This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

RelTransformer: A Transformer-Based Long-Tail Visual Relationship Recognition

Jun Chen[1], Aniket Agarwal[1,2], Sherif Abdelkarim[1], Deyao Zhu[1], Mohamed Elhoseiny[1]
[1]King Abdullah University of Science and Technology
[2] Indian Institute of Technology
{jun.chen,deyao.zhu,mohamed.elhoseiny}@kaust.edu.sa
aagarwal@ma.iitr.ac.in, sherif.abdelkarim91@gmail.com

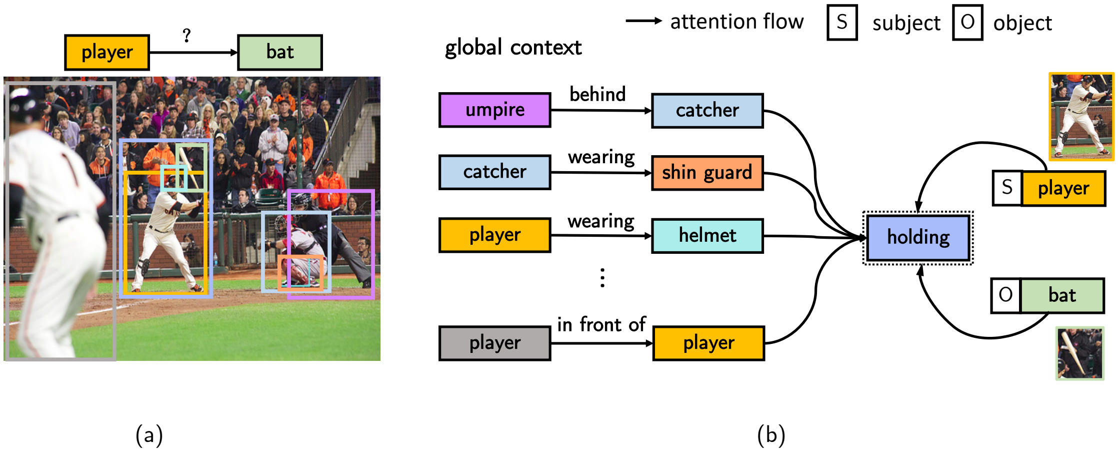


Figure 1. (a) Given an image with several annotated objects, the objective is to predict the visual relationship between the image region of "player" and "bat". (b) It illustrates our message-passing strategy: we attentively aggregate visual features from all the triplets in the global context, the subject, and object to the "holding" feature via attention. All the boxes above denote the corresponding visual features.

Abstract

The visual relationship recognition (VRR) task aims at understanding the pairwise visual relationships between interacting objects in an image. These relationships typically have a long-tail distribution due to their compositional nature. This problem gets more severe when the vocabulary becomes large, rendering this task very challenging. This paper shows that modeling an effective message-passing flow through an attention mechanism can be critical to tackling the compositionality and long-tail challenges in VRR. The method, called RelTransformer, represents each image as a fully-connected scene graph and restructures the whole scene into the relation-triplet and global-scene contexts. It directly passes the message from each element in the relation-triplet and global-scene contexts to the target relation via self-attention. We also design a learnable memory to augment the long-tail relation representation learning. Through extensive experiments, we find that our model generalizes well on many VRR benchmarks. Our model outperforms the best-performing models on two large-scale long-tail VRR benchmarks, VG8K-LT (+2.0% overall acc) and GQA-LT (+26.0% overall acc), both having a highly skewed distribution towards the tail. It also achieves strong results on the VG200 relation detection task. Our code is available at https://github.com/Vision-CAIR/RelTransformer.

1. Introduction

The Visual Relationship Recognition (VRR) task goes beyond recognizing individual objects by comprehensively understanding relationships between interacting objects in a visual scene. Owing to the enriched scene understanding provided by VRR, it benefits various other vision tasks such as image captioning (e.g., [10,46,47]), VQA (e.g., [15,40]), image generation (e.g., [16]), and 3D scene synthesis (e.g.,

[31]). However, due to the imbalanced class distribution in many VRR datasets [15, 20], predictions of the most existing models are dominated by the head/frequent relations, lacking generalization on tail/low-shot relationships.

Many previous approaches characterize the VRR problem under a graph scenario. Popular graph-based methods iteratively pass massages from other direct or indirect nodes to the relation along with the structure of the graph using the long short-term memory [39, 44, 50], or graph attention networks [24, 45]. However, the graph structure may implicitly constraint the relation to focus on its nearby neighbors. This phenomenon has been observed in recent works [2, 43], showing that graph neural networks incline to pay most attention to the local surrounding nodes but could not benefit much from distant nodes [2], and node representations will become indistinguishable if there are many layers [43]. Such problems can also be seen in long short-term memory networks due to their iterative message-passing learning nature. However, the target relation can also benefit a lot from the distant nodes. e.g., when we predict the "holding" relationship between "player" and "bat" in Fig. 1, the distant objects such as "catcher" and "umpire" can provide a context that the "player" is in a baseball game, and those can help the model better predict "holding" relationship.

To alleviate the aforementioned problems, we propose to adapt self-attention mechanisms originally introduced in Transformer [41] to tackle the VRR challenges. Self-attention can be viewed as a non-local mean operation [3], which computes the weighted average of all the inputs. When it applies to the VRR problem, it assumes that the relation has a full connection with all the other nodes in the graph and directly passes the messages among them via attention. In contrast to GNN/LSTM approaches, this strategy can allow the relation to have a larger attention scope and pass the message regardless of the graph structure or spatial constraints. It also avoids the valuable information from distant nodes being suppressed by nearby neighbors. Hence, each relation can selectively attend to its relevant features without spatial constraints and learn a richer-contextualized representation, which can benefit the long-tail visual relationship understanding.

