Beyond Entities: A Large-Scale Multi-Modal Knowledge Graph with Triplet Fact Grounding

Jingping Liu[1*†], Mingchuan Zhang[2*], Weichen Li[2*], Chao Wang[3], Shuang Li[2], Haiyun Jiang[4], Sihang Jiang[2], Yanghua Xiao[2], Yunwen Chen[5]

[1]School of Information Science and Engineering, East China University of Science and Technology, Shanghai, China
[2]Shanghai Key Laboratory of Data Science, School of Computer Science, Fudan University, Shanghai, China
[3]School of Future Technology, Shanghai University, Shanghai, China
[4]Tencent AI Lab, Shenzhen, China
[5]DataGrand Inc., Shanghai, China
jingpingliu@ecust.edu.cn, {mczhang18, wcli18, lishuang18, shawyh}@fudan.edu.cn
cwang@shu.edu.cn, haiyunjiang@tencent.com, tedsihangjiang@gmail.com, chenyunwen@datagrand.com

Abstract

Much effort has been devoted to building multi-modal knowledge graphs by visualizing entities on images, but ignoring the multi-modal information of the relation between entities. Hence, in this paper, we aim to construct a new large-scale multi-modal knowledge graph with triplet facts grounded on images that reflect not only entities but also their relations. To achieve this purpose, we propose a novel pipeline method, including triplet fact filtering, image retrieving, entity-based image filtering, relation-based image filtering, and image clustering. In this way, a multi-modal knowledge graph named ImgFact is constructed, which contains 247,732 triplet facts and 3,730,805 images. In experiments, the manual and automatic evaluations prove the reliable quality of our ImgFact. We further use the obtained images to enhance model performance on two tasks. In particular, the model optimized by our ImgFact achieves an impressive 8.38% and 9.87% improvement over the solutions enhanced by an existing multi-modal knowledge graph and VisualChatGPT on F1 of relation classification. We release ImgFact and its instructions at https://github.com/kleinercubs/ImgFact.

Introduction

Multi-modal knowledge graphs (MMKGs) are important resources for a wide range of NLP and multimodal tasks. Diverging from the symbol-based knowledge graph (KG), MMKGs primarily link structured KG information with visual content, thereby establishing connections between textual and image-based knowledge representations.

However, the current focus of MMKG-related work mainly focuses on grounding entities to images (Wu et al. 2023b), without considering the visual semantics of relations, thus limiting their effectiveness in downstream tasks. As illustrated in Figure 1, the goal of entity grounding, taking the example of `David_Beckham` (`Victoria_Beckham`), is to identify images that accurately represent the entity. Obviously, it is insufficient to consider entities alone without their relations for two reasons. **(1)** From a cognitive perspective, grounding symbolic relations enhances machine comprehension of abstract relations by providing them with multi-modal experiences (Zhu et al. 2022). Figure 1 illustrates that visualizing `Spouse` in (`David_Beckham`, `Spouse`, `Victoria_Beckham`) enables machines to grasp the implicit intimate relationships between the couple, such as *wedding*, *pregnant*, and *family portrait*. **(2)** From an application perspective, a variety of real-world tasks (e.g., link prediction and relation classification) would benefit if we can ground symbolic relations to images (see Section "Experiments"). The benefit comes from image information not explicitly mentioned in symbolic knowledge. Thus, it is crucial to ground both entities and their relation on other modalities at once.

Figure 1: The difference between entity grounding and triplet fact grounding. The former aims at visualizing entities to images and most existing MMKGs are built under this task. The latter is to ground triplet facts to images and our ImgFact is constructed under this task.

Hence, in this paper, we aim to construct a new large-scale MMKG by grounding triplet facts in an existing symbolic KG on images. These images not only represent entities but also their relations. For example, given a triplet fact (`David_Beckham`, `Spouse`, `Victoria_Beckham`), we expect to find intimate images of `David_Beckham` and `Victoria_Beckham`, as shown in Figure 1.

*These authors contributed equally.
†Corresponding Author.

For this purpose, a straightforward solution is to transfer the entity grounding techniques to our task. However, this solution is inherently insufficient. Existing entity grounding technologies can be mainly divided into two categories. The first is to obtain entity images from online encyclopedias like Wikipedia (Ferrada, Bustos, and Hogan 2017; Wang et al. 2020). However, this method is unsuitable for our task as encyclopedia images mainly describe entities, not triplets. The second involves using a Web search engine like Google and designing an entity-image matching model to select high-quality images from the returned results (Oñoro-Rubio et al. 2017; Liu et al. 2019). However, applying this method directly to the triplet fact grounding task (take the triplet fact as a query) would bring the following problems: **(1)** It fails to identify triplets that cannot be grounded, which ultimately undermines the overall quality of MMKGs. In practice, some triplets may involve non-visualizable entities or relations, such as the entity `Naturalist` or the relation `TimeZone`. **(2)** The entity-image matching model cannot be directly used to measure the alignment between the triplet fact and its associated images. Furthermore, designing triplet-image matching methods is challenging due to the difficulty faced by current language-image models in capturing the deep semantics of relations (e.g., *wedding*, *pregnant*, and *family portrait* for the relation `Spouse`) (Zheng et al. 2021).

To address the above problems, we design a novel pipeline method to ground triplet facts in KGs on images found from a Web search engine. Specifically, for the first problem, we design a multi-modal binary classifier and criteria based on *confidence* and *support* to identify visualizable entities and relations in KGs, respectively. For the second problem, we divide the triplet-image matching process into two steps. The first is the matching of the entity pair (head and tail entities) and image. In this step, we employ a language-image pre-trained model (e.g., CLIP (Radford et al. 2021)) enhanced by language prompts to compute the similarity scores. The second is the matching of the relation and image. We crawl the image title and adopt contrastive learning (CL) (Peng et al. 2020; Hadsell, Chopra, and LeCun 2006) to judge whether the image reflects the relation by calculating the similarities between the triplet and image title.

