1.

We employ several state-of-the-art negative sampling methods for general KGE (NSCaching [37], NoSampling[12]) for comparison. We implement the VBKGC model based on their open-source code and conduct experiments with the same embedding dimension $d_e = 128$ on the WN9 dataset. We tuned the hyperparameters according to the original paper and the results are shown in Table 3.

Table 3: Expeirment results of different negative sampling methods on WN9. The best results in each metric are bold and the second best results are underlined.

	MRR	Hit@10	Hit@3	Hit@1
Normal	0.749	0.919	0.901	0.592
NSCaching	0.725	0.868	0.804	0.630
NoSampling	0.426	0.662	0.492	0.306
Twins	**0.857**	**0.922**	**0.904**	**0.803**

We could found that negative sampling methods for general KGE might not be acclimatized for the multimodal scenario as we make our best to tune hyperparameters for better results. They even get a marked regression on some metrics. The performance of twins negative sampling on the WN9 dataset exceeds the existing baselines in all aspects. It could align structural and multimodal embeddings to achieve better performance in the multimodal scenario.

Besides, as twins negative sampling gets limited improvement on FB15K237 dataset, we make a further exploration about this problem. We trained several VBKGC models with different amounts of negative samples for both normal and twins negative sampling on FB15K-237 dataset. The evaluation results are plotted as a line graph (Figure 2).

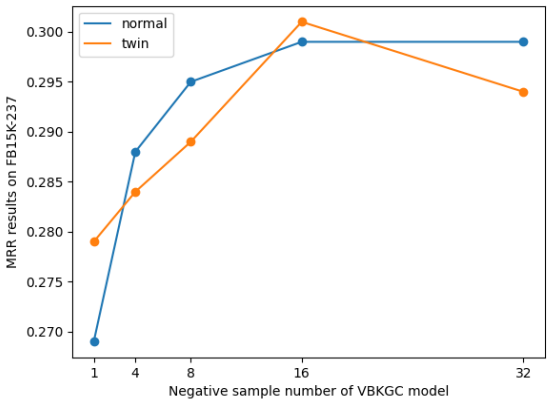

Figure 2: Link prediction performance (MRR) on FB15K-237 with different k (numbers of negative samples).

We could observe that when $k = 1$, twins negative sampling could perform better than normal negative sampling. But when k increases, the impact of the twins negative sampling on the VBKGC model seems to be less significant than normal negative sampling. Existing research shows that increasing the number of negative samples is also an effective means [21] to improve the model performance. It might be a better choice than twins negative sampling which aligns different embeddings in the FB15K-237 dataset. Nonetheless, it is still a good design for the multimodal scenario as it brings effective enhancement to the WN9 dataset, only it has yet more exploration.

5.6 Analysis of Q2: Inference Speed

To answer Q2, we employ several MMKGC models and measure the overall time they need to infer on the FB15K-237 test data. We plot the results of the measurements as a scatter plot shown in Figure 3.

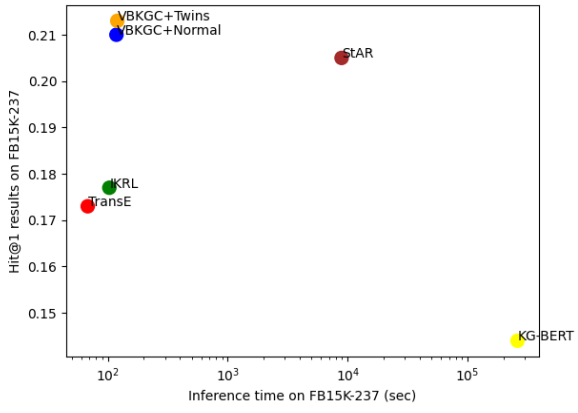

Figure 3: Inference time and MRR results of different models on FB15K-237 dataset.

From the figure we can learn that VBKGC could inference fast like many other embedding-based approaches (TransE [2], IKRL [30]). Compare with finetune-based methods (KG-BERT [32], StAR [25]), VBKGC could achieve both better link prediction performance and faster inference speed. Besides, VBKGC would perform a bit slower than other embedding-based methods as VBKGC employs a more complex score function than baselines.

