Contrastive Language-Image Pre-Training with Knowledge Graphs

Anonymous Author(s)
Affiliation
Address
email

Abstract

Recent years have witnessed the vast development of large-scale pre-training frameworks that can extract multi-modal representations in a unified form and achieve promising performances when transferred to downstream tasks. Nevertheless, existing approaches mainly focus on pre-training with simple image-text pairs, while neglecting the semantic connections between concepts from different modalities. In this paper, we propose a knowledge-based pre-training framework, dubbed *Knowledge-CLIP*, that injects semantic information into the widely used CLIP model [41]. Through introducing knowledge-based objectives in the pre-training process and utilizing different types of knowledge graphs as training data, our model can semantically align the representations in vision and language, and also enhance the reasoning ability across scenarios and modalities. Extensive experiments on various vision-language downstream tasks demonstrate the effectiveness of Knowledge-CLIP comparing with the original CLIP and competitive baselines.

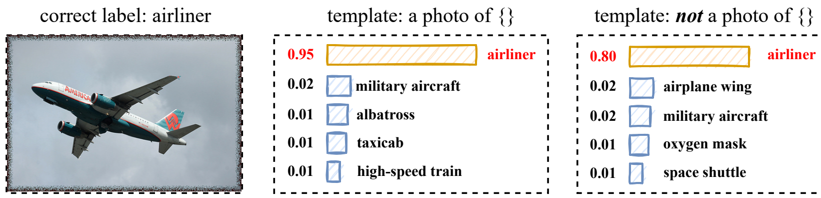
1 Introduction

Large-scale vision-language pre-training has attracted wide research interests in recent years [10, 29, 41, 76]. Different from training different models for each specific task, pre-trained models take the analogy of human biological intelligence system, trying to perceive the world from various data modalities and handle comprehensive tasks. Specifically, it aims to provide a unified inference paradigm that simultaneously learns representations for multi-modal data and can easily transfer to a variety of downstream tasks. Benefiting from the accessibility of massive image-text pairs from the web, the pre-training scheme can leverage a broader source of supervision, and effectively improves the model's generalization power.

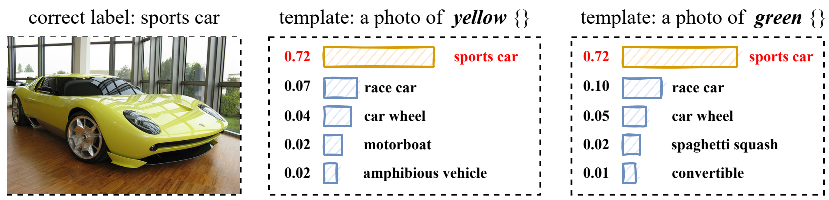
Early attempts on vision-language pre-training mainly focus on detecting objects in the images and aligning the corresponding word tokens with object regions [10, 31, 54]. Though effective, the entanglement with the concept of objects, and the additional resources for pre-trained object detectors impose restrictions on real-world applications. One of the pioneer works, CLIP [41], extends the scale of the pre-training dataset to 400 million image-text pairs, and learns representations by directly matching raw text with the corresponding image. Through a contrastive-based training scheme, CLIP learns visual concepts under a large vocabulary which greatly improves the model performances on various downstream tasks. Taking inspiration from CLIP, the following researches further extend the work from several perspectives, including data modality [76], downstream tasks [62], and training data efficiency [21, 47].

Although showing promising results, the current pre-training frameworks also suffer from limitations. Specifically, the data pairs for pre-training are organized in the simplest manner, where only the descriptions of *matched* and *unmatched* are used to represent the relation between a given image and text pair. This usually leads to a degenerated scenario, where the model tends to rely on the

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.



(a) Results on templates with opposite semantic description



(b) Results on templates with wrong semantic description

Figure 1: CLIP fails to accurately capture some fine-grained semantic information. When given opposite semantic descriptions, *e.g.,* adding 'not' in the template or describing an image with wrong color, CLIP tends to give similar distribution as the correct counterpart. Best view in color.

co-occurrence of inputs instead of their semantic meanings. We give a toy example in Fig. 1 by evaluating the zero-shot transfer performance of CLIP on the ImageNet dataset [11] with the templates 'a photo of a { }' and 'not a photo of a { }'. It is shown that the distributions of CLIP outputs under two templates are quite similar, suggesting that the current model fails to understand the semantic meaning of word tokens. As a result, the transferability of the model is restricted, and tends to show worse performances on tasks that require reasoning ability, *e.g.,* visual question answering.

To address the limitation of pre-trained models on semantic perceiving, we resort to the technique of knowledge graph, which has been widely studied in the field of natural language processing [8, 63]. Knowledge graph (KG) is a large-scale semantic network that comprises entities as nodes and semantic relations as edges. Through organizing data in a graph structure, knowledge graphs provide rich information on describing the relations between entities and enable a reasoning process through the whole graph. These advantages over regular-structured data are favorable on various tasks including question-answering [20, 74], relation prediction [32, 46] and knowledge reasoning [7, 64]. In recent years, knowledge graph has also been investigated in the field of computer vision, *e.g.*, scene graph [69], and the integration of both language and image [2]. This bridges the gap between different modalities in the knowledge graph, which inspires us to explore a new knowledge-based pre-training framework, and inject semantic information into simple image-text pairs.

In this paper, we propose a novel vision-language pre-training approach, dubbed *Knowledge-CLIP*, by constructing a knowledge-enhanced pre-training framework based on the widely used CLIP models. As illustrated in Fig. 2, we follow the structure of CLIP, and use two Transformer-based models as image and text encoders respectively. These two encoders take entities and relations in the knowledge graph as input and extract raw features for both entities and relations. Notably, entities can be in the form of image/text, while the relations are constantly described by language tokens. Then, a multi-modal Transformer encoder is adopted to fuse the entity features conditioned on their relations. In this way, the pre-trained model is pushed to concentrate on understanding semantic relations between visual and word concepts, thereby establishing strong semantic connections between vision and language modalities.

To additionally improve the training efficiency and avoid the massive computation cost in the pre-training procedure, we adopt a simple continuous learning strategy by training our model based on the pre-trained weights of CLIP. This provides a possibility of efficiently promoting the model performance of CLIP with low training resources.