Contributions. Our contributions are summarized as:

- As far as we know, we are the first to propose triplet facts grounding on images. The most significant characteristic of these images lies in their capacity to convey the visual semantics of relations, a dimension that previous MMKGs failed to capture.
- We construct a new large-scale MMKG (i.e., ImgFact) with a novel pipeline method, which contains 247,732 triplets and 3,730,805 images. Manual and automatic evaluations prove the reliable quality of our ImgFact.
- We use the images from our ImgFact to enhance the model performance on two real-world tasks. In particular, the model optimized by our ImgFact achieves an impressive 8.38% and 9.87% improvement in F1 score over the solutions enhanced by an existing MMKG and VisualChatGPT, respectively, on relation classification.

Related Work

Entity Grounding aims to find images for entities in KGs. Existing methods can be divided into two categories. One way is to obtain entity images from online encyclopedias. MMKGs along this line include IMGpedia (Ferrada, Bustos, and Hogan 2017), Richpedia (Wang et al. 2020), and VisualSem (Alberts et al. 2020). Although these images are of high quality, it is difficult to obtain images for entities not mentioned in the encyclopedia. To address this limitation, another way is to harvest entity images from Web search engines. Since this way easily introduces noisy images, much effort has been devoted to re-designing entity queries by adding entity types (Oñoro-Rubio et al. 2017; Liu et al. 2019) or parent synsets (Deng et al. 2009). MMKGs like ImageGraph (Oñoro-Rubio et al. 2017) and MMKG (Liu et al. 2019) are built in this way. However, previous studies focus on grounding entities to images, ignoring their relations. This paper aims to construct a new MMKG by grounding triplets on images that reflect entities and their relations.

Relation Detection is to localize object pairs in images and classify the relation between them. According to the relation types, existing relation detection datasets can be split into two groups: action relation (e.g., `Ride` and `Eat`) and spatial relation (e.g., `Above` and `On`). Action relation datasets like HICO (Chao et al. 2015) and HICO-DET (Chao et al. 2018) rely on manual labeling for the object bounding boxes. For spatial relations, datasets such as SpatialSense (Yang, Russakovsky, and Deng 2019) and SpatialVOC2K (Belz et al. 2018) are created using manual-based methods. In addition, there are many datasets with images that embody both action and spatial relations, such as Scene Graph (Johnson et al. 2015) and VrR-VG (Liang et al. 2019). However, these studies focus on detecting objects and their relations in images, while our purpose is to ground triplet on images. Moreover, they emphasize shallow semantic relations observed visually. This paper focuses on deep semantic relations (e.g., `Spouse` and `Team`) that are not explicitly expressed in images.

ImgFact Construction

Our goal is to construct a large-scale MMKG named ImgFact by grounding triplet facts (*head entity*, *relation*, *tail entity*), in short (h, r, t), on images. Here, r models the relation between h and t. In this paper, we obtain the raw triplet facts from DBpedia[1] (Lehmann et al. 2015). Based on this KG, we propose a five-step approach, as depicted in Figure 2. 1) Triplet fact filtering. We remove the triplets containing non-visualizable entities or relations because it is difficult to find appropriate images for these triplets. 2) Image retrieving. We retrieve the Top-100 images from a search engine for each entity pair (h, t) in the remaining triplets. 3) Entity-based image filtering. We introduce a language-image pre-trained model enhanced by language prompts to measure the matching between the entity pair and its associated images. 4) Relation-based image filtering. We adopt a method of contrastive learning to determine whether the relation r is reflected in images by computing the similarities between the

[1] In our downloaded version, DBpedia contains 7,195,709 entities, 633 relations, and 21,687,345 triplet facts.

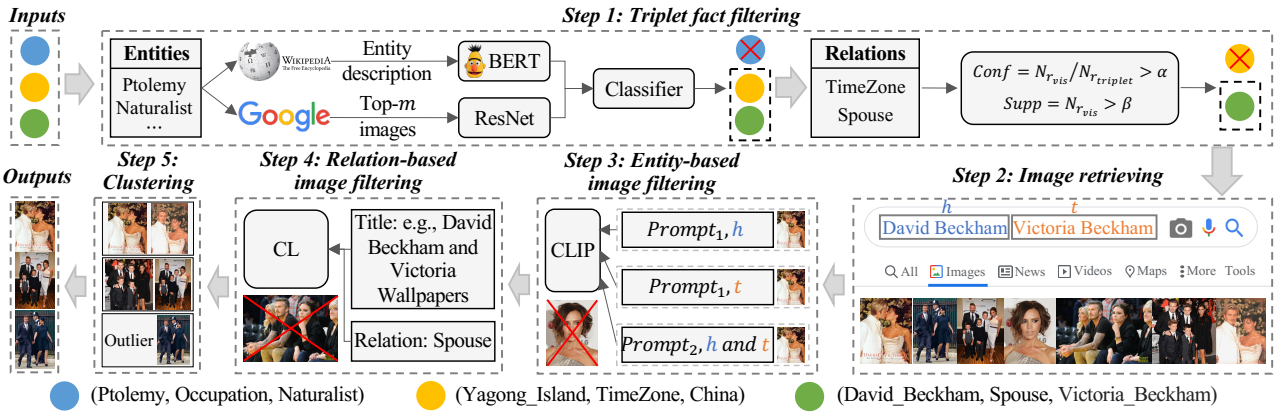

Figure 2: The framework of our ImgFact construction with a five-step pipeline method.

triplet and image titles. 5) Image clustering. For each triplet, we use a clustering algorithm to group the images, and select outliers and Top-1 image in each cluster as the final results.

Triplet Fact Filtering

To remove the triplets containing non-visualizable entities or relations, we design entity filtering and relation filtering.

Entity Filtering. The triplet would be removed if its head or tail entity is non-visualizable because these entities (e.g., `Naturalist`) cannot be accurately characterized in images. To achieve this purpose, we design a binary classification model, where the input is an entity and the output is *true* or *false*. The model is defined as $f: \boldsymbol{x}_e \longrightarrow \pm 1$, where f is a classifier, $\boldsymbol{x}_e$ is the feature vector of the entity e, and +1/-1 represents that the entity e is visualizable or non-visualizable. In this paper, $\boldsymbol{x}_e$ is designed as a concatenation of the image feature vector $\boldsymbol{x}_{e_{img}}$ and text feature vector $\boldsymbol{x}_{e_{text}}$.