In our approach, dubbed as RelTransformer, we reconstruct the scene graph into the relation-triplet and global-scene context as we demonstrated in the Fig. 1. The relation triplet here refers to the target relation and its referred subject and object, such as $\langle \text{player}, \text{holding}, \text{bat} \rangle$ in the figure. The global context represents all the relation triplets that are gathered for each appearing relation. We directly connect the target relation "holding" with every element from the relation triplet and global context, and pass their information to the target relation via self-attention. Furthermore, since the long-tail relations tend to be amenable to forgetting, we also propose a novel memory attention module to augment the relation representation with external persistent memory vectors, as we will detail later.

We showcase the effectiveness of our model on VG200 [20] and two recently proposed large-scale long-tail VRR benchmarks, GQA-LT [1] and VG8K-LT [1]. GQA-LT and VG8K-LT scale the number of relation types up to 300 and 2,000 compared to only 50 relation types in VG200. These two benchmarks are highly skewed (e.g., the VG8K-LT benchmark ranges from 14 to 618,687 examples per relation type) and offer us a suitable platform for studying long-tail VRR problems. Our approach achieves the state-of-the-art on those three datasets in our experimental results, demonstrating its effectiveness. We also conducted several ablative experiments and showed the usefulness of each component design in RelTransformer.

2. Background

Visual Relationship Recognition. Correct visual relation prediction requires having a comprehensive understanding of the image contents, which guides many successful works in the literature. Early works employ RNN models to construct a global context by aggregating the node and edge features via iterative message passing such as [14, 22, 39, 44, 50]. e.g., VCTree [39] composes a dynamic tree structure to organize the object orders and apply TreeLSTM [36] to aggregate features. There are also several graph convolutional network (GCN) [18] approaches [5, 21, 30, 45], which attempt to learn different importance weights to the neighborhood nodes. Lin *et al.* [24] extend the graph attention to also capture the node-specific contextual information and encode the edge direction information. More recently, there is also emerging a Transformer-based approach [19], which models the pairwise interaction among nodes and edges in two separate Transformer networks. Our model differs from it mainly in two aspects: a) we have a different message-passing flow in which we specifically aggregate relation features from the relation-triplet and global-context information. b) We further design an effective memory attention module for augmenting the long-tail relation representation.

Long-Tail Visual Relationship Recognition. Long-tail problem is very severe in visual relation recognition (VRR) [1, 24, 51]. There are mainly two approaching directions to alleviate this problem. The first one is semantically guided visual recognition. In this case, language models [27] are employed for zero-shot or few-shot recognition [11, 29]. There are also several VRR works [1, 45, 48, 51] using the language priors as a guidance to learn relation features, which can derive a better classification on long-tail classes. The second direction is to apply the strategies that have been designed for unbalanced object detection, including various class imbalance loss functions (e.g. weighted cross entropy, focal loss [23], equalization loss [37]), sampling

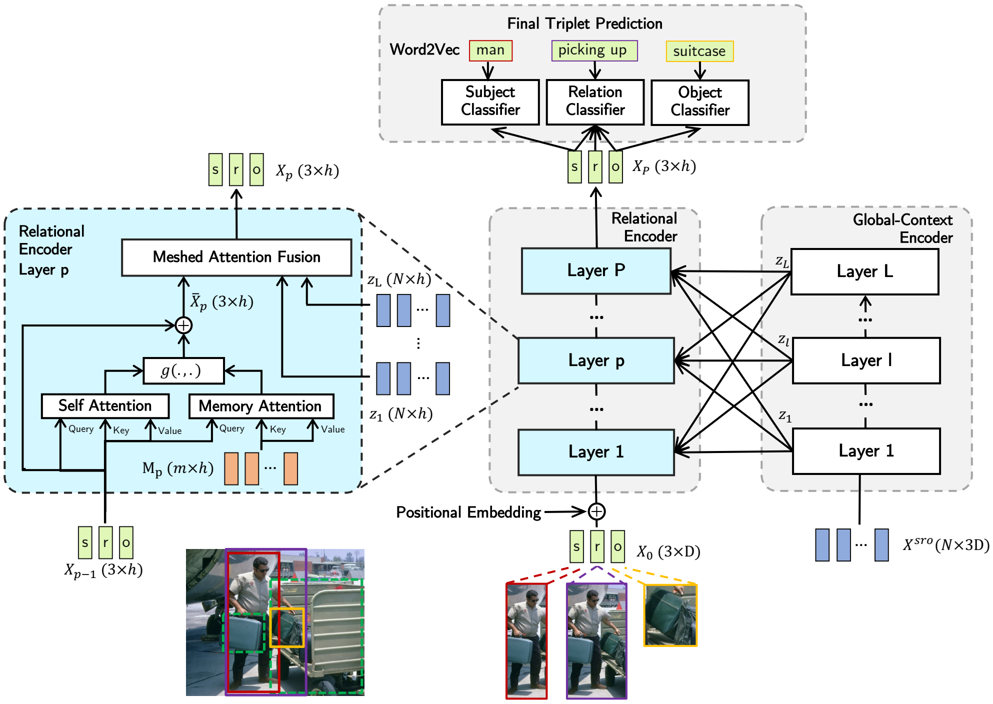


Figure 2. The architecture of our RelTransformer. It comprises a multi-layer global-context and a relational encoder. We display the detailed operations from a relational encoding layer in the left, which include memory attention and meshed attention fusion modules.

strategies (e.g., under-sampling [9], over-sampling [13] and class-balanced sampling [33]), data augmentation [1, 49], meta learning [12, 42], counterfactual learning [38], memory modules [25] and decoupling methods [17]. In our long-tail VRR experiment, we also leverage the language models and many aforementioned long-tail strategies to better classify long-tail relation types. But we focus on understanding their collaborative effect with our message-passing mechanism and how they influence head and tail predictions.

3. Approach

3.1. Problem Definition

An image can be decomposed into a scene graph $G = (N, E)$, where each node ($n_i \in N$) represents an object and each edge ($e_i \in E$) represents the spatial or semantic relationships between two interacting objects. We denote the visual relationship between a subject n_s and an object n_o as r. In the visual relationship recognition (VRR) task, the goal is to predict r between the given n_s and n_o.

$$y^r = f(b^s, b^o, b^r, I) \tag{1}$$

where b^s, b^o, and b^r are the subject, object and relationship bounding boxes. b^r is obtained by the minimum enclosing region of b^s and b^o. y^r is the relationship label. I denotes the arbitrary information such as image raw RGB pixel features. f is the inference model.