We practically train our model on three knowledge graph datasets, namely Visual-Genome [27] (scene graph), ConceptNet [49] (language-based graph), and VisualSem [2] (multi-modal graph), and also adopt part of datasets from CLIP to avoid the model forgetting problem. With the knowledge-enhanced pre-training, Knowledge-CLIP achieves consistent improvements over the original CLIP models on various vision and language downstream tasks. Our model can also transfer to several graph-based tasks, including link prediction and entity classification, and achieve competitive results.

2 Related works

Large-scale pre-training. Large-scale pre-training framework has received wide concerns in recent years and shown promising results in the field of computer vision and natural language processing. GPT [42] is one of the pioneer works for language pre-training which optimizes the probability of output based on previous words in the sequence. BERT [13] adopts the masked language modeling technique and predicts the masked tokens conditioned on the unmasked ones.

Similarly, computer vision society also witnesses the development of pre-training models thanks to the emergence of large-scale image datasets. IGPT [6] proposes a generative pre-training technique and shows promising results on classification task. MAE [19] adopts a similar pre-training scheme as BERT and predicts the masked regions of an image with unmasked ones.

Multi-modal pre-training bears differences from the aforementioned frameworks and requires the alignment between various data modalities. Using enormous image-text pairs collected from Internet, vision-language models show significant improvements on various downstream tasks. Among these approaches, various pre-training scheme is adopted, including contrastive learning [1, 30, 34], masked language modeling [50, 55], and masked region modeling [10].

The problem of semantic misunderstanding has also been investigated by previous works. EI-CLIP [37] considers the problem of cross-modal retrieval in the field of E-commerce. Sharing similar insight with our work, the authors notice the model bias towards some specific word tokens in CLIP, and introduce causal inference to align the text encoder with e-commerce domain knowledge. K3M [77] focuses on the modality-missing and modality-noise problem and introduces knowledge modality into E-commerce tasks. DeVLBert [73] studies the spurious correlations between different modalities and adjusts the conditional probability of image tokens and word tokens. Kaleido-BERT [78] focuses on image-text coherence by introducing several novel self-supervised tasks.

Comparing to previous approaches, we are the first to incorporate multi-modal knowledge graphs into the pre-training process, and effectively enhance the model perception on semantic relations between visual and language concepts.

Knowledge Graph. Knowledge graph is first introduced in the field of natural language processing, and the knowledge graph embedding approaches have been successful on capturing the semantics of symbols (entities and relations) and achieving impressive results on a wide range of real-world applications including text understanding [15, 70], recommendation system [18, 61] and natural language question answering [20, 74]. On the other hand, scene graphs represent a type of graph-structured data in computer vision, where the visual concepts in the image are connected with semantic relations. Scene graphs emphasize the fine-grained semantic features for images and are widely adopted in various downstream tasks, including scene graph generation [69], and Scene Graph Parsing [72]. Besides scene graph, knowledge graph is also adopted in other computer vision tasks, including image classification [25], panoptic segmentation [67], and image captioning [75]. On this basis, multi-modal knowledge graph earns wide concerns in recent years. Considering the natural alignment between different data modalities, multi-modal knowledge graphs have been widely adopted in various graph-based tasks including link prediction [3, 33], entity classification [66], while also showing great potential on out of graph applications like visual question answering [22, 44] and recommendation systems [52, 56].

3 Contrastive Language-Image Pre-training (CLIP)

We first provide a brief review of model architectures and training settings in CLIP.

CLIP uses two separate models for image encoder and text encoder respectively. For text inputs, a 12-layer Transformer is adopted with 512 width and 8 attention heads. Raw texts are first converted using byte pair encoding [43] technique under a vocabulary size of 49,152. The text sequence length is capped at 76 and added by a positional encoding before being sent into the text encoder. On the other hand, CLIP has different versions of image encoder with ResNet-based and Vision Transformer-based architectures. As the following researches have demonstrated the better performances of Vision Transformer models, we only consider Transformer-based image encoders in this paper. Similar to the text input, images are first converted to patches, and added by a positional encoding. At the last stage of both encoders, a global pooling function is adopted to compress the feature map into a single

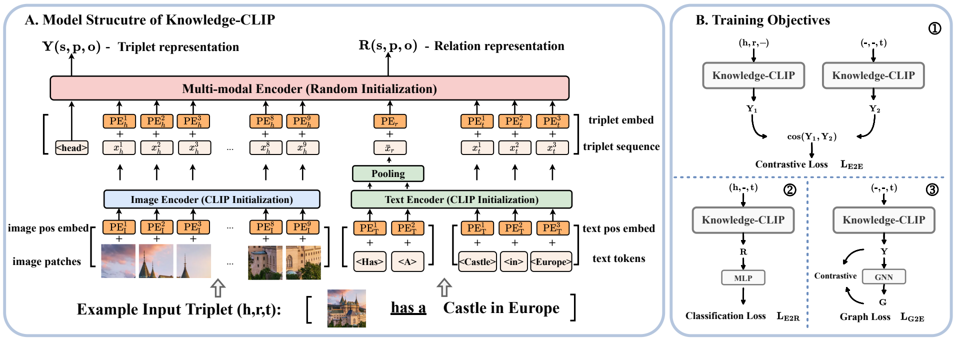


Figure 2: An overview of our framework. (A) Given a data triplet h, r, t with entities h, t and their relation r, image and text encoders first extract raw features, then a multi-modal encoder consumes the concatenated triplet sequence and outputs triplet and relation representations. (B) Three types of training objectives adopted in our framework.

feature, which serves as the representation of the whole image/text sequence. The cosine distance of the image and text features is computed as the similarity of the data pair. For training supervision, a contrastive loss is adopted to maximize the similarity of matched pairs while minimizing the similarity of unmatched pairs. Given a batch of N data pairs $\{\mathrm{I}_i, \mathrm{T}_i\}_{i=1}^N$, where I_i and T represents the i_{th} image and text respectively, the loss function can be parameterized as:

$$L = -\frac{1}{2}\sum_{i=1}^{N}\left(\log\frac{\exp(\cos(f_{\mathrm{I}}(\mathrm{I}_i), f_{\mathrm{T}}(\mathrm{T}_i))/\tau)}{\sum_{j=1}^{N}\exp(\cos(f_{\mathrm{I}}(\mathrm{I}_i), f_{\mathrm{T}}(\mathrm{T}_j))/\tau)} + \log\frac{\exp(\cos(f_{\mathrm{I}}(\mathrm{I}_i), f_{\mathrm{T}}(\mathrm{T}_i))/\tau)}{\sum_{j=1}^{N}\exp(\cos(f_{\mathrm{I}}(\mathrm{I}_j), f_{\mathrm{T}}(\mathrm{T}_i))/\tau)}\right), \tag{1}$$

where f_{I} and f_{T} correspond to image and text encoders, $\cos(\cdot)$ denotes the cosine similarity between the inputs, and τ is a learnable temperature initialized at 0.07.