To obtain $\boldsymbol{x}_{e_{img}}$, we first collect the Top-m ($m = 20$) images retrieved from Google with a query (i.e., the entity e). Then, we encode each image p_i with ResNet (He et al. 2016) to obtain its representation $\boldsymbol{p}_i$. We finally concatenate these representations as $[\boldsymbol{p}_1; ...; \boldsymbol{p}_m]$ and feed it into a linear layer to obtain the representation $\boldsymbol{x}_{e_{img}}$ of the images. That is,

$$\begin{aligned} \boldsymbol{p}_i &= ResNet\left([p_i]\right), \quad i = 1, ..., m, \\ \boldsymbol{x}_{e_{img}} &= \boldsymbol{W}_{img}[\boldsymbol{p}_1; ...; \boldsymbol{p}_m] + \boldsymbol{b}_{img}, \end{aligned} \tag{1}$$

where $\boldsymbol{p}_i \in \mathbb{R}^{d_u}$, $\boldsymbol{W}_{img} \in \mathbb{R}^{d_v \times m \cdot d_u}$ and $\boldsymbol{b}_{img} \in \mathbb{R}^{d_v}$ are the weight matrices. We set $d_u = 256$ and $d_v = 256$.

To obtain $\boldsymbol{x}_{e_{text}}$, we collect the first sentence (often the entity definition) of the Wikipedia entry of the entity e. The text with the placeholders (i.e., $\langle CLS \rangle$ and $\langle SEP \rangle$) is encoded by BERT (Devlin et al. 2018). The final layer representation of $\langle CLS \rangle$ is taken as the text representation and also fed into a linear layer to obtain $\boldsymbol{x}_{e_{text}}$. This process is defined as

$$\begin{aligned} \boldsymbol{x}'_{e_{text}} &= BERT([\langle CLS \rangle, w_1, ..., w_l, \langle SEP \rangle]), \\ \boldsymbol{x}_{e_{text}} &= \boldsymbol{W}_{text}\boldsymbol{x}'_{e_{text}}[0] + \boldsymbol{b}_{text}, \end{aligned} \tag{2}$$

where $[w_1, ..., w_l]$ is a sequence of tokens in the text description, $\boldsymbol{x}'_{e_{text}}[0] \in \mathbb{R}^{d_o}$ is the representation of $\langle CLS \rangle$, $\boldsymbol{W}_{text} \in \mathbb{R}^{d_s \times d_o}$ and $\boldsymbol{b}_{text} \in \mathbb{R}^{d_s}$ are the learnable weight matrices of the linear layer. We set $d_s = 256$ and $d_o = 768$.

After obtaining $\boldsymbol{x}_{e_{img}}$ and $\boldsymbol{x}_{e_{text}}$, we employ a linear layer and a Softmax function to build a multi-modal image-text classifier f that takes $\boldsymbol{x}_e = \boldsymbol{x}_{e_{img}} \oplus \boldsymbol{x}_{e_{text}}$ as the input and output +1/-1. If the head or tail entity of a triplet fact is judged as "-1" by the classifier, we remove this triplet. Since there is no labeled data to train the classifier, we randomly select 3,000 entities from DBpedia and ask three volunteers to label them. If at least two annotators consider that more than 10 images in the Top-20 reflect the corresponding entity, we label it as 1. Otherwise, it is marked as 0. The dataset is constructed, containing 1,566 and 1,434 positive and negative entities, respectively, and then randomly split into training, validation, and test sets with 8:1:1, where the Fleiss' kappa (Fleiss 1971) is 0.782, showing substantial agreement among these annotators. After training on labeled data, the model achieves 82% accuracy and 85% recall on the test set. Although the accuracy is not very high, it is acceptable because there are three more steps (Relation filtering, Entity-based image filtering, and Relation-based image filtering) to discard the triplets that cannot be matched with images. Finally, we utilize the trained model to predict labels of the remaining entities and remove the triplets containing non-visualizable entities. There remain 1,776,872 entities, 653 relations, and 4,146,669 triplet facts.

Relation Filtering. Although the previous sub-step discards many triplets, there are still numerous noises. For instance, the triplet (`Yagong_Island`, `TimeZone`, `China`) retained in the previous step is unable to be visualized because `TimeZone` cannot be characterized by images. Hence, we need to remove the triplets with non-visualizable relations.

According to our observation, the relation cannot be grounded if most of its entity pairs are non-visualizable. Consequently, the previous triplet would be removed since the relation `TimeZone` is often associated with non-visualizable head and tail entities, such as (`Algoma_District`, `TimeZone`, `Eastern_Time_Zone`) and (`Jubeiha_area`, `TimeZone`, `UTC`). To capture this regularity, we design *confidence* and *support* metrics, i.e.,

$$Conf = \frac{N_{r_{vis}}}{N_{r_{triplet}}}, \quad Supp = N_{r_{vis}}. \tag{3}$$

Here, $N_{r_{vis}}$ denotes the number of triplets whose head and

tail entities linked by the relation r are both visualizable, and $N_{r_{triplet}}$ is the total number of triplets with r. *Support* is used to prevent high confidence scores of long-tail non-visualizable relations. Only relations with *confidence* and *support* above the pre-determined thresholds (0.15 and 50 in our experiment) are retained, and others are removed. This step produces 142 remaining relations. Since the number of remaining relations is limited, we further ask the annotators to guarantee the relation quality. If the relation is reflected in more than half of the Top-20 retrieved images, it is marked as 1. Otherwise, we label it as 0. After human labeling, there remain 64 relations (1,776,872 entities and 1,502,722 triplet facts), where the Fleiss' kappa on this task is 0.802, showing substantial agreement among annotators.