3.2. RelTransformer Architecture

RelTransformer mainly comprises two components, an L-layer global-context encoder and a P-layer relational encoder. The overall architecture can be seen in Fig. 2. Given an image, we first extract the object and relation features via a faster R-CNN detection network [32]. Those features are grouped together as a tuple of triplets $(\langle s_1, r_1, o_1\rangle, \ldots, \langle s_N, r_N, o_N\rangle)$ according to their spatial or semantic relationships provided in the dataset, where each s_i, r_i and $o_i \in \mathbb{R}^{1\times D}$ and N is the number of triplets. We first feed all the triplets into a multi-layer global-context encoder to learn a scene-contextualized representation. Then we concentrate on learning the target relation r_i representation from the context of $\langle s_i, r_i, o_i\rangle$ in the relational encoder. An external memory module is also introduced here to augment the long-tail relation representations. Finally, we employ a meshed connection (see Fig. 2) to integrate global-context features into the relation representation learning.

Global-Context Encoder. We employ an L-layer Transformer [41] to model the global-context information and learn their pairwise relationships. We first concatenate each triplet from $(\langle s_1, r_1, o_1\rangle, \ldots, \langle s_N, r_N, o_N\rangle)$ together into a compact representation as $X^{\text{sro}} = (x_1^{\text{sro}} \ldots x_N^{\text{sro}})$ where

$X^{\text{sro}} \in \mathbb{R}^{N \times 3D}$. We then feed X^{sro} into the global-context encoder in a permutation-invariant order.

The Transformer [41] is a stack of multi-headed self-attention (MSA) and MLP layers. Its core component is the self-attention as defined in Eq. 2. In each Transformer encoding layer, the multi-head self-attention repeats the self-attention multiple times and concatenate the results together; the result is then projected back to the same dimensionality. After that, the result is fed into an MLP network and produces each layer's output.

$$f_{\text{sa}}(Q,K,V) = \text{softmax}\left(\frac{(W_q Q)(W_k K)^\top}{\sqrt{d}}\right) W_v V \quad (2)$$

where Q, K and $V \in \mathbb{R}^{t \times h}$ are query, key and value vectors. t is the number of input tokens and h is the hidden size. d is a scaling factor. W_q, W_k and W_v are the learnable weight parameters.

We feed X^{sro} into the global-context encoder, each layer l outputs a contextualized representation z_l. We gather them together as $Z = (z_1, \ldots, z_L)$ where $z_l \in \mathbb{R}^{N \times h}$.

Relational Encoder. It has multiple functions: 1) it specifically aggregates the relation representation from its referred subject and object via self-attention 2) it incorporates a persistent memory to augment the relations with "out-of-context" information, which could especially benefit the long-tail relations. 3) it also has a "bridging" role to aggregate each global-context encoding layer's output into the relation representation through a meshed attention module. The detailed operating procedure is described as follows.

We first add a learnable positional embedding [8] to each position of s_i, r_i, and o_i in order to distinguish their semantic differences in the input sequence, and we denote this sequence as X_{p-1} in the layer p. We apply the self-attention defined in Eq. 2 on X_{p-1} in each layer p and model their pairwise relationship. Their result will be X_p^{att} as shown in Eq. 3.

$$X_p^{\text{att}} = f_{\text{sa}}(X_{p-1}, X_{p-1}, X_{p-1}) \quad (3)$$

Memory Augmentation. The trained model can easily forget long-tail relations because the model training is dominated by the instance-rich (or head) relations, and hence it tends to underperform on lower-frequent relations. Also, the self-attention is limited to attending to the features only from the tokens in an input sequence; hence, each relation only learns a representation based on the immediate context. To alleviate this issue, we propose a novel memory attention module motivated by several successful persistent memory ideas [35, 53] in the literature. We denote a group of persistent and differentiable memory vectors as M. Each time the relation passes its features to M and retrieves the information from M via attention. The memory here captures the information not dependent on the immediate context; instead, it is shared across the whole dataset [35]. Through this way, long-tail relations are able to access the information (e.g., from other relations or itself in different training steps) with relevance to itself, and they can augment the target relation with the useful "out-of-context" information in a well-trained model.

To compute this memory, we first randomly initialize m memory vectors in each relational encoding layer p as $M_p \in \mathbb{R}^{m \times h}$. We then compute the memory attention between the input feature with M_p, we treat X_{p-1} as the query, and M here is the Key and Value. Same self-attention operation is applied here in Eq. 4 and we can obtain X_p^{mem}. The memory is directly updated via SGD.

$$X_p^{\text{mem}} = f_{\text{sa}}(X_{p-1}, M_{p-1}, M_{p-1}) \quad (4)$$

To aggregate X_p^{mem} into X_p^{att}, we design a fusion function as $g(x,y)$ in Eq. 5. $g(x,y)$ is a attention gate and determines how to effectively combine two input features. It computes the complementary attention weights to each input and weighted combine them as the output. Through this fusion function in combination with a skip connection, we can get the fused feature as $\bar{X}_p = g(X_p^{\text{att}}, X_p^{\text{mem}}) + X_{p-1}$.

$$\begin{gathered} g(x,y) = \alpha \odot x + (J - \alpha) \odot y \\ \alpha = \sigma(W[x;y] + b) \end{gathered} \quad (5)$$

where W is a 2D × D matrix. b is a bias term. $[;]$ denotes the concatenation. $\odot$ is the Hadamard product. J is an all-one matrix with the same dimensions as α.