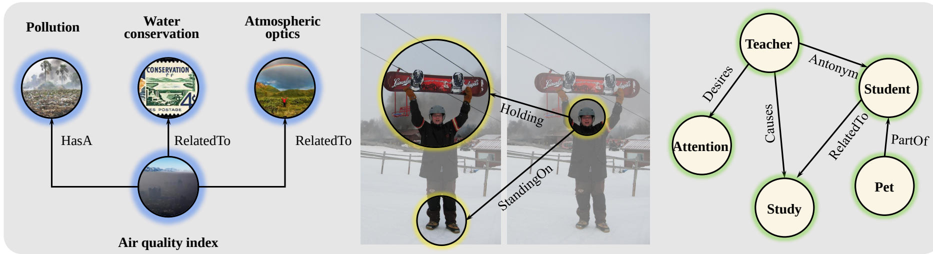
While effective, this simple training framework actually brings several concerns that need to be addressed. First, the pre-training framework fails to model the semantic information of inputs due to the simplicity of the data structure. This results in inferior performances on tasks that require reasoning ability, *e.g.*, visual question answering and visual commonsense reasoning. Second, the image and text features reside in separate spaces, which makes it difficult to model the interactions between different modalities. Third, the massive time and resource consumption in the training procedure set restrictions on performing a full pre-training schedule from scratch.

4 Knowledge-CLIP

As we have summarized above, there are several concerns that hinder the transferability of CLIP and potential improvements on model performances. In this paper, we propose a novel pre-training framework based on knowledge graphs, that addresses the limitation of the original CLIP model from several perspectives: (1) We introduce knowledge graphs into the training dataset where the graph-structured data and semantic relations between concepts enable the model to extract semantic features and establish semantic connection across inputs; (2) A multi-modal encoder is added on top of the current image and text encoders to fuse the features from different modalities, and model the joint distribution between inputs; (3) A continuous learning strategy based on the pre-trained model of CLIP is adopted which greatly avoids the massive computation cost in the pre-training procedure, and enhance the generalization power of the model efficiently. We introduce our framework in detail in the following sections, and show the overview in Fig. 2.

4.1 Data Preparation

Different from raw image-text pairs adopted in the original CLIP, our model takes knowledge graphs as input. A knowledge graph can be defined as a directed graph $\mathcal{G} = \{\xi, \mathcal{R}, \mathcal{T}_{\mathcal{R}}\}$, where ξ, $\mathcal{R}$ correspond to sets of entities and relations, and $\mathcal{T}_{\mathcal{R}}$ represent the set of relation triplets. A triplet $(h, r, t) \in \mathcal{T}_{\mathcal{R}}$ denotes that entity $h \in \xi$ has relation $r \in \mathcal{R}$ with entity $t \in \xi$. As illustrated in Fig. 3, we pre-train our model on three types of knowledge graphs, including multi-modal knowledge graph,



(a) VisualSem **(b) Visual Genome** **(c) ConceptNet**

Figure 3: Illustrations of the pre-training knowledge graph datasets, including ViusalSem [2] (multi-modal graph), Visual Genome [27] (scene graph), and ConceptNet [49] (language-based graph).

scene graph, and language-based knowledge graph. Among these, relations are constantly described in language tokens, where the entities are from different modalities in different forms.

For multi-modal knowledge graph, the entities contain both illustrative images and language descriptions. Through representing the same entity under various modalities and connecting entities with relations, it helps to build semantic connections between vision and language concepts. In practice, language and vision descriptions are randomly chosen for each entity. In this way, the triplet set $\mathcal{T}_{\mathcal{R}}$ contains different forms including (Img, Rel, Img), (Img, Rel, Text), and (Text, Rel, Text), providing rich information across modalities while also enhancing perceptions within modalities.

Different from multi-modal knowledge graph, scene graph extracts visual concepts (mainly objects) for each image, and connects them with predefined semantic relations describing relative locations, actions, etc. Therefore, the entities in the scene graph correspond to a certain region in an image, with the triplet form of (Img, Rel, Img). We practically use the selected regions as the input and discard the irrelevant parts. As two entities in the same triplet denote different regions in the same image, it forces the model to extract more fine-grained features.

Lastly, language-based knowledge graph connects words and phrases of natural language with labeled edges. It is built on only language modality with the triplet form of (Text, Rel, Text), while helping to build semantic alignment within word tokens.

4.2 Model Architecture

The model architecture and the training framework are illustrated in Fig. 2(A). Specifically, we first process the inputs into token sequences with modality-specific tokenizers. The BPE tokenzier [43] is adopted for language inputs, while image inputs are sliced into non-overlapped patches and converted into a sequence of patches following ViT [14]. For convenient processing, we set the length of the image sequence and text sequence as l_{I} and l_{T} respectively for all inputs. To preserve the relative position information in the input, learnable positional encodings are added to the corresponding sequences before being sent to the model.

Two separate image encoder $f_{\mathrm{I}}(\cdot)$ and text encoder $f_{\mathrm{T}}(\cdot)$ are then adopted to extract features from raw inputs. For a given triplet (h, r, t), the entities h and t are sent to the encoders with respect to their modalities (image or text). The relation r, which is represented by language tokens, is sent to text encoder similar to text entity.

Comparing to the model structure in CLIP, we introduce a modification to better adapt our framework. Specifically, vanilla CLIP models use a pooling function at the last layer of two encoders to compress the feature map into a global representation. Namely, for an input $u \in \mathcal{R}^{L \times d_i}$, where L and d_i denote the sequence length and feature dimension, the output of the encoder can be formulated as:

$$x_u = f(u) \in \mathcal{R}^{L \times d_o}, \;\; \bar{x}_u = \mathrm{Pool}(x_u) \in \mathcal{R}^{d_o}, \tag{2}$$

where f represents the feature extraction module, $\mathrm{Pool}(\cdot)$ denotes the pooling function, and d_o is the output dimension. Though efficient, it also leads to inevitable information loss in the local region, especially for the image inputs. Therefore, we remove the pooling functions for image and text entities to preserve the local information, and use $x_u \in \mathcal{R}^{L \times d_o}$ as the extracted feature. The relation, on the other hand, is normally under a limited sequence length, *e.g.*, one or two word tokens, where the information density is smaller than entities. Therefore, we retain the pooling function for relation input and use $\bar{x}_u \in \mathcal{R}^{d_o}$ as the extracted features.