Image Retrieving

After filtering non-visualizable triplet facts, we collect images for the remaining ones. To this end, we develop a distributed web crawler deployed on multiple machines to obtain images from a search engine (i.e., Google). For each triplet, we take the head and tail entities separated by a space as the search query and collect Top-100 images retrieved by the crawler. The relation is omitted in this procedure since introducing the relation (e.g., `DraftTeam` and `AssociatedBand`) would cause a puzzle for the search engine. As some pairs have fewer than 100 images on Google, we obtain a total of 90,716,130 images for 1,502,722 triplet facts.

Entity-based Image Filtering

Since the returned images may not reflect the head and tail entities, we select high-quality images from Top-100 for each entity pair. To this end, we calculate the similarities between 1) the head entity and image, 2) the tail entity and image, and 3) the entity pair and image. The image is kept if all three similarities are above the pre-defined thresholds.

To compute the similarities, we employ a language-image pre-trained model named CLIP (Radford et al. 2021) as the matching model. Specifically, we encode the entity (pair) and image with CLIP and output their similarity. To improve the matching quality, we convert the entity (pair) into a sentence. A straightforward method is to use manually pre-defined templates, e.g., "the picture of h (t, or h and t)". However, this template is too rigid, so we introduce language prompts as trainable templates. Similar to (Li and Liang 2021), we add several virtual tokens in front of the entity (pair) as the language prompts. The prompts with the entity (pair) are considered as the input of the text encoder in CLIP:

$$\begin{aligned} T_h = [q_1, ..., q_a, h], \quad T_t = [q_1, ..., q_a, t], \\ T_{(h,t)} = [g_1, ..., g_a, h, and, t], \end{aligned} \tag{4}$$

where $[q_1, ..., q_a]$ and $[g_1, ..., g_a]$ are the prompts for the head-/tail entity and entity pair respectively, and $a = 8$ is the number of virtual tokens in prompts. To keep a balance between effectiveness and efficiency, we freeze the parameters of the CLIP model and only train the prompts. To train the model, we need a labeled dataset. Since only a small number of parameters need to be trained, we randomly select 1,800 triplet-image pairs and invite the previous annotators to label them as 1 or 0 with a voting mechanism, which indicates whether entity pairs are reflected in images. In this way, the built dataset contains 727 positive and 1,073 negative triplet-images pairs and is split into training, validation, and test sets according to 8:1:1, where the Fleiss' kappa among the annotators on this task is 0.796. After training our model on the labeled data, we determine the three similarity thresholds by maximizing the accuracy of the validation set. The threshold for the matching of the head/tail entity and image is set to 0.87, and it is 0.5 for the entity pair and image. Our model reaches 92% precision on the test set. Finally, we obtain 8,644,407 images for 540,145 triplet facts (281,284 entities and 64 relations). Each triplet has an average of 16 images, providing enough data to construct a large-scale MMKG.

Relation-based Image Filtering

Although we have images with the head and tail entities, they may not reflect the relation r between them. Hence, we need to select images depicting r. To this end, we crawl the image title and calculate the similarity between the triplet fact and title to measure whether the relation is represented in the image. If the title is semantically similar to the triplet, we consider the image to be appropriate. To model this similarity, we adopt contrastive learning (Peng et al. 2020; Liu et al. 2022b), which aims to learn similar representations for "neighbors" and dissimilar representations for "non-neighbors". In our case, the triplet and image title are defined as neighbors if they are semantically similar. Otherwise, they are non-neighbors.

Given a triplet (h, r, t) and its matching image title c_t, we denote c_{pos} and c_t as neighbors, where c_{pos} is a positive sentence "h's r is t" transformed from (h, r, t). c^i_{neg} $(i = 1, ..., N)$ and c_t are considered non-neighbors, where c^i_{neg} is the negative sentence transformed from (h, r', t) $(r' \neq r, r' \in R)$, N is the number of wrong triplets, and R is the set of 64 relations. In the CL model, we employ two BERT as the sentence encoder $Enc_b(\cdot)$ and title encoder $Enc_t(\cdot)$ to learn the representations of the inputs. Then, we adopt the final hidden states of the $\langle CLS \rangle$ tokens as the representations of C_b and c_t, where $C_b = \{c_{pos}\} \cup \{c^i_{neg}\}^N_{i=1}$. That is,

$$\boldsymbol{c} = \begin{cases} Enc_b(c), & c \in C_b \\ Enc_t(c), & c = c_t \end{cases}, \tag{5}$$

where $\boldsymbol{c} \in \mathbb{R}^{d_y}$ $(d_y = 768)$. Based on the representations, we have the following training objective:

$$\mathcal{L} = -\log \frac{\exp(\boldsymbol{c}_t^T \cdot \boldsymbol{c}_{pos}/\tau)}{\exp(\boldsymbol{c}_t^T \cdot \boldsymbol{c}_{pos}/\tau) + \sum_{i=1}^N \exp(\boldsymbol{c}_t^T \cdot \boldsymbol{c}^i_{neg}/\tau)}, \tag{6}$$

where $\tau = 0.1$ is a temperature hyper-parameter and $N = 15$. Since there is no open-source dataset to train the CL model, we randomly select 4,000 triplet-title pairs and ask the previous annotators to label them with 1 or 0, indicating if the title reflects the relation in the triplet fact. We integrate the annotation results according to the voting mechanism and the dataset is split into training, validation, and test sets in a ratio of 8:1:1, where the Fleiss' kappa on this task is 0.832. After training the CL model on the labeled data, we use the model to calculate the similarities between the image

MMKG	# Img	# GE	# GR	# GT
IMGpedia	14,765,300	14,765,300	-	-
ImageGraph	829,931	14,870	-	-
Richpedia	2,883,162	30,638	-	-
MMKG	37,479	45,011	-	-
VisualSem	938,100	89,896	-	-
ImgFact (Ours)	3,730,805	182,589	64	247,732

Table 1: Statistics of ImgFact and typical MMKGs. GE, GR, and GT denote grounded entities, relations, and triplets.

title and triplets (the candidate one and corrupted ones with wrong relations). If the matching of the title and candidate triplet ranks in Top-3, we consider the relation in the triplet to be reflected in the image. In this setting, our CL model achieves 98% accuracy and 90% recall on the test set, and this step finally produces 182,589 entities, 64 relations, 247,732 triplets, and 5,086,535 candidate images.