Meshed Attention Fusion. The features from different global-context encoding layers capture different vision granularity, and leveraging features from all of them has shown to be better than the one only from the last encoding layer [4,7]. Therefore, we adopt a meshed connection in our model and contribute each layer's output z_l to the relation representation. To compute the meshed attention, we first compute the cross-attention between $\bar{X}_p$ and each global-context encoding output in $(z_1, \ldots, z_L)$; Its attention output is fused with $\bar{X}_p$ through Eq. 5. Their results are averaged up for each layer. We then project the averaged output in an MLP network and incorporate a skip connection to compute the final fused relation representation X_p in Eq. 6 as the layer output.

$$\begin{aligned} z_p^l &= f_{\text{sa}}(\bar{X}_p, z_l, z_l) \\ X_p &= \text{MLP}(\frac{1}{L}\sum_{l=1}^{L} g(\bar{X}_p, z_p^l)) + \bar{X}_p \end{aligned} \quad (6)$$

Final Triplet Prediction. In the last layer P of the relational encoder, we extract subject x_s, relation x_r and object x_o from X_P accordingly. In the prediction stage, we leverage the language prior knowledge following previous

Architecture	Learning Methods	VG8K-LT				GQA-LT			
		many	medium	few	all	many	medium	few	all
		100	300	1,600	2,000	16	46	248	310
LSVRU	VilHub [1]	27.5	17.4	14.6	15.7	63.6	17.6	7.2	11.7
LSVRU	VilHub + RelMix [1]	24.5	16.5	14.4	15.4	63.4	14.9	8.0	11.9
LSVRU	OLTR [25]	22.5	15.6	12.6	13.6	63.5	15.0	8.2	12.1
LSVRU	EQL [37]	22.6	15.6	12.6	13.6	62.3	15.8	6.6	10.8
LSVRU	Counterfactual $^\sharp$ [38]	12.1	25.6	14.9	17.1	38.6	38.0	9.4	15.2
LSVRU	CE	22.2	15.5	12.6	13.5	62.6	15.5	6.8	11.0
RelTransformer (ours)	CE	**26.8**	**18.6**	**15.0**	**16.1**	**63.4**	**16.6**	**7.0**	**11.2**
LSVRU	Focal Loss [23]	24.5	16.2	13.7	14.7	60.4	15.7	7.7	11.6
RelTransformer (ours)	Focal Loss	**30.5**	**22.8**	**14.8**	**16.8**	**61.9**	**16.8**	**8.3**	**12.2**
LSVRU	DCPL [17]	34.3	15.4	12.9	14.4	**61.4**	23.6	7.6	12.7
RelTransformer (ours)	DCPL	**37.3**	**27.6**	**16.5**	**19.2**	58.4	**38.6**	**13.2**	**19.3**
LSVRU	WCE	35.5	24.7	15.2	17.2	53.4	35.1	15.7	20.5
RelTransformer (ours)	WCE	**36.6**	**27.4**	**16.3**	**19.0**	**63.6**	**59.1**	**43.1**	**46.5**

Table 1. Average per-class accuracy in relation prediction on VG8K-LT and GQA-LT datasets. We evaluate the average per-class accuracy for many, medium, few, and all classes. The best performance for each column is underlined. $\sharp$ denotes our reproduction. Learning methods include various class-imbalance loss functions, augmentation and counterfactual approaches. Our model is marked in gray .

work [51], and represent each ground-truth label in their Word2Vec [27] embeddings, we project them into hidden representations with a 2-layer MLP in the classifier. Finally, we maximize their cosine similarity with x_s, x_r, and x_o individually during the training.

4. Experiments

4.1. Datasets

We evaluate our model on the VG200 dataset and two large-scale long-tail VRR datasets, named GQA-LT [1] and VG8K-LT [1].

GQA-LT. This dataset contains $72,580$ training, $2,573$ validation and $7,722$ testing images. Overall, it contains $1,703$ objects and 310 relationships. The GQA-LT has a heavy "long-tail" distribution where the example numbers per each class range from merely 1 to $1,692,068$.

VG8K-LT. It is collected from Visual Genome (v1.4) [20] dataset, containing $97,623$ training, $1,999$ validation and $4,860$ testing images. It covers $5,330$ objects and $2,000$ different relation types in total, in which least frequent objects/relationships only have 14 examples and the most frequent ones have $618,687$ examples.

VG200. This dataset has been widely studied in the literature [44, 51, 52]. It contains 50 relationships, and the category frequency in this dataset is considerably more balanced than that in GQA-LT and VG8K-LT. We follow the same data split as in [51] in our experiment.

4.2. Experimental Settings

GQA-LT & VG8K-LT Baselines. We compare RelTransformer with several state-of-the-art models. The most popular models in this benchmark are implemented based on LSVRU [51] framework. To improve over the long-tail performance, baseline models usually are combined with the following strategies: 1) class-imbalance loss functions such as weighted cross entropy (WCE), equalization loss (EQL) [37], focal loss [23], and ViLHub loss [1]. 2) relation augmentation strategies such as RelMix [1] to augment more examples in long-tail relations. 3) Decoupling [17] which decouples the learning procedure into representation learning and classification. 4) Counterfactual [38], which alleviates the biased scene graph generation via the counterfactual learning. 5) OLTR [25], which has memory module with augmented attention.

VG200 Baselines. We compare with several strong baselines including Visual Relationship Detection [26], Message Passing [44], Associative Embedding [28], MotifNet [50], Permutation Invariant Predication [14], LSVRU [51], relationship detection with graph contrastive loss (RelDN) [52], GPS-Net [24], Visual Relationship Detection with Visual-Linguistic Knowledge (RVL-BERT) [6] and Relational Transformer Network (RTN) [19].

Evaluation Metrics. For the GQA-LT and VG8K-LT datasets, we report the average per-class accuracy, which is commonly used for long-tail evaluation [1, 17, 37]. Following the same evaluation setting as [1], we split the relationship classes into the many, medium, and few based on the relation frequency in the training dataset as shown in Table 1. For the VG200 dataset, following the previous evaluation setting in [34, 51], we measure the Recall@k and mean Recall@K for predicate classification (PRDCLS), which is to predict the relation labels given the ground truth boxes and labels of the subject and object.