In this way, we have extracted the features defined as $(x_h, \bar{x}_r, x_t)$, which correspond to the elements in the input triplet (h, r, t). To model the joint distribution of different elements in the triplet, we consider a multi-modal encoder $\mathrm{TransEncoder}(\cdot)$ to fuse the features from different sources. Specifically, we first concatenate all the features in the triplet into a single sequence and use a head token $<\mathrm{head}>$ at the beginning of the sequence. To emphasize the status of the tokens in the sequence, we consider additional learnable encodings for each element h, r, t in the triplet:

$$X(h, r, t) = [<\text{head}>,\ x_h{+}\mathrm{PE}_h,\ \bar{x}_r{+}\mathrm{PE}_r,\ x_t{+}\mathrm{PE}_t]. \tag{3}$$

After processing by the multi-modal encoder, the feature of the head token $<\text{head}>$ finally serves as the representation of the whole sequence:

$$Y(h, r, t) = \mathrm{TransEncoder}(X(h, r, t))[0, :]. \tag{4}$$

Also, representation for relation is extracted from the corresponding token:

$$R(h, r, t) = \mathrm{TransEncoder}(X(h, r, t))[1 + \mathrm{len}(x_h), :]. \tag{5}$$

4.3 Training Targets

Considering the unique data structure of knowledge graphs, we mainly adopt two types of training targets in our framework, including triplet-based loss and graph-based loss as illustrated in Fig. 2(B). Besides, a knowledge distillation loss is also considered due to the continuous learning strategy adopted in our framework.

Triplet-based loss considers a batch of triplets as the input and supervises the training of our model by estimating the joint distribution of elements in the triplets. Inspired by the mask prediction technique that models the distribution of masked tokens conditioned on the unmasked regions, we similarly mask the elements in the triplets and predict the distribution with the help of a multi-modal encoder. Specifically, for incomplete triplets where certain elements are missing in the input, the concatenated sequence can be similarly derived as in Eq. 3 by masking the corresponding feature. For example, the concatenated sequence for an input $(h, r, \text{-})$ can be represented as:

$$X(h, r, \text{-}) = [<\text{head}>,\ x_h{+}\mathrm{PE}_h,\ \bar{x}_r{+}\mathrm{PE}_r,\ \mathbf{0}]. \tag{6}$$

On this basis, given a set of input $D = \{(h_i, r_i, t_i)\}_{i=1}^{N}$, we first model the distribution when one of the entities, *i.e.*, t_i, is masked, and derive the Entity-Entity (E2E) Loss by minimizing the negative log-likelihood:

$$-E_{(h,r)\sim D}\log(P(x_t | x_h, \bar{x}_r)). \tag{7}$$

We practically approximate the distribution $P(x_t | x_h, \bar{x}_r)$ as the cosine similarity of $P(x_t)$ and $P(x_h, \bar{x}_r)$, and defined the loss function as:

$$L_{\mathrm{E2E}} = -\sum_{i=1}^{N} \log\left(\frac{\exp(\cos(Y(\text{-}, \text{-}, t_i), Y(h_i, r_i, \text{-}))/\tau)}{\sum_j \exp(\cos(Y(\text{-}, \text{-}, t_i), Y(h_j, r_j, \text{-}))/\tau)}\right). \tag{8}$$

We also model the distribution when the relation in the triplet is masked, and similarly derive the Entity-Relation (E2R) Loss:

$$-E_{(h,t)\sim D}\log(P(\bar{x}_r | x_h, x_t)). \tag{9}$$

Different from E2E loss, the relations in the triplets are defined in a limited set of relation groups. Therefore, we instead extract the representation of relation through an auxiliary two-layer MLP network, and model the objective as a classification problem from a predefined set of relation labels $\mathcal{R}$. In this way, the loss function can be defined as:

$$L_{\mathrm{E2R}} = -\sum_{i=1}^{N} \sum_{r\in\mathcal{R}} \mathbf{1}_{(r=r_i)} \log(y(\bar{x}_{r_i})), \quad \text{where}\ \ y(\bar{x}_{r_i}) = \mathrm{MLP}(R(h_i, \text{-}, t_i)), \tag{10}$$

is extracted from an MLP model followed by the output of multi-modal encoder defined in Eq. (5).

Graph-based loss. We also take advantage of the graph structure in knowledge graph datasets, and adopt a graph neural network to extract deeper structural information among entities. We propagate information through connected edges in the graph, and update entity representations with aggregated feature. Specifically, for a graph neural network with L layers, the update function for the l_{th} layer can be formulated as:

$$G^{(l)}(t) = E_{\{h_i, r_i, t\} \in \mathcal{T}_{\mathcal{R}}} \, g^{(l-1)}(R(h_i, \text{-}, t)) G^{(l-1)}(h_i), \;\; G^0(t) = Y(\text{-}, \text{-}, t), \tag{11}$$

$$\text{where } \; g^{(l)}(R(h_i, \text{-}, t)) = W^{(l)} R(h_i, \text{-}, t), \tag{12}$$

calculates the aggregation weights by relation representation $R(h_i, \text{-}, t)$ with a learnable matrix $W^{(l)}$.

Finally, we define the Graph-Entity(G2E) Loss by computing the cosine similarity of entity features before and after the propagation procedure in the graph:

$$L_{\mathrm{G2E}} = -\frac{1}{\mathcal{N}_\xi} \sum_{t_i \in \xi} \log\left(\frac{\exp(\cos(Y(\text{-}, \text{-}, t_i), G^{(L)}(t_i))/\tau)}{\sum_{t_j} \exp(\cos(Y(\text{-}, \text{-}, t_i), G^{(L)}(t_j))/\tau)}\right). \tag{13}$$

Continuous Learning. Large-scale pre-training usually requires massive computation resources which makes it highly inefficient when training from scratch. Therefore, to inject the semantic information in an efficient manner, we consider training our model based on the pre-trained weights from the original CLIP. This powerful initialization promotes the convergence of our model and greatly enhances the training efficiency. However, naively extending the training process with new data leads to severe forgetting problem that hampers the performance of the original models.