Image Clustering

After image filtering, we notice that some triplet facts have many similar images. To show image diversity and prevent redundancy, we employ DBSCAN (Ester et al. 1996) for image clustering within each triplet. DBSCAN is chosen because it does not require a predetermined number of clusters. The algorithm ensures that each image in a cluster has a minimum number (*MinPts*) of neighbors within a given radius (*Eps*). In this paper, we define $\{p'_2 \in P' | dist\,(p'_1, p'_2) \leq Eps\}$ as the *Eps*-neighborhood of the image p'_1, where P' is the remaining images for a given triplet fact and $dist\,(p'_1, p'_2) = 1 - cos\,(p'_1, p'_2)$. $cos\,(p'_1, p'_2)$ denotes the cosine similarity between the vectors of p'_1 and p'_2 obtained by the VGG algorithm (Simonyan and Zisserman 2014). In DBSCAN, we set $MinPts = 1$ and $Eps = 0.25$. The reason for setting $MinPts = 1$ is that our clustering aims to show image diversity and avoid redundancy rather than image removal. After clustering, we select images with the highest scores (the average of the three similarities defined in Section Entity-based Image Filtering) in each cluster as the final results. Outlier images are also retained. Overall, this step reserves 182,589 entities, 64 relations, 247,732 triplets, and 3,730,805 images.

ImgFact Analysis

To understand the properties of ImgFact, we analyze the MMKG from the following three aspects: dataset statistics, image quality, and image diversity.

ImgFact Statistics. As reported in Table 1, ImgFact contains 182,589 entities, 64 relations, 247,732 triplets, and 3,730,805 images. On average, each triplet has 15 images. Unlike existing MMKGs, ImgFact's images are tailored to triplets rather than just entities. In addition, the distributions of entity categories and relations are shown in Figure 3.

Image Quality. To evaluate image quality, we employ human and automatic evaluations. For human evaluation, we randomly select 500 triplet facts and their images. Three volunteers (not the same volunteers mentioned in the construction process) are invited to score each triplet-image pair

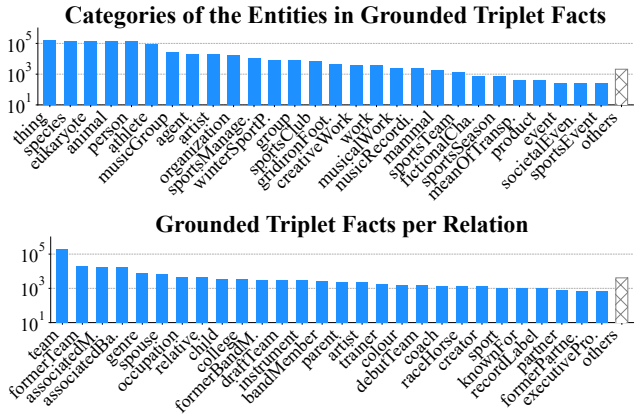

Figure 3: The distributions of entity categories and relations. *others* in the upper and lower histograms contain 74 entity categories and 36 relations, respectively.

based on whether the head entity, relation, and tail entity are all reflected in the image. The images for each triplet fact are sorted according to the average of the three similarities defined in Section Entity-based Image Filtering. We report three metrics: the accuracy of the pairs and the proportions of correct images in Top-1 (H@1) and Top-3 (H@3). After human evaluation, ImgFact achieves 80.6%, 83.2%, and 92.4% on accuracy, H@1, and H@3. The Fleiss' kappa (Fleiss 1971; Liu et al. 2022a) on the above metrics is 0.853, 0.887, and 0.783, respectively, indicating substantial evaluators' agreement. Additionally, automatic evaluations (see Section "Experiments") on link prediction and relation classification also demonstrate ImgFact's reliable quality.

Image Diversity. To show image diversity, we randomly select 200 triplets' images and calculate the proportions of the similar image pairs to all pairs within each cluster and between clusters. The similarity is determined by the voting mechanism of three evaluation volunteers. When two images are similar on the pixel level, we label them 1. Otherwise, we mark it as 0. The average similarity of intra-cluster and inter-cluster are 0.94 and 0.05, respectively. The Fleiss' kappa on this task is 0.790, indicating substantial agreement. These results show the diversity of our ImgFact.

Experiments

In this section, we utilize the link prediction and relation classification tasks to verify the quality of our ImgFact. Furthermore, we utilize images from our ImgFact to enhance model performance on these two tasks.

Link Prediction

Link prediction aims to predict the missing head or tail entity for a given triplet (Liu et al. 2021). For each test sample, we first remove the head or tail entity and replace it with all entities in the dataset. Then, we rank these entities in descending order according to predicted scores and record their ranking. In this task, we report three metrics: the proportion of correct entities ranked in Top-1 (H@1) and Top-10 (H@10), and

Method	Input	Predicting Head			Predicting Tail		
		H@1	H@10	MRR	H@1	H@10	MRR
BERT	h-r / r-t	0.29	2.00	1.40	12.04	41.79	22.64
BERT +ResNet50	+ p_{noise}	0.05	0.49	0.36	9.34	37.55	18.62
	+ p	**0.59**	**3.37**	**1.88**	**14.76**	**48.89**	**26.65**
ViLT	+ p_{noise}	0.05	0.44	0.30	8.42	35.22	17.11
	+ p	**0.36**	**2.34**	**1.39**	**13.61**	**45.93**	**24.81**

Table 2: The automatic evaluation (%) of our ImgFact on link prediction. "h-r" means the prediction of t given h and r.

the mean reciprocal rank (MRR) of all correct entities. Since ImgFact has fewer triplet facts in existing public datasets (e.g., only about 600 triplets matched with DB15K (Sun, Hu, and Li 2017)), this would lead to insufficiently valid results. Hence, we construct a new dataset from our ImgFact for automatic evaluation. The construction principle is that entities and relations in both validation and test sets need to appear in the training set. Based on this principle, a dataset (denoted as D_L) is built, containing 3,340, 717, and 716 positive triplets for training, validation, and testing, respectively.