Models	PRDCLS		
	R@20	R@50	R@100
VRD [26]	-	27.9	35.0
Message Passing [44]	52.7	59.3	61.3
Associative Embedding [28]	47.9	54.1	55.4
MotifNet (Left to Right) [50]	58.5	65.2	67.1
Permutation Invariant [14]	-	65.1	66.9
LSVRU [51]	66.8	68.4	68.4
RelDN [52]	66.9	68.4	68.4
Graph-RCNN [45]	-	54.2	59.1
VCTREE-SL	59.8	66.2	67.9
GPS-Net [24]	60.7	66.9	68.8
RVL-BERT [6]	-	62.9	66.6
RTN [19]	68.3	68.7	68.7
RelTransformer (ours)	**68.5**	**69.7**	**69.7**

Table 2. Relation prediction on VG200 dataset.

4.3. Quantitative Results

GQA-LT and VG8K-LT Evaluation. We present our results for GQA-LT and VG8K-LT datasets in Table 1. There is a clear performance improvement over all the baselines with the addition of RelTransformer, especially on the med and few classes. The combination of RelTransformer with WCE improves the med and few category in GQA-LT by a considerable margin of ≈20% compared to all the baselines. This huge gain can be attributed to the weighted assignment of different classes when WCE loss is applied. This further refines the attention weights assigned to different classes in the global context and hence helps the overall performance. While previous works [38] improved the tail performance at the cost of head class accuracy, RelTransformer consistently improves on the tail as well as the head as seen from the table, underlying the effectiveness of our model.

For VG8K-LT, we also see performance gains with the addition of RelTransformer on all baselines across "many", "medium" and "few" categories. A considerable improvement of ≈5% can be seen when RelTransformer is combined with DCPL [17], performing the best for the VG8K-LT dataset. While we see consistent improvements with the addition of RelTransformer for the VG8K-LT dataset, the improvement margins are certainly lower as compared to GQA-LT as seen in Table 1. This is due to the more challenging nature of VG8K-LT, containing 2000 relation classes compared to 300 classes present in GQA-LT. Some qualitative examples with the addition of RelTransformer can be seen in Fig. 5.

VG200 Evaluation. We also evaluate our model on VG200 dataset in Table 2. We compare with many different message-passing approaches in our baselines including RNN-based [39, 50], GCN-based [24, 45] and Transformer-based [6, 19]. These two Transformer-based approaches either only focus on the relational triplet context [6] or neglect it completely [19]. Our model differs from them with a different message-passing strategy, different context construction and a novel memory attention. The experimental results show that our method can better exploit the relational features, and we improve over the best-performing baselines by 0.2% on R@20, 1.0% on R@50, and 0.9% on R@100.

Models	Method	PRDCLS		
		mR@20	mR@50	mR@100
IMP	CE	8.85	10.97	11.77
IMP	EBM [34]	9.43	11.83	12.77
Motif	CE	12.45	15.71	16.8
Motif	EBM	14.2	18.2	19.7
VCTREE	CE	13.07	16.53	17.77
VCTREE	EBM	14.17	18.02	19.53
Ours	CE	**18.51**	**19.58**	**20.19**

Table 3. Mean Recall@K Performance on VG200 Dataset.

Mean Recall@K on VG200. We evaluate the mean recall@k performance on the VG200 dataset for elation predication, and compared RelTransformer with several strong baselines such as VCTREE and Motif with both cross-entropy and EBM losses [34]. The results are summarized in Table 3. We observe that RelTransformer can outperform all the baselines on mR@(20, 50, 100) while only being combined with cross-entropy loss, which shows its robustness on other data-imbalanced datasets.

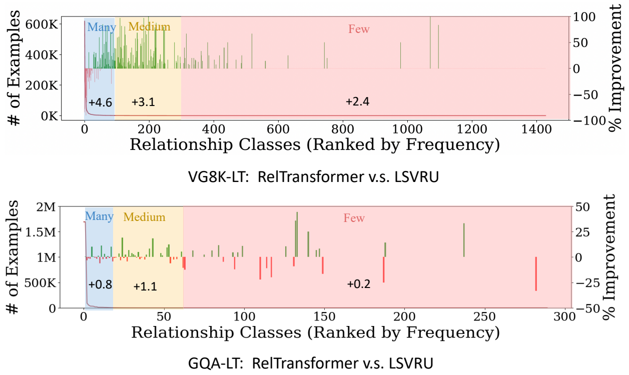
4.4. Further Analysis

To analyze our results in more depth, we quantify our model's improvement per each class and visualize them in Fig 4. We provide the contrast between RelTransformer and LSVRU with cross entropy loss. From the figure, we can observe that RelTransformer can improve the the majority of classes on both datasets. In particular, RelTransformer improves 173 relations while only worsening 32 ones on VG8K-LT dataset with most performance gain from the medium and few classes.

GQA-LT and VG8K-LT Compositional Prediction. The compositional prediction is the correct prediction of the subject, relation, and object together. This could trigger a more skewed long-tail distribution due to its combinatorial nature. To evaluate our model's compositional behaviors, we follow the work [1] to group the classification results by the pairs of (subject, object), (subject, relation), and (object, relation). The results are provided in Table 4, and we can see noticeable performance improvement on all the classes in contrast to the baselines. But we also observe that both RelTransformer(CE) and LSVRU (CE) perform better than the ones combining with focal loss and WCE in many categories, which differs from the results for only predicting the relations. The main reason is that those class imbalance losses hurt more "head" performance on subject/object

Architecture	Learning Methods	many SO	many SR	many OR	medium SO	medium SR	medium OR	few SO	few SR	few OR
LSVRU	VilHub	40.5	32.8	33.7	25.7	14.2	13.9	10.2	5.3	5.2
LSVRU	CE	38.6	30.3	31.5	21.8	11.3	10.8	7.5	4.3	4.2
RelTransformer	CE	**54.2**	**46.6**	**47.2**	**37.4**	**20.8**	**21.8**	**16.1**	**8.6**	**7.7**
LSVRU	Focal Loss	39.2	31.1	32.3	23.2	11.9	11.5	8.2	4.3	4.2
RelTransformer	Focal Loss	**49.9**	**41.7**	**42.5**	**32.2**	**17.7**	**8.0**	**13.1**	**7.0**	**6.4**
LSVRU	WCE	18.3	17.3	17.2	13.7	9.4	9.4	7.1	4.2	3.6
RelTransformer	WCE	**19.2**	**20.0**	**19.5**	**15.7**	**13.6**	**13.5**	**10.3**	**8.7**	**8.1**