To address this limitation, we adopt simple solutions to maintain CLIP performances while improving its ability to extract semantic features from knowledge graphs. (1) Besides the knowledge graph datasets, we also train our model on several widely adopted image-text datasets that share a similar data distribution with the training data in CLIP. To better fit our pre-training framework, we convert the original image-text pair into the form of triplets, with specifically designed relations 'image of' and 'caption of'. (2) We also use the original CLIP model as the teacher, and use an auxiliary loss L_{KD} to measure the KL distance between the output of CLIP and our model.

Overall, the final pre-training objective of Knowledge-CLIP is formulated as:

$$L = L_{\mathrm{E2E}} + L_{\mathrm{E2R}} + L_{\mathrm{G2E}} + L_{\mathrm{KD}}. \tag{14}$$

5 Experiments

5.1 Implementation Details

Experimental Setup. In all the experiments, we use the same model structure as CLIP [41]. A 12-layer Transformer model with 512 width is adopted for text encoder, and ViT-L/14 is adopted for image encoder. For text and image encoder, we use the pre-trained weights in the original CLIP as the initialization. For the multi-modal encoder, we consider a 4 layer Transformer model with 1024 width. The rate for drop path is set as 0.1 during training. As the added multi-modal encoder is trained from random initialization, we decrease the learning rate for the pre-trained weights from CLIP to achieve a more balanced step in the optimization. We train Knowledge-CLIP with an initial learning rate of 1e-5 for image and text encoders, and 1e-3 for the multi-modal encoder. Cosine learning rate with linear warmup is used in the training schedule. Weight decay and gradient clip are also adopted. See more details in the supplemental material.

Pre-train Dataset. Three knowledge graph datasets are adopted in the pre-training process. VisualSem [2] is a high-quality multi-modal knowledge graph dataset for vision and language concepts, including entities with multilingual glosses, multiple illustrative images, and visually relevant relations, covering a total number of 90k nodes, 1.3M glosses and 938k images. 13 semantic relations are used to connect different entities in the graph, while the entities in VisualSem are linked to Wikipedia articles, WordNet [38], and high-quality images from ImageNet [11]. Visual Genome [27] is a knowledge-based scene graph dataset that connects structured image concepts with semantic relations. Visual Genome serves as the benchmark for various vision tasks, *e.g.*, visual grounding, and scene graph generation. ConceptNet [49] is a knowledge graph that connects words and phrases of natural language with labeled edges. Its knowledge is collected from many sources including expert-created resources and crowd-sourcing built on only language modality.

Besides the three knowledge graph datasets, we also train our model on two widely adopted image-text datasets that share the similar data distribution with the training data in CLIP. We practically add

Table 1: Fine-tuned image-text retrieval results on Flockr30K and COCO datasets. The best result is shown in blue and the better result between CLIP and our approach is shown in **bold**.

Method	Flickr30K (1K test set)						MSCOCO(5K test set)					
	Text Retrieval			Image Retrieval			Text Retrieval			Image Retrieval		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
UNITER [10]	87.3	98.0	99.2	75.6	94.1	96.8	65.7	88.6	93.8	52.9	79.9	88.0
VILLA [16]	87.9	97.5	98.8	76.3	94.2	96.8	-	-	-	-	-	-
OSCAR [31]	-	-	-	-	-	-	73.5	92.2	96.0	57.5	82.8	89.8
ERNIE-ViL [71]	88.7	98.0	99.2	76.7	93.6	96.4	-	-	-	-	-	-
Unicoder-VL [28]	86.2	96.3	99.0	71.5	91.2	95.2	62.3	87.1	92.8	48.4	76.7	85.9
ViLT [26]	83.5	96.7	98.6	64.4	88.7	93.8	61.5	86.3	92.7	42.7	72.9	83.1
Uni-Perceiver [76]	87.9	98.2	99.1	74.9	93.5	96.0	64.7	87.8	93.7	48.3	75.9	84.5
CLIP [41]	88.6	98.5	**99.4**	72.4	92.3	96.6	67.3	85.4	92.4	54.3	83.5	90.0
Ours	**89.2**	**98.9**	**99.4**	**75.7**	**94.4**	**96.8**	**70.2**	**89.2**	**94.4**	**57.6**	**83.9**	**90.4**

COCO Caption [9] and CC3M [45] to the training set, while large-scale datasets like CC12M [5] or YFCC [24] are not considered to maintain training efficiency.

Downstream Task. To validate the effectiveness of our framework, we conduct experiments on various downstream tasks, including multi-modal tasks like text and image retrieval, visual question answering, and uni-modal tasks like image classification and natural language understanding. We also show the performances of our models on several knowledge-based tasks including link prediction and triple classification, where our model can benefit from the graph-based training schedule.

5.2 Multi-modal Tasks

Image and text retrieval. We first conduct experiments on Flickr30k [40] and COCO Caption [9] dataset to show the performances of our model on image-text retrieval tasks. Given input sets $\mathcal{X}$ and $\mathcal{Y}$ of images and texts, we use Knowledge-CLIP to extract features for each input, and model the joint probability with the cosine similarity between image and text pairs. We summarize the comparison results of Knowledge-CLIP with competitive baselines in Tab. 1. It is shown that our model consistently achieves better results over the original CLIP on both datasets, while comparable with competitive baselines like OSCAR.

Visual question answering / Visual Entailment. We also validate the effectiveness of Knowledge-CLIP on other vision-language tasks, including VQA [17] and SNLI-VE [68]. We show the comparison results in Tab. 2. Comparing to competitive baselines including VILLA [16] and ALBEF [29], Knowledge-CLIP with ViT-L/14 shows better performances under all settings, while the smaller model also achieves competitive results. Comparing to the original CLIP model, our pre-trained model practically improves its transferability on downstream tasks, especially on the datasets like VQA that requires reasoning ability.

Table 2: Fine-tuned results on other V-L tasks.