To verify whether the image p reflects h and t in (h, r, t), we design an A/B testing. We take the prediction of the tail entity t as an example. Experiment A_1 uses h and r as the input, and experiment B_1 has two kinds of inputs: 1) h, r, and the image p of (h, r, t), and 2) h, r, and the image p_{noise} of another randomly selected triplet. In experiment A_1, a BERT-based classifier is trained on the input "h's r is [MASK]" using the $\langle CLS \rangle$ representation. In experiment B_1, we employ (BERT+ResNet50)- and ViLT-based (Kim, Son, and Kim 2021) classifiers to predict t, respectively. Notably, the parameters of BERT, ResNet50, and ViLT are fixed and only classifier parameters are updated. The results are reported in Table 2. We notice that both BERT+ResNet50 and ViLT with p outperform BERT, showing that the image encoded by the encoders is helpful for the task. Besides, both methods with p_{noise} perform worse than BERT, proving that the previous improvement is mostly due to the image p rather than the added image encoders. Hence, it is reasonable to infer that the image embodies the head and tail entities.

To evaluate our ImgFact is useful for link prediction, we also design an A/B testing for predicting t. Note that the experiments to predict h are not designed due to numerous corresponding h for a given (r, t) so the model cannot be trained effectively. For example, given (?, `Nationality`, `America`), we can replace "?" with hundreds of millions of American names. For the A/B testing, the input of experiment A_2 is h and r, while there are five kinds of inputs for experiment B_2: 1) h, r, and the image p'_h of h from MMKG (Liu et al. 2019), 2) h, r, and the image p^*_r[2] of (h_1, r, t_1) generated by VisualChatGPT (VCG), where $h_1 \neq h$ and $t_1 \neq t$, 3) h, r, and the images p'_h and p^*_r generated by VCG, 4) h, r, and the image p_r of (h_2, r, t_2) from ImgFact, where $h_2 \neq h$ and $t_2 \neq t$, and 5) h, r, and the images p'_h and p_r. In experiments,

[2]The input of VisualChatGPT used to generate images for triplets in D_L: "please generate an image of [head]'s [relation] is [tail]".

Method	Input	H@1	H@10	MRR
BERT	(h, r)	12.04	41.79	22.64
BERT+ResNet50	+ p'_h	12.95	42.31	23.16
	+ p^*_r	13.04	47.02	24.33
	+ p'_h & p^*_r	13.53	48.38	25.07
	+ p_r	14.12	48.89	25.34
	+ p'_h & p_r	**14.68**	**49.01**	**26.03**
ViLT	+ p'_h	12.05	42.96	22.87
	+ p^*_r	12.01	42.97	21.47
	+ p'_h & p^*_r	12.77	44.31	23.94
	+ p_r	12.91	43.07	23.04
	+ p'_h & p_r	**13.54**	**45.23**	**24.56**

Table 3: Results (%) of integrating images on link prediction. p'_h, p^*_r, and p_r are derived from MMKG (Liu et al. 2019), VisualChatGPT (Wu et al. 2023a), and ImgFact, respectively.

we employ a BERT-based classifier for A_2, while we utilize (BERT+ResNet50)- and ViLT-based classifiers for B_2. The results are listed in Table 3. We observe that regardless of whether the models incorporate the entity image (p'_h) from MMKG in advance, p^*_r and p_r can further improve the model performance, indicating that the models have learned the relation semantics from p^*_r and p_r. The reason is that tail entities of different triplet facts with the same relation often belong to the same domain, and additional images p^*_r and p_r can provide more information (e.g., entity type) about the tail entity. In addition, when comparing images generated by VCG with those in our ImgFact, our models show a noteworthy improvement. This indicates that the quality of our obtained images surpasses those generated by VCG. While most of VCG's generated images can indeed reflect relations, they still suffer from a notable problem concerning entities. This problem, known as *distortion error*, results in images appearing unnaturally deformed, as depicted in Figure 4.

Relation Classification

In this paper, the task is to assign a semantic relation from a pre-defined set to a pair of entities. We use D_L as the dataset and report accuracy (Acc), weighted precision (w-Pre), w-recall (w-Rec), and w-F1 due to the label imbalance.

To evaluate whether the relation r in (h, r, t) is reflected in the image p, we design an A/B testing. Experiment A_3 takes h and t as the input, and experiment B_3 has two kinds of inputs: 1) h, t, and the image p of (h, r, t), and 2) h, t, and the image p_{noise}. Similar to link prediction, we employ BERT-, (BERT+ResNet50)-, and ViLT-based classifiers, where the input is "h and t" and the output is a relation. The results are shown in Table 4. We observe that BERT outperforms both BERT+ResNet50 and ViLT with p_{noise}, but it still underperforms the methods with p. Similar to link prediction, it is reasonable to infer that the image reflects the relation.

To demonstrate our ImgFact is helpful for relation classification, we still design an A/B testing. A_4 uses h and t as the input, while experiment B_4 has six kinds of inputs:

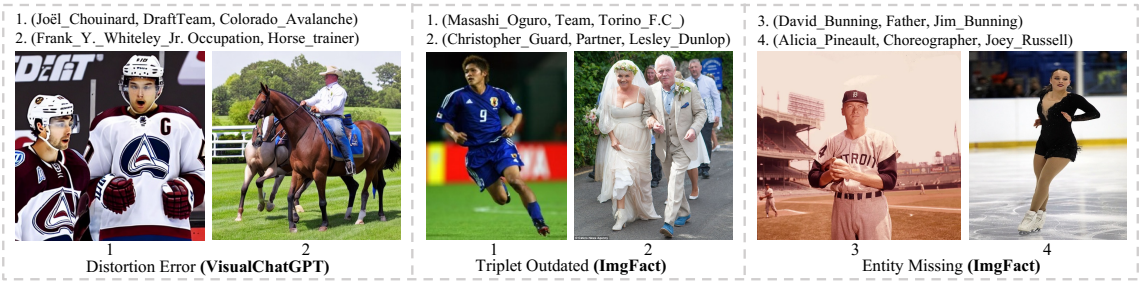

Figure 4: Noisy images generated by VisualChatGPT and two kinds of image noises in ImgFact.