5.7 Analysis of Q3: Ablation Study

To further prove the effects of different modules in VBKGC, we conduct the ablation study with six different settings of experiments (S1-S6). The settings of S1 to S6 are as follows: (1)(Encoding Module) S1 refers to the model with random multimodal embeddings without VisualBERT to verify the quality of multimodal features captured by VisualBERT. (2) (Scoring Module) S2 to S5 apply just several parts of the overall score function $\mathcal{F}(h, r, t)$. S2 only employs $\mathcal{F}_{ss}$ as a score function. S3 employs $\mathcal{F}_{all}$. S4 employs the unimodal scores $\mathcal{F}_{ss} + \mathcal{F}_{mm}$. S5 employs the multimodal scores $\mathcal{F}_{sm} + \mathcal{F}_{ms}$. (3) (Training Objective) S6 sets $k = 1$ and samples only 1 negative triple during training.

Table 4: Expeirment results of ablation study on FB15K-237. The exact meaning of S1 to S6 could be found in the main text.

	MRR	Hit@10	Hit@3	Hit@1
VBKGC	0.299	0.477	0.331	0.210
S1: w/o VisualBERT	0.226	0.409	0.261	0.132
S2: only $\mathcal{F}_{ss}$	0.147	0.242	0.153	0.097
S3: only $\mathcal{F}_{all}$	0.261	0.435	0.293	0.173
S4: only $\mathcal{F}_{ss} + \mathcal{F}_{mm}$	0.251	0.407	0.273	0.174
S5: only $\mathcal{F}_{sm} + \mathcal{F}_{ms}$	0.285	0.464	0.317	0.195
S6: only 1 negative	0.269	0.436	0.297	0.184

The results of the ablation study are shown in Table 4. We could find that all of S1 to S6 get worse performance than the full model (VBKGC with normal negative sampling). Thus, the design of each part of our model VBKGC is necessary to get better performance.

6 CONCLUSION

In this paper, we present an embedding-based model VBKGC for MMKGC, which employs VisualBERT as a multimodal feature encoder. We achieve co-design of both model and negative sampling by proposing twins negative sampling to align different embeddings for the multimodal scenario. VisualBERT extracts deeply fused multimodal information for better link prediction, which is free of finetuning and makes VBKGC inference fast and precise. Extensive experiment results on two datasets and link prediction tasks with three further explorations demonstrate the effectiveness of VBKGC.

In the future, we plan to 1) explore more effective ways to leverage multimodal information in knowledge graphs to benefit more kinds of in-KG and out-KG tasks; 2) find a more expressive architecture and try to pre-train KGs on it to capture the deep knowledge in KGs; 3) borrowing solutions from other multimodal machine learning tasks for multimodal knowledge graphs.

REFERENCES

[1] Kian Ahrabian, Aarash Feizi, Yasmin Salehi, William L Hamilton, and Avishek Joey Bose. 2020. Structure aware negative sampling in knowledge graphs. *arXiv preprint arXiv:2009.11355* (2020).

[2] 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 (2013).

[3] Liwei Cai and William Yang Wang. 2017. Kbgan: Adversarial learning for knowledge graph embeddings. *arXiv preprint arXiv:1711.04071* (2017).

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

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[6] Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2010, Chia Laguna Resort, Sardinia, Italy, May 13-15, 2010 (JMLR Proceedings, Vol. 9)*, Yee Whye Teh and D. Mike Titterington (Eds.). JMLR.org, 249–256. http://proceedings.mlr.press/v9/glorot10a.html

[7] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. *Advances in neural information processing systems* 27 (2014).

[8] 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*. 139–144.

[9] Xiaotian Jiang, Quan Wang, and Bin Wang. 2019. Adaptive convolution for multi-relational learning. 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)*. 978–987.

[10] 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*. 1737–1743.

[11] Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. 2019. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557* (2019).

[12] Zelong Li, Jianchao Ji, Zuohui Fu, Yingqiang Ge, Shuyuan Xu, Chong Chen, and Yongfeng Zhang. 2021. Efficient Non-Sampling Knowledge Graph Embedding. In *Proceedings of the Web Conference 2021*. 1727–1736.

[13] 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 *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 2901–2908.

[14] Ye Liu, Hui Li, Alberto Garcia-Duran, Mathias Niepert, Daniel Onoro-Rubio, and David S Rosenblum. 2019. MMKG: multi-modal knowledge graphs. In *European Semantic Web Conference*. Springer, 459–474.

[15] Hatem Mousselly-Sergieh, Teresa Botschen, Iryna Gurevych, and Stefan Roth. 2018. A multimodal translation-based approach for knowledge graph representation learning. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*. 225–234.