Method	VQA		SNLI_VE	
	test-dev	test-std	val	test
UNITER [10]	72.70	72.91	78.59	78.28
VILLA [16]	73.59	73.67	79.47	79.03
OSCAR [31]	73.16	73.44	-	-
ALBEF [29]	74.54	74.70	80.14	80.30
Uni-Perceiver [76]	73.4	74.1	-	-
FLAVA [48]	72.8	-	78.89	-
CLIP [41]	74.10	73.56	79.51	80.01
Ours	**76.11**	**75.24**	**80.52**	**80.97**

5.3 Uni-modal Tasks

Image Classification. To further demonstrate the generalization power of Knowledge-CLIP, we compare the performances of pre-train models on the ImageNet classification task [11]. We summarize the comparison results in Tab. 3, and show that Knowledge-CLIP can also handle vision tasks well. We argue the improvements over baselines may attribute to the scene graphs in our pre-training dataset, which emphasize the visual concepts in the images.

Table 3: Fine-tuned results on ImageNet.

Method	Acc(%)
DeiT [58]	83.4
CLIP [41]	84.2
Ours	**84.4**

Language Understanding. We validate the generalization performance of Knowledge-CLIP for language understanding tasks on the widely adopted GLUE dataset [60]. Specifically, we conduct

Table 4: Fine-tuned language understanding results on GLUE dataset. The best result is shown in blue and the better result between CLIP and our approach is shown in **bold**.

Method	CoLA Mcc.	SST-2 Acc.	RTE Acc.	MRPC Acc./F1	QQP Acc./F1	MNLI Acc	QNLI Acc
VilBERT [35]	36.1	90.4	53.7	69.0/79.4	88.6/85.0	79.9	83.8
VL-BERT [51]	38.7	89.8	55.7	70.6/81.8	89.0/85.4	81.2	86.3
UNITER [10]	37.4	89.7	55.6	69.3/80.3	89.2/85.7	80.9	86.0
SimVLM [65]	46.7	90.0	63.9	75.2/84.4	90.4/87.2	83.4	88.6
FLAVA [48]	50.7	90.9	57.8	81.4/86.9	90.4/87.2	80.3	87.3
CLIP [41]	42.1	90.5	59.2	82.4/87.0	90.4/87.1	80.9	87.1
Ours	**50.4**	**91.2**	**62.4**	**83.5/87.6**	**90.5/87.9**	**83.6**	**89.5**

Table 5: Fine-tuned link prediction results on WN18RR and FB15K-237.

Method	WN18RR					FB15k-237				
	MR	MMR	Hits @1	Hits @3	Hits @10	MR	MMR	Hits @1	Hits @3	Hits @10
TransE [4]	3384	0.182	0.027	0.295	0.444	357	0.257	0.174	0.284	0.420
ConvE [12]	4187	0.430	0.400	0.440	0.520	244	0.325	0.237	0.356	0.501
RotatE [53]	3340	**0.476**	0.428	**0.492**	0.571	177	0.338	0.241	0.375	0.533
InteractE [59]	5202	0.463	-	0.430	0.528	**172**	0.354	0.263	-	**0.535**
Ours	**2689**	0.467	**0.430**	0.477	**0.572**	182	**0.356**	**0.281**	**0.391**	0.530

experiments on 7 tasks in GLUE and summarize the comparison results in Tab. 4. It is shown that our model achieves comparable performances with competitive baseline models. Also, for tasks like QQP and MNLI that require sentence-pair matching, Knowledge-CLIP shows higher performances, due to the existence of language triplets in the pre-training dataset.

5.4 Knowledge-based Tasks

Benefiting from the graph-based learning framework in the pre-training process, our models enjoy advantages on several knowledge-based downstream tasks. Therefore, we conduct experiments on link prediction, entity classification and triple classification tasks.

Link prediction task aim to recover an incomplete triplet when one of the entities is masked, *i.e.*, predicting entity h given $(\text{-}, r, t)$. This task shares certain similarities with our pre-training objectives. We validate the performances of our model on the WN18RR [12] and FB15K-237 [57] datasets, where MR (MeanRank), MRR(Mean Reciprocal Rank), and Hit@n are adopted as the evaluation metrics. As shown in Tab. 5, Knowledge-CLIP is able to perform competitive performances comparing to several baseline models, and achieves better results on 3 of 5 metrics.

Triple classification requires the model to distinguish matched triples from unmatched ones, which can serve as a binary classification task. We validate our model on YAGO39K [36] dataset, with Accuracy, Precision, Recall, and F1-Score as the evaluation metric. It is shown in Tab. 6 that our model shows promising results over competitive baselines.

Table 6: Fine-tuned results on YAGO39K.

Method	Triple Classification(%)			
	Accuracy	Precision	Recall	F1-Score
TransE [4]	92.1	**92.8**	91.2	92.0
TransD [23]	89.3	88.1	91.0	89.5
HolE [39]	92.3	92.6	**91.9**	92.3
Ours	**92.7**	92.6	**91.9**	**92.5**

6 Conclusion

In this paper, we propose a novel vision-language pretraining framework that incorporates knowledge information to model the semantic connections between vision and language entities. We introduce three types of graph-structured datasets into the training process, and adopt a multi-modal encoder to model the joint distribution of entities and their semantic relations. Extensive experiments on various downstream tasks including multi-modal, uni-modal, and graph-based tasks validate the transfer and generalization ability of our model. Our approach is now limited in injecting knowledge information into the CLIP models. However, our training objectives and new knowledge graph datasets are technically compatible with other large-scale pretraining frameworks. We will explore the possibility of further applications in the future.

References

[1] Chris Alberti, Jeffrey Ling, Michael Collins, and David Reitter. Fusion of detected objects in text for visual question answering. *arXiv preprint arXiv:1908.05054*, 2019. 3

[2] Houda Alberts, Teresa Huang, Yash Deshpande, Yibo Liu, Kyunghyun Cho, Clara Vania, and Iacer Calixto. Visualsem: a high-quality knowledge graph for vision and language. *arXiv preprint arXiv:2008.09150*, 2020. 2, 5, 7

[3] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. *Advances in neural information processing systems*, 26, 2013. 3

[4] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. *Advances in neural information processing systems*, 26, 2013. 9

[5] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3558–3568, 2021. 8

[6] Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In *International Conference on Machine Learning*, pages 1691–1703. PMLR, 2020. 3

[7] Wenhu Chen, Wenhan Xiong, Xifeng Yan, and William Wang. Variational knowledge graph reasoning. *arXiv preprint arXiv:1803.06581*, 2018. 2