Method	Input	Acc	w-Pre	w-Rec	w-F1
BERT	(h, t)	46.70	74.72	46.70	44.00
BERT +ResNet50	$+ p_{noise}$	27.88	42.58	27.88	27.23
	$+ p$	**57.00**	**76.76**	**57.00**	**58.63**
ViLT	$+ p_{noise}$	30.61	50.05	30.61	32.25
	$+ p$	**67.79**	**81.48**	**67.79**	**70.73**

Table 4: The automatic evaluation (%) of our ImgFact on relation classification.

1) h, t, and the image p'_h of h from (Liu et al. 2019), 2) h, t, and the image p^*_h of (h, r_3, t_3) generated by VCG, 3) h, t, and the image p_h of (h, r_4, t_4) from ImgFact, 4) h, t, and the image p'_t of t from (Liu et al. 2019), 5) h, t, and the image p^*_t of (h_5, r_5, t) generated by VCG, 6) h, t, and the image p_t of (h_6, r_6, t) from ImgFact. The results are reported in Table 5. We observe that the two models with the image from MMKG, VCG, or ImgFact outperform BERT, showing that the images can be used to improve the model performance. Notably, despite VCG's generated images suffering from distortion errors, their impact on the relation task is relatively limited. These errors tend to manifest in minor details rather than affecting the overall semantics of the images. For instance, the man's appearance in the first picture of Figure 4 is vague and the uniforms' logos are distorted, but we can still readily recognize that the image depicts people playing ice hockey in a professional team. In addition, the models that incorporate the images in MMKG outperform the ones incorporating images produced by VCG. However, it still falls short compared to the models utilizing images from our ImgFact. Specifically, the (BERT+ResNet50)-based classifier incorporating p_t achieves an impressive 8.38% and 9.87% improvement over the classifier with p'_t and p^*_t on the F1, respectively. The reason is that relations present in ImgFact for the same head or tail entity tend to be highly relevant. For instance, in ImgFact, the relations of the head entity `Ben_Williams_(musician)` include `associatedMusicalArtist` and `associatedBand`, which are related to each other.

Method	Input	Acc	w-Pre	w-Rec	w-F1
BERT	(h, t)	46.70	74.72	46.70	44.00
BERT +ResNet50	$+ p'_h$	68.15	76.69	68.16	69.40
	$+ p^*_h$	67.23	77.31	63.47	66.79
	$+ p_h$	**68.59**	**77.69**	**68.60**	**69.97**
	$+ p'_t$	60.47	70.18	62.34	64.78
	$+ p^*_t$	61.32	71.35	61.28	63.29
	$+ p_t$	**68.32**	**81.93**	**80.25**	**73.16**
ViLT	$+ p'_h$	63.14	**79.46**	62.78	64.73
	$+ p^*_h$	62.41	77.25	63.20	63.92
	$+ p_h$	**67.84**	79.33	**65.52**	**68.18**
	$+ p'_t$	63.32	72.23	63.55	66.34
	$+ p^*_t$	62.74	71.31	60.81	63.86
	$+ p_t$	**64.25**	**74.34**	**64.61**	**69.65**

Table 5: The results (%) of incorporating images on relation classification. p'_h (p'_t), p^*_h (p^*_t), and p_h (p_t) are derived from MMKG (Liu et al. 2019), VisualChatGPT (Wu et al. 2023a) and our ImgFact, respectively.

Conclusion and Discussion

In this paper, we aim to construct a new large-scale MMKG by grounding triplet facts on images, where the images reflect not only entities but also their relations. To this end, we propose a novel pipeline method with five steps: triplet fact filtering, image retrieving, entity-based image filtering, relation-based image filtering, and image clustering. In this way, ImgFact is built and contains 247,732 triplets and 3,730,805 images. Manual and automatic evaluations verify the reliability of ImgFact's quality. Furthermore, experiments also demonstrate that ImgFact is helpful for real-world tasks.

Limitations. ImgFact still suffers from two kinds of image noises: 1) Triplet outdated, where the triplet in DBPedia is inconsistent with the current real-world fact. As shown in Figure 4, the current team of `Masashi_Oguro` is `Tochigi_SC`, and `Torino_F.C_` is one of his former teams. This noise arises because the information in the KG is not updated in real-time, causing some triplets to become outdated. 2) Entity missing, where the images lack the presence of either the head or tail entity. As shown in Figure 4, the image of (`David_Bunning`, `Father`, `Jim_Bunning`) only contains `Jim_Bunning` and misses `David_Bunning`. This error may be attributed to that not all entities mentioned in the pre-training data (i.e., image-text pairs) of the CLIP model are explicitly depicted in images.

Ethical Statement

There are two main ethical considerations: 1) Since our images are sourced from the Internet, there is a risk of harmful images (e.g., personal privacy information, racist language, and violent images). Fortunately, our MMKG is free of this ethical issue for two reasons. First, Google Image Search has its own SafeSearch content filter to automatically remove dirty content. Second, we randomly select 30,000 images from ImgFact and assign them to the volunteers mentioned above for manual evaluation. The results indicate no harmful images in the testing samples. 2) Since our images are crawled from Google, we need to check the copyright issue for data sharing. In Google Image Search, we set the "usage rights" field to "Creative Commons licenses". Hence, the images in our ImgFact can be distributed under a Creative Commons license. In summary, ImgFact is released under Creative Commons Non-Commercial (CC-NC), prohibiting commercial use of our MMKG.