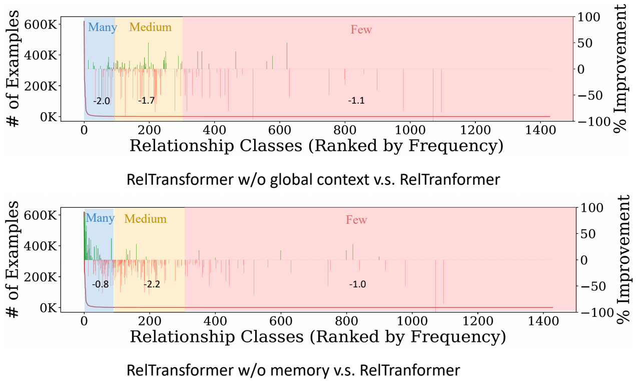
Table 4. Relationship triplet performance on GQA-LT dataset. SO = (subject, object), SR = (subject, relation) and OR = (object, relation).



VG8K-LT: RelTransformer v.s. LSVRU

GQA-LT: RelTransformer v.s. LSVRU

Figure 3. Per-class relationship accuracy comparisons between our RelTransformer and LSVRU [51] baseline on VG8K-LT(top) and GQA-LT(bottom) dataset. The green bars indicate the improvement of RelTransformer over LSVRU, red bars indicate worsening and no bars mean no change. The left-side y-axis represents the number of examples per class. The right-side y-axis shows the absolute accuracy improvement. The x-axis represents the relation classes which are sorted by their frequency.



RelTransformer w/o global context v.s. RelTranformer

RelTransformer w/o memory v.s. RelTranformer

Figure 4. Per-class accuracy comparisons between the RelTransformer and its version without global-context encoder and memory attention using CE loss from Table 5.

compared to CE (see supplementary) and they are reflected in the compositional prediction.

Models	Losses	many 100	med 300	few 1,600	all 2,000
Full model	CE	**26.8**	**18.6**	**15.0**	**16.1**
✗global	CE	24.8	16.9	13.9	14.8
✗mem	CE	26.0	16.4	14.0	15.0
Full model	Focal Loss	**30.5**	**22.8**	**14.8**	**16.8**
✗global	Focal Loss	28.9	19.8	13.3	15.1
✗mem	Focal Loss	30.1	20.8	14.2	15.9
Full model	DCPL	**37.3**	**27.6**	**16.5**	**19.2**
✗global	DCPL	36.1	25.4	15.7	18.2
✗mem	DCPL	37.0	26.8	16.1	18.8
Full model	WCE	**36.6**	**27.4**	**16.3**	**19.0**
✗global	WCE	35.6	24.5	15.2	17.6
✗mem	WCE	36.5	27.2	16.0	18.7

Table 5. Ablation study of RelTransformer on VG8K-LT dataset. global and mem represent the global-context encoder and memory attention module, respectively. ✗represents the removal operation. Our default setting is marked in gray .

4.5. Ablation Studies

To quantify the contributions of each component to the whole model performance, we ablate and evaluate our RelTrasnformer in different versions on VG8K-LT dataset as shown in Table 5. We choose VG8K-LT instead of GQA-lT since it is more challenging and covers more classes.

The role of global-context encoder. To investigate the effect of the global-context encoder, we ablate our RelTransformer with the version without learning global context. The results in Table 5 indicate that the performance will drop on all the categories if we exclude it. It brings the performance down by 1.45% accuracy (acc) on many, 2.35% acc on medium, and 1.13% acc on few when we average all the combined loss functions' results. This analytic shows that incorporating global context can benefit all categories with gaining the most performance on medium classes. We also demonstrate its per-class analysis in Fig. 4, from which we can observe that the performance drops in most classes.

The role of memory attention. The persistent memory vectors are aimed to augment the relation representation

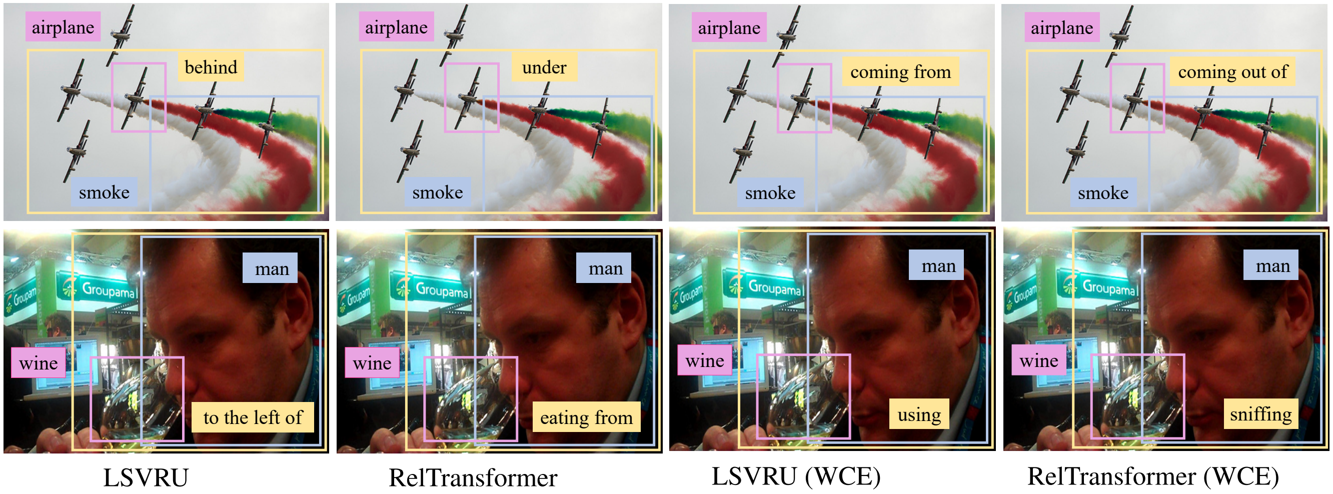


Figure 5. Long-tail relationship recognition visualization. We compare our model with LSVRU in the combination of CE and WCE loss functions. Yellow color denotes the relation, blue denotes the subject, and pink denotes the object.