[8] Xiaojun Chen, Shengbin Jia, and Yang Xiang. A review: Knowledge reasoning over knowledge graph. *Expert Systems with Applications*, 141:112948, 2020. 2

[9] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015. 8

[10] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Learning universal image-text representations. 2019. 1, 3, 8, 9

[11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 2, 7, 8

[12] Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. Convolutional 2d knowledge graph embeddings. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018. 9

[13] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 3

[14] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 5

[15] Hao Fei, Yafeng Ren, Yue Zhang, Donghong Ji, and Xiaohui Liang. Enriching contextualized language model from knowledge graph for biomedical information extraction. *Briefings in bioinformatics*, 22(3):bbaa110, 2021. 3

[16] Zhe Gan, Yen-Chun Chen, Linjie Li, Chen Zhu, Yu Cheng, and Jingjing Liu. Large-scale adversarial training for vision-and-language representation learning. *Advances in Neural Information Processing Systems*, 33:6616–6628, 2020. 8

[17] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6904–6913, 2017. 8

[18] Qingyu Guo, Fuzhen Zhuang, Chuan Qin, Hengshu Zhu, Xing Xie, Hui Xiong, and Qing He. A survey on knowledge graph-based recommender systems. *IEEE Transactions on Knowledge and Data Engineering*, 2020. 3

[19] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. *arXiv preprint arXiv:2111.06377*, 2021. 3

[20] Xiao Huang, Jingyuan Zhang, Dingcheng Li, and Ping Li. Knowledge graph embedding based question answering. In *Proceedings of the twelfth ACM international conference on web search and data mining*, pages 105–113, 2019. 2, 3

[21] Zhenyu Huang, Guocheng Niu, Xiao Liu, Wenbiao Ding, Xinyan Xiao, Hua Wu, and Xi Peng. Learning with noisy correspondence for cross-modal matching. *Advances in Neural Information Processing Systems*, 34, 2021. 1

[22] Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6700–6709, 2019. 3

[23] Guoliang Ji, Shizhu He, Liheng Xu, Kang Liu, and Jun Zhao. 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 (volume 1: Long papers)*, pages 687–696, 2015. 9

[24] Sebastian Kalkowski, Christian Schulze, Andreas Dengel, and Damian Borth. Real-time analysis and visualization of the yfcc100m dataset. In *Proceedings of the 2015 workshop on community-organized multimodal mining: opportunities for novel solutions*, pages 25–30, 2015. 8

[25] Michael Kampffmeyer, Yinbo Chen, Xiaodan Liang, Hao Wang, Yujia Zhang, and Eric P Xing. Rethinking knowledge graph propagation for zero-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11487–11496, 2019. 3

[26] Wonjae Kim, Bokyung Son, and Ildoo Kim. Vilt: Vision-and-language transformer without convolution or region supervision. In *International Conference on Machine Learning*, pages 5583–5594. PMLR, 2021. 8

[27] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123(1):32–73, 2017. 2, 5, 7

[28] Gen Li, Nan Duan, Yuejian Fang, Ming Gong, and Daxin Jiang. Unicoder-vl: A universal encoder for vision and language by cross-modal pre-training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11336–11344, 2020. 8

[29] Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. *Advances in Neural Information Processing Systems*, 34, 2021. 1, 8

[30] Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*, 2019. 3

[31] Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, et al. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *European Conference on Computer Vision*, pages 121–137. Springer, 2020. 1, 8

[32] Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. Learning entity and relation embeddings for knowledge graph completion. In *Twenty-ninth AAAI conference on artificial intelligence*, 2015. 2

[33] Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. Learning entity and relation embeddings for knowledge graph completion. In *Twenty-ninth AAAI conference on artificial intelligence*, 2015. 3

[34] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Advances in neural information processing systems*, 32, 2019. 3

[35] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Advances in neural information processing systems*, 32, 2019. 9

[36] Xin Lv, Lei Hou, Juanzi Li, and Zhiyuan Liu. Differentiating concepts and instances for knowledge graph embedding. *arXiv preprint arXiv:1811.04588*, 2018. 9

[37] Haoyu Ma, Handong Zhao, Zhe Lin, Ajinkya Kale, Zhangyang Wang, Tong Yu, Jiuxiang Gu, Sunav Choudhary, and Xiaohui Xie. Ei-clip: Entity-aware interventional contrastive learning for e-commerce cross-modal retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18051–18061, 2022. 3

[38] George A Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995. 7

[39] Maximilian Nickel, Lorenzo Rosasco, and Tomaso Poggio. Holographic embeddings of knowledge graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30, 2016. 9

[40] Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Proceedings of the IEEE international conference on computer vision*, pages 2641–2649, 2015. 8

[41] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 1, 7, 8, 9

[42] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. 2018. 3

[43] Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*, 2015. 3, 5

[44] Sanket Shah, Anand Mishra, Naganand Yadati, and Partha Pratim Talukdar. Kvqa: Knowledge-aware visual question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8876–8884, 2019. 3

[45] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565, 2018. 8

[46] Baoxu Shi and Tim Weninger. Open-world knowledge graph completion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018. 2

[47] Aman Shrivastava, Ramprasaath R Selvaraju, Nikhil Naik, and Vicente Ordonez. Clip-lite: Information efficient visual representation learning from textual annotations. *arXiv preprint arXiv:2112.07133*, 2021. 1

[48] Amanpreet Singh, Ronghang Hu, Vedanuj Goswami, Guillaume Couairon, Wojciech Galuba, Marcus Rohrbach, and Douwe Kiela. Flava: A foundational language and vision alignment model. *arXiv preprint arXiv:2112.04482*, 2021. 8, 9

[49] Robyn Speer, Joshua Chin, and Catherine Havasi. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Thirty-first AAAI conference on artificial intelligence*, 2017. 2, 5, 7

[50] Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. Vl-bert: Pre-training of generic visual-linguistic representations. *arXiv preprint arXiv:1908.08530*, 2019. 3

[51] Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. Vl-bert: Pre-training of generic visual-linguistic representations. *arXiv preprint arXiv:1908.08530*, 2019. 9

[52] Xiaoyuan Su and Taghi M Khoshgoftaar. A survey of collaborative filtering techniques. *Advances in artificial intelligence*, 2009, 2009. 3

[53] Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. Rotate: Knowledge graph embedding by relational rotation in complex space. *arXiv preprint arXiv:1902.10197*, 2019. 9