Acknowledgments

This paper was supported by the National Natural Science Foundation of China (No. 62306112), Shanghai Sailing Program (No. 23YF1409400), and Science and Technology Commission of Shanghai Municipality Grant (No. 22511105902).

References

Alberts, H.; Huang, T.; Deshpande, Y.; Liu, Y.; Cho, K.; Vania, C.; and Calixto, I. 2020. VisualSem: a high-quality knowledge graph for vision and language. *arXiv preprint arXiv:2008.09150*.

Belz, A.; Muscat, A.; Anguill, P.; Sow, M.; Vincent, G.; and Zinessabah, Y. 2018. Spatialvoc2k: A multilingual dataset of images with annotations and features for spatial relations between objects. In *Proceedings of the 11th International Conference on Natural Language Generation*, 140–145.

Chao, Y.-W.; Liu, Y.; Liu, X.; Zeng, H.; and Deng, J. 2018. Learning to detect human-object interactions. In *2018 ieee winter conference on applications of computer vision (wacv)*, 381–389. IEEE.

Chao, Y.-W.; Wang, Z.; He, Y.; Wang, J.; and Deng, J. 2015. Hico: A benchmark for recognizing human-object interactions in images. In *Proceedings of the IEEE International Conference on Computer Vision*, 1017–1025.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255. IEEE.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Ester, M.; Kriegel, H.-P.; Sander, J.; Xu, X.; et al. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In *kdd*, volume 96, 226–231.

Ferrada, S.; Bustos, B.; and Hogan, A. 2017. IMGpedia: a linked dataset with content-based analysis of Wikimedia images. In *International Semantic Web Conference*, 84–93. Springer.

Fleiss, J. L. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5): 378.

Hadsell, R.; Chopra, S.; and LeCun, Y. 2006. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, 1735–1742. IEEE.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Johnson, J.; Krishna, R.; Stark, M.; Li, L.-J.; Shamma, D.; Bernstein, M.; and Fei-Fei, L. 2015. Image retrieval using scene graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3668–3678.

Kim, W.; Son, B.; and Kim, I. 2021. Vilt: Vision-and-language transformer without convolution or region supervision. In *International Conference on Machine Learning*, 5583–5594. PMLR.

Lehmann, J.; Isele, R.; Jakob, M.; Jentzsch, A.; Kontokostas, D.; Mendes, P. N.; Hellmann, S.; Morsey, M.; Van Kleef, P.; Auer, S.; et al. 2015. Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic web*, 6(2): 167–195.

Li, X. L.; and Liang, P. 2021. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*.

Liang, Y.; Bai, Y.; Zhang, W.; Qian, X.; Zhu, L.; and Mei, T. 2019. Vrr-vg: Refocusing visually-relevant relationships. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 10403–10412.

Liu, J.; Chen, T.; Wang, C.; Liang, J.; Chen, L.; Xiao, Y.; Chen, Y.; and Jin, K. 2022a. VoCSK: Verb-oriented commonsense knowledge mining with taxonomy-guided induction. *Artificial Intelligence*, 310: 103744.

Liu, J.; Liu, J.; Chen, L.; Liang, J.; Xiao, Y.; Xu, H.; Zhang, F.; Wang, Z.; and Xie, R. 2022b. Noun Compound Interpretation With Relation Classification and Paraphrasing. *IEEE Transactions on Knowledge and Data Engineering*.

Liu, J.; Wang, M.; Wang, C.; Liang, J.; Chen, L.; Jiang, H.; Xiao, Y.; and Chen, Y. 2021. Learning Term Embeddings for Lexical Taxonomies. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 6410–6417.

Liu, Y.; Li, H.; Garcia-Duran, A.; Niepert, M.; Onoro-Rubio, D.; and Rosenblum, D. S. 2019. MMKG: multi-modal knowledge graphs. In *European Semantic Web Conference*, 459–474. Springer.

Oñoro-Rubio, D.; Niepert, M.; García-Durán, A.; González, R.; and López-Sastre, R. J. 2017. Answering Visual-Relational Queries in Web-Extracted Knowledge Graphs. *arXiv preprint arXiv:1709.02314*.

Peng, H.; Gao, T.; Han, X.; Lin, Y.; Li, P.; Liu, Z.; Sun, M.; and Zhou, J. 2020. Learning from Context or Names? An Empirical Study on Neural Relation Extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 3661–3672.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*.

Simonyan, K.; and Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Sun, Z.; Hu, W.; and Li, C. 2017. Cross-lingual entity alignment via joint attribute-preserving embedding. In *International Semantic Web Conference*, 628–644. Springer.

Wang, M.; Wang, H.; Qi, G.; and Zheng, Q. 2020. Richpedia: a large-scale, comprehensive multi-modal knowledge graph. *Big Data Research*, 22: 100159.

Wu, C.; Yin, S.; Qi, W.; Wang, X.; Tang, Z.; and Duan, N. 2023a. Visual chatgpt: Talking, drawing and editing with visual foundation models. *arXiv preprint arXiv:2303.04671*.

Wu, Y.; Wu, X.; Li, J.; Zhang, Y.; Wang, H.; Du, W.; He, Z.; Liu, J.; and Ruan, T. 2023b. MMpedia: A Large-Scale Multi-modal Knowledge Graph. In *International Semantic Web Conference*, 18–37. Springer.

Yang, K.; Russakovsky, O.; and Deng, J. 2019. Spatialsense: An adversarially crowdsourced benchmark for spatial relation recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2051–2060.

Zheng, C.; Wu, Z.; Feng, J.; Fu, Z.; and Cai, Y. 2021. MNRE: A Challenge Multimodal Dataset for Neural Relation Extraction with Visual Evidence in Social Media Posts. In *2021 IEEE International Conference on Multimedia and Expo (ICME)*, 1–6. IEEE.

Zhu, X.; Li, Z.; Wang, X.; Jiang, X.; Sun, P.; Wang, X.; Xiao, Y.; and Yuan, N. J. 2022. Multi-Modal Knowledge Graph Construction and Application: A Survey. *arXiv preprint arXiv:2202.05786*.