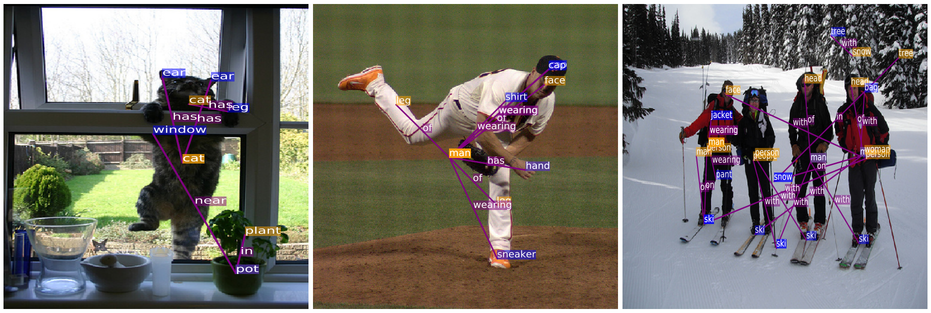


Figure 6. The visualization of relation recognition with our model on VG200 dataset

with more useful "out-of-context" information, especially for the low-frequent relations. In Table 5, we evaluate the version without the memory attention, and we can observe that it brings a performance reduction for all the categories. It drops the performance by 0.4% acc on many, 1.2% acc on medium, and 0.58% acc on few relations when we average the results from all the loss functions. The medium and few are more influenced than many relations, indicating that memory attention can benefit the infrequent relation classes more. We also show the per-class analysis with removing the memory in Fig. 4, where we can find the performances for medium and few relations drop most.

4.6. Qualitative Results

We randomly sample 2 images which include long-tail relationships from the testing dataset. We evaluate RelTransformer and LSVRU [51] in terms of CE and WCE loss functions and contrast them in Fig. 5. It can be observed that LSVRU predicts very trivial relationships such as "behind", "under," or "to the left of". Those trivial relationships convey a vague and inaccurate understanding of the image content. Our RelTransformer instead can describe the relations more accurately. e.g., it predicts "man sniffing wine" for the bottom image. "sniffing" is a long-tail relation type and this prediction exactly matches the ground truth. We also randomly sample 3 images from VG200 dataset and visualize our prediction results in Fig. 6.

5. Conclusion

We presented a Transformer-based long-tail visual relationship recognition model, dubbed RelTransformer, which directly connects the relations with all the visual objects via the attention mechanism. Empirically, we organize the whole scene into the relation-triplet and global-scene context, and attentively aggregate their information to the relation representation under our message-passing flow. We also propose a memory attention module to augment the relation representation with "out-of-context" information, which is shown to be more effective for infrequent relations. With our design, RelTransformer surpasses all previous stat-of-the-art results on GQA-LT, VG8K-lT, and VG200 datasets.

References

[1] Sherif Abdelkarim, Aniket Agarwal, Panos Achlioptas, Jun Chen, Jiaji Huang, Boyang Li, Kenneth Church, and Mohamed Elhoseiny. Exploring long tail visual relationship recognition with large vocabulary. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15921–15930, 2021. 2, 3, 5, 6

[2] Uri Alon and Eran Yahav. On the bottleneck of graph neural networks and its practical implications. In *International Conference on Learning Representations*, 2020. 2

[3] Antoni Buades, Bartomeu Coll, and J-M Morel. A non-local algorithm for image denoising. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 60–65. IEEE, 2005. 2

[4] Jun Chen, Han Guo, Kai Yi, Boyang Li, and Mohamed Elhoseiny. Visualgpt: Data-efficient image captioning by balancing visual input and linguistic knowledge from pretraining. *arXiv preprint arXiv:2102.10407*, 2021. 4

[5] Tianshui Chen, Weihao Yu, Riquan Chen, and Liang Lin. Knowledge-embedded routing network for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6163–6171, 2019. 2

[6] Meng-Jiun Chiou, Roger Zimmermann, and Jiashi Feng. Visual relationship detection with visual-linguistic knowledge from multimodal representations. *IEEE Access*, 9:50441–50451, 2021. 5, 6

[7] Marcella Cornia, Matteo Stefanini, Lorenzo Baraldi, and Rita Cucchiara. Meshed-memory transformer for image captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10578–10587, 2020. 4

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. 4

[9] Chris Drummond, Robert C Holte, et al. C4. 5, class imbalance, and cost sensitivity: why under-sampling beats over-sampling. In *Workshop on learning from imbalanced datasets II*, volume 11, pages 1–8. Citeseer, 2003. 3

[10] Mohamed Elhoseiny, Scott Cohen, Walter Chang, Brian Price, and Ahmed Elgammal. Sherlock: Scalable fact learning in images. volume 31, 2017. 1

[11] Andrea Frome, Greg S Corrado, Jonathon Shlens, Samy Bengio, Jeffrey Dean, Marc'Aurelio Ranzato, and Tomas Mikolov. Devise: a deep visual-semantic embedding model. In *Proceedings of the 26th International Conference on Neural Information Processing Systems-Volume 2*, pages 2121–2129, 2013. 2

[12] David Ha, Andrew Dai, and Quoc V Le. Hypernetworks. *arXiv preprint arXiv:1609.09106*, 2016. 3

[13] Hui Han, Wen-Yuan Wang, and Bing-Huan Mao. Borderline-smote: a new over-sampling method in imbalanced data sets learning. In *International conference on intelligent computing*, pages 878–887. Springer, 2005. 3

[14] Roei Herzig, Moshiko Raboh, Gal Chechik, Jonathan Berant, and Amir Globerson. Mapping images to scene graphs with permutation-invariant structured prediction. *Advances in Neural Information Processing Systems*, 31:7211–7221, 2018. 2, 5, 6

[15] 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. 1, 2

[16] Justin Johnson, Agrim Gupta, and Li Fei-Fei. Image generation from scene graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1219–1228, 2018. 1