[54] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. *arXiv preprint arXiv:1908.07490*, 2019. 1

[55] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. *arXiv preprint arXiv:1908.07490*, 2019. 3

[56] Shaohua Tao, Runhe Qiu, Yuan Ping, and Hui Ma. Multi-modal knowledge-aware reinforcement learning network for explainable recommendation. *Knowledge-Based Systems*, 227:107217, 2021. 3

[57] Kristina Toutanova, Danqi Chen, Patrick Pantel, Hoifung Poon, Pallavi Choudhury, and Michael Gamon. Representing text for joint embedding of text and knowledge bases. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 1499–1509, 2015. 9

[58] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, pages 10347–10357. PMLR, 2021. 8

[59] Shikhar Vashishth, Soumya Sanyal, Vikram Nitin, Nilesh Agrawal, and Partha Talukdar. Interacte: Improving convolution-based knowledge graph embeddings by increasing feature interactions. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 3009–3016, 2020. 9

[60] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*, 2018. 8

[61] Hongwei Wang, Fuzheng Zhang, Jialin Wang, Miao Zhao, Wenjie Li, Xing Xie, and Minyi Guo. Ripplenet: Propagating user preferences on the knowledge graph for recommender systems. In *Proceedings of the 27th ACM international conference on information and knowledge management*, pages 417–426, 2018. 3

[62] Peng Wang, An Yang, Rui Men, Junyang Lin, Shuai Bai, Zhikang Li, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. *arXiv preprint arXiv:2202.03052*, 2022. 1

[63] Quan Wang, Zhendong Mao, Bin Wang, and Li Guo. Knowledge graph embedding: A survey of approaches and applications. *IEEE Transactions on Knowledge and Data Engineering*, 29(12):2724–2743, 2017. 2

[64] Zhouxia Wang, Tianshui Chen, Jimmy Ren, Weihao Yu, Hui Cheng, and Liang Lin. Deep reasoning with knowledge graph for social relationship understanding. *arXiv preprint arXiv:1807.00504*, 2018. 2

[65] Zirui Wang, Jiahui Yu, Adams Wei Yu, Zihang Dai, Yulia Tsvetkov, and Yuan Cao. Simvlm: Simple visual language model pretraining with weak supervision. *arXiv preprint arXiv:2108.10904*, 2021. 9

[66] WX Wilcke, Peter Bloem, Victor de Boer, RH van t Veer, and FAH van Harmelen. End-to-end entity classification on multimodal knowledge graphs. *arXiv preprint arXiv:2003.12383*, 2020. 3

[67] Yangxin Wu, Gengwei Zhang, Yiming Gao, Xiajun Deng, Ke Gong, Xiaodan Liang, and Liang Lin. Bidirectional graph reasoning network for panoptic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 3

[68] Ning Xie, Farley Lai, Derek Doran, and Asim Kadav. Visual entailment: A novel task for fine-grained image understanding. *arXiv preprint arXiv:1901.06706*, 2019. 8

[69] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. Graph r-cnn for scene graph generation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 670–685, 2018. 2, 3

[70] Donghan Yu, Chenguang Zhu, Yiming Yang, and Michael Zeng. Jaket: Joint pre-training of knowledge graph and language understanding. *arXiv preprint arXiv:2010.00796*, 2020. 3

[71] Fei Yu, Jiji Tang, Weichong Yin, Yu Sun, Hao Tian, Hua Wu, and Haifeng Wang. Ernie-vil: Knowledge enhanced vision-language representations through scene graph. *arXiv preprint arXiv:2006.16934*, 2020. 8

[72] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. Neural motifs: Scene graph parsing with global context. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5831–5840, 2018. 3

[73] Shengyu Zhang, Tan Jiang, Tan Wang, Kun Kuang, Zhou Zhao, Jianke Zhu, Jin Yu, Hongxia Yang, and Fei Wu. Devlbert: Out-of-distribution visio-linguistic pretraining with causality. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1744–1747, 2021. 3

[74] Yuyu Zhang, Hanjun Dai, Zornitsa Kozareva, Alexander J Smola, and Le Song. Variational reasoning for question answering with knowledge graph. In *Thirty-second AAAI conference on artificial intelligence*, 2018. 2, 3

[75] Wentian Zhao, Yao Hu, Heda Wang, Xinxiao Wu, and Jiebo Luo. Boosting entity-aware image captioning with multi-modal knowledge graph. *arXiv preprint arXiv:2107.11970*, 2021. 3

[76] Xizhou Zhu, Jinguo Zhu, Hao Li, Xiaoshi Wu, Xiaogang Wang, Hongsheng Li, Xiaohua Wang, and Jifeng Dai. Uni-perceiver: Pre-training unified architecture for generic perception for zero-shot and few-shot tasks. *arXiv preprint arXiv:2112.01522*, 2021. 1, 8

[77] Yushan Zhu, Huaixiao Zhao, Wen Zhang, Ganqiang Ye, Hui Chen, Ningyu Zhang, and Huajun Chen. Knowledge perceived multi-modal pretraining in e-commerce. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 2744–2752, 2021. 3

[78] Mingchen Zhuge, Dehong Gao, Deng-Ping Fan, Linbo Jin, Ben Chen, Haoming Zhou, Minghui Qiu, and Ling Shao. Kaleido-bert: Vision-language pre-training on fashion domain. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12647–12657, 2021. 3

Checklist

1. For all authors...
    (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] We have addressed our contribution.
    (b) Did you describe the limitations of your work? [Yes] See Section 6.
    (c) Did you discuss any potential negative societal impacts of your work? [Yes] See Supplementary material.
    (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes] I have carefully read the ethics review guidelines and checked our paper.
2. If you are including theoretical results...
    (a) Did you state the full set of assumptions of all theoretical results? [N/A]
    (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
    (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [No] The code will be released when the paper is accepted.
    (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Section 5 and supplemental material.
    (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No] We follow the experiment routine in the previous works.
    (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See supplemental material.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
    (a) If your work uses existing assets, did you cite the creators? [Yes] See Section 5.
    (b) Did you mention the license of the assets? [Yes] See Section 5.
    (c) Did you include any new assets either in the supplemental material or as a URL? [No]
    (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [No] The dataset is readily available in public.
    (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [No]
5. If you used crowdsourcing or conducted research with human subjects...
    (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
    (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
    (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]