[16] Guanglin Niu, Bo Li, Yongfei Zhang, and Shiliang Pu. 2022. CAKE: A Scalable Commonsense-Aware Framework For Multi-View Knowledge Graph Completion. *arXiv preprint arXiv:2202.13785* (2022).

[17] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 1532–1543.

[18] Pouya Pezeshkpour, Liyan Chen, and Sameer Singh. 2018. Embedding multimodal relational data for knowledge base completion. *arXiv preprint arXiv:1809.01341* (2018).

[19] 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*. Springer, 593–607.

[20] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014).

[21] 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* (2019).

[22] Théo Trouillon, Christopher R Dance, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. 2017. Knowledge graph completion via complex tensor factorization. *arXiv preprint arXiv:1702.06879* (2017).

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

[24] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).

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

[26] Quan Wang, Zhendong Mao, Bin Wang, and Li Guo. 2017. Knowledge graph embedding: A survey of approaches and applications. *IEEE Transactions on Knowledge and Data Engineering* 29, 12 (2017), 2724–2743.

[27] Yansen Wang, Zhen Fan, and Carolyn Rose. 2020. Incorporating Multimodal Information in Open-Domain Web Keyphrase Extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, Online, 1790–1800. https://doi.org/10.18653/v1/2020.emnlp-main.140

[28] Zikang Wang, Linjing Li, Qiudan Li, and Daniel Zeng. 2019. Multimodal data enhanced representation learning for knowledge graphs. In *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 1–8.

[29] 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*, Vol. 28.

[30] Ruobing Xie, Zhiyuan Liu, Huanbo Luan, and Maosong Sun. 2016. Image-embodied knowledge representation learning. *arXiv preprint arXiv:1609.07028* (2016).

[31] 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* (2014).

[32] Liang Yao, Chengsheng Mao, and Yuan Luo. 2019. KG-BERT: BERT for knowledge graph completion. *arXiv preprint arXiv:1909.03193* (2019).

[33] Michihiro Yasunaga, Hongyu Ren, Antoine Bosselut, Percy Liang, and Jure Leskovec. 2021. Qa-gnn: Reasoning with language models and knowledge graphs for question answering. *arXiv preprint arXiv:2104.06378* (2021).

[34] Wen Zhang, Bibek Paudel, Liang Wang, Jiaoyan Chen, Hai Zhu, Wei Zhang, Abraham Bernstein, and Huajun Chen. 2019. Iteratively learning embeddings and rules for knowledge graph reasoning. In *The World Wide Web Conference*. 2366–2377.

[35] Wen Zhang, Bibek Paudel, Wei Zhang, Abraham Bernstein, and Huajun Chen. 2019. Interaction embeddings for prediction and explanation in knowledge graphs. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*. 96–104.

[36] Wen Zhang, Chi-Man Wong, Ganqiang Ye, Bo Wen, Wei Zhang, and Huajun Chen. 2021. Billion-scale Pre-trained E-commerce Product Knowledge Graph Model. In *2021 IEEE 37th International Conference on Data Engineering (ICDE)*. IEEE, 2476–2487.

[37] Yongqi Zhang, Quanming Yao, Yingxia Shao, and Lei Chen. 2019. NSCaching: simple and efficient negative sampling for knowledge graph embedding. In *2019 IEEE 35th International Conference on Data Engineering (ICDE)*. IEEE, 614–625.

[38] Zhanqiu Zhang, Jie Wang, Jiajun Chen, Shuiwang Ji, and Feng Wu. 2021. Cone: Cone embeddings for multi-hop reasoning over knowledge graphs. *Advances in Neural Information Processing Systems* 34 (2021).

A REPRODUCIBILITY

A.1 Implemention Details

We implement our methods based on the open-source library OpenKE. We utilize Pytorch to conduct experiments with one NVIDIA RTX 3090 GPU. Other information about parameter selection is mentioned in the previous section 5.3.

A.2 Datasets

The FB15k-237 dataset is available here. The WN9 datasets is available in here . We use the images of FB15K-237 released here.

A.3 Optimal parameters

The optimal hyper-parameters of our model VBKGC with both negative sampling methods on WN9 dataset is:

(1) number of batches: 100
(2) margin λ: 8
(3) learning rate: 2e-5

The optimal hyper-parameters on FB15K-237 dataset is:

(1) number of batches: 400
(2) margin λ: 6
(3) learning rate: 2e-5