[17] Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. In *International Conference on Learning Representations*, 2019. 3, 5, 6

[18] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. 2

[19] Rajat Koner, Poulami Sinhamahapatra, and Volker Tresp. Relation transformer network. *arXiv preprint arXiv:2004.06193*, 2020. 2, 5, 6

[20] R Krishna, Y Zhu, O Groth, J Johnson, K Hata, J Kravitz, S Chen, Y Kalantidis, L Li, DA Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, pages 1–42, 2017. 2, 5

[21] Yikang Li, Wanli Ouyang, Bolei Zhou, Jianping Shi, Chao Zhang, and Xiaogang Wang. Factorizable net: an efficient subgraph-based framework for scene graph generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 335–351, 2018. 2

[22] Yikang Li, Wanli Ouyang, Bolei Zhou, Kun Wang, and Xiaogang Wang. Scene graph generation from objects, phrases and region captions. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1261–1270, 2017. 2

[23] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017. 2, 5

[24] Xin Lin, Changxing Ding, Jinquan Zeng, and Dacheng Tao. Gps-net: Graph property sensing network for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3746–3753, 2020. 2, 5, 6

[25] Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X Yu. Large-scale long-tailed recognition in an open world. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2537–2546, 2019. 3, 5

[26] Cewu Lu, Ranjay Krishna, Michael Bernstein, and Li Fei-Fei. Visual relationship detection with language priors. In *European conference on computer vision*, pages 852–869. Springer, 2016. 5, 6

[27] Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, 2013. 2, 5

[28] Alejandro Newell and Jia Deng. Pixels to graphs by associative embedding. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 2168–2177, 2017. 5, 6

[29] Mohammad Norouzi, Tomas Mikolov, Samy Bengio, Yoram Singer, Jonathon Shlens, Andrea Frome, Greg Corrado, and Jeffrey Dean. Zero-shot learning by convex combination of semantic embeddings. In *ICLR*, 2014. 2

[30] Mengshi Qi, Weijian Li, Zhengyuan Yang, Yunhong Wang, and Jiebo Luo. Attentive relational networks for mapping images to scene graphs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3957–3966, 2019. 2

[31] Siyuan Qi, Yixin Zhu, Siyuan Huang, Chenfanfu Jiang, and Song-Chun Zhu. Human-centric indoor scene synthesis using stochastic grammar. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5899–5908, 2018. 2

[32] Shaoqing Ren, Kaiming He, Ross B Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, 2015. 3

[33] Li Shen, Zhouchen Lin, and Qingming Huang. Relay backpropagation for effective learning of deep convolutional neural networks. In *European conference on computer vision*, pages 467–482. Springer, 2016. 3

[34] Mohammed Suhail, Abhay Mittal, Behjat Siddiquie, Chris Broaddus, Jayan Eledath, Gerard Medioni, and Leonid Sigal. Energy-based learning for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13936–13945, June 2021. 5, 6

[35] Sainbayar Sukhbaatar, Edouard Grave, Guillaume Lample, Herve Jegou, and Armand Joulin. Augmenting self-attention with persistent memory. *arXiv preprint arXiv:1907.01470*, 2019. 4

[36] Kai Sheng Tai, Richard Socher, and Christopher D Manning. Improved semantic representations from tree-structured long short-term memory networks. 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 1556–1566, 2015. 2

[37] Jingru Tan, Changbao Wang, Buyu Li, Quanquan Li, Wanli Ouyang, Changqing Yin, and Junjie Yan. Equalization loss for long-tailed object recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11662–11671, 2020. 2, 5

[38] Kaihua Tang, Yulei Niu, Jianqiang Huang, Jiaxin Shi, and Hanwang Zhang. Unbiased scene graph generation from biased training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3716–3725, 2020. 3, 5, 6

[39] Kaihua Tang, Hanwang Zhang, Baoyuan Wu, Wenhan Luo, and Wei Liu. Learning to compose dynamic tree structures for visual contexts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6619–6628, 2019. 2, 6

[40] Damien Teney, Lingqiao Liu, and Anton van Den Hengel. Graph-structured representations for visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2017. 1

[41] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. 2, 3, 4

[42] Yu-Xiong Wang, Deva Ramanan, and Martial Hebert. Learning to model the tail. In *Advances in Neural Information Processing Systems*, pages 7029–7039, 2017. 3

[43] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and S Yu Philip. A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems*, 32(1):4–24, 2020. 2

[44] Danfei Xu, Yuke Zhu, Christopher B Choy, and Li Fei-Fei. Scene graph generation by iterative message passing. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5410–5419, 2017. 2, 5, 6

[45] 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, 6

[46] Xu Yang, Kaihua Tang, Hanwang Zhang, and Jianfei Cai. Auto-encoding scene graphs for image captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10685–10694, 2019. 1

[47] Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. Exploring visual relationship for image captioning. In *Proceedings of the European conference on computer vision (ECCV)*, pages 684–699, 2018. 1

[48] Ruichi Yu, Ang Li, Vlad I Morariu, and Larry S Davis. Visual relationship detection with internal and external linguistic knowledge distillation. In *Proceedings of the IEEE international conference on computer vision*, pages 1974–1982, 2017. 2

[49] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6023–6032, 2019. 3

[50] 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. 2, 5, 6

[51] Ji Zhang, Yannis Kalantidis, Marcus Rohrbach, Manohar Paluri, Ahmed Elgammal, and Mohamed Elhoseiny. Large-scale visual relationship understanding. In *Proceedings of*

the AAAI Conference on Artificial Intelligence, volume 33, pages 9185–9194, 2019. 2, 5, 6, 7, 8

[52] Ji Zhang, Kevin J Shih, Ahmed Elgammal, Andrew Tao, and Bryan Catanzaro. Graphical contrastive losses for scene graph parsing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11535–11543, 2019. 5, 6

[53] Yingzhu Zhao, Chongjia Ni, Cheung-Chi Leung, Shafiq R Joty, Eng Siong Chng, and Bin Ma. Speech transformer with speaker aware persistent memory. In *INTERSPEECH*, pages 1261–1265, 2020. 4