RelTransformer: Balancing the Visual Relationship Detection from Local Context, Scene and Memory

Jun Chen1 Aniket Agarwal1,2 Sherif Abdelkarim1 Deyao Zhu1 Mohamed Elhoseiny1
1King Abdullah University of Science and Technology
2Indian Institute of Technology
{jun.chen,deyao.zhu,mohamed.elhoseiny}@kaust.edu.sa
aagarwal@ma.iitr.ac.in,sherif.abdelkarim91@gmail.com

Abstract

Visual relationship recognition (VRR) is a fundamental scene understanding task. The structure that VRR provides is essential to improve the AI interpretability in downstream tasks such as image captioning and visual question answering. Several recent studies showed that the long-tail problem in VRR is even more critical than that in object recognition due to the compositional complexity and structure. To overcome this limitation, we propose a novel transformer-based framework, dubbed as RelTransformer, which performs relationship prediction using rich semantic features from multiple image levels. We assume that more abundant contextual features can generate more accurate and discriminative relationships, which can be useful when sufficient training data are lacking. The key feature of our model is its ability to aggregate three different-level features (local context, scene, and dataset-level) to compositionally predict the visual relationship. We evaluate our model on the visual genome and two “long-tail” VRR datasets, GQA-LT and VG8k-LT. Extensive experiments demonstrate that our RelTransformer could improve over the state-of-the-art baselines on all the datasets. In addition, our model significantly improves the accuracy of GQA-LT by 27.4% upon the best baselines on tail-relationship prediction. Our code is available in https://github.com/Vision-CAIR/RelTransformer.

Figure 1. Overview of our problem statement: (a) an image with grounded bounding boxes as input. (b) The top 20 relationship in Visual Genome dataset with their relative frequencies. (c) We leverage the relationship representation learning with multi-level features including local, scene and dataset-level context.

1. Introduction

Scene graph generation (SGG) is a task that goes beyond recognizing individual objects by comprehensively understanding relationships between interacting objects in a visual scene. Owning to the enriched scene understanding provided by SGG, it can benefit various vision tasks such as image captioning [51, 50], visual question answering [44, 19], image generation [20], and 3D scene synthesis [35]. However, due to the significant “long-tail” problems existing in many visual relationship recognition (VRR) datasets [22, 19], the predictions of most existing methods are biased toward head classes, leading to a less rich understanding of visual scenes.

Scene graph generation is usually performed in two steps [25, 54, 55, 28]: First, a pretrained object detector such as Faster RCNN [36] is employed to identify visual objects. Then it infers the relationship between each pair of objects. The relationship prediction step is very challenging because it requires high-level logical and precise reasoning from the image content. The model should also be capable of distinguishing the object’s behavior going beyond describing trivial positional relations. For instance, In Fig 1 (a) a good VRR model is required to identify the relation “passengers boarding bus” rather than “passengers to the right of bus”. Furthermore, the highly-skewed long-tail distribution of re-
relationship in human data collection [1, 22, 19] (Fig 1 (b)) renders the identification of infrequent relations even more challenging.

Many previous works have studied this VRR task and proposed various architectures with different organization of the whole scene structure, in which the relation representation is learnt in different context. The main architectures include the GCN network [49], bidirectional LSTMs [53] and dynamic tree structures [43] where they focus on either local or global features. In contrast to them, we propose a novel transformer-based framework, RelTransformer, which could effectively incorporate multi-level features for relationship learning. We explicitly incorporate three different types of features (see Fig 1): (1) **Local-context features**: It refers to the objects that closely connect to the relationship features with respect to the spatial and semantic relevance. (2) **Scene-level features**: The global scene context augments the relationship learning with more surrounding background features, and it can reduce the bias from the local context. (3) **Dataset-level features**: A learnable memory module is designed to encode the prior knowledge across the whole dataset. This memory is independent from the data input and is optimized by our proposed memory-based self-attention module.

Long-tail VRR task is less covered and developed in the literature. Recently, several works [1, 42, 46] realize the importance of “long-tail relation” problems, and they proposed some different benchmarks and approaches to well study this problem, including augmentation [54], counterfactual causality [42] and head-to-tail knowledge transfer [14] etc. Unlike those approaches, our model can also alleviate this “long-tail” problem by leveraging multi-level context features. One of the important reasons for failing to predict long-tail relationships is due to lack of sufficient discriminative features [46], and our model makes the long-tail relation representation more discriminative through aggregating more diverse context features.

The contributions of this study are summarized below.

- We propose a novel transformer-based VRR model that can effectively leverage multi-level features to enhance the relationship prediction by generating more discriminative features.
- We propose a novel memory-based self-attention module that can augment the self-attention with an independent learnable memory. This memory module is flexible to be applied in any transformer-based models and effectively encode the prior knowledge of the training dataset.
- We systematically evaluate our model on the VG200 dataset [22] and two large-scale long-tail datasets named GQA-LT [1] and VG8K-LT [1]. Our experimental results demonstrate the effectiveness of our method, which achieves state-of-the-art performance on three datasets.

2. Related Work

**Visual Relationship Recognition (VRR).** Numerous VRR methods have been studied in the literature [24, 54, 48, 30] and inspired many subsequent studies [53, 54, 55, 26, 34, 49], all of which relied on a pretrained object detector to distinguish the relationships. Li et al. [26] proposed to augment the relationship prediction from the region caption. Additionally, various approaches that focus on different architectures of information propagation have also been examined in [49, 53, 43]. Moreover, some studies [43, 5] have considered the graph attributes such as the edge direction and node priority in their modeling.

**Long-Tail Classification** Extensive works have investigated the long-tail relation problems in various domains during the past years [29, 2, 33, 41, 27, 21]. Several approaches, e.g. weighted cross entropy and focal loss [27] are designed to reweigh more weights to the tail classes. Furthermore, many approaches involving meta learning [12, 47], regularization [52], sampling (e.g., undersampling [11], over-sampling [13] and class-balanced sampling [38]), counterfactual learning [42], memory modules [29] and decoupling methods [21] have been proposed. These approaches have achieved huge success in mitigating long-tail problems.

**Transformers** Transformer [45] has been successfully applied in the field of natural language processing [9, 3] during the past few years. It is now gradually extended into many other domains including vision, biology and chemistry [6, 10, 4, 23, 37, 18]. Transformer has the advantages of fast training speed, long-range dependency [8, 39] and strong generation ability with a large amount of training data [9, 3] compared to conventional language models such as LSTM [17]. With those advantages, we decide to adapt the transformer into VRR domain.

In our work, Unlike previous VRR methods, we design a Transformer-based model and rely on various attention modules to explicitly aggregate the local, global and dataset-level features into the relation learning. Also different from previous long-tail classification methods, our model is to generate more discriminative long-tail relation representation by leveraging from richer context features.

3. Approach

An image $I$ can be described in a scene graph $G = (N, E)$, where each node ($n_i \in N$) represents an object in the image and each edge ($e_i \in E$) represents the spatial
or semantic relationships between two objects. We denote a visual relationship between a subject $n_s$ and an object $n_o$ as the $r_{s,o}$. In the Visual Relationship Recognition (VRR) task, the goal is to predict the relationship labels given the subject and object bounding boxes.

$$y^s, y^o, y^r = f(b^s, b^o, I)$$

where $b^s$ and $b^o$ are the bounding boxes of the subject and object, $I$ denotes the arbitrary information of the image including the ground-truth labels and spatial information, $f$ represents an inference model.

### 3.1. RelTransformer Architecture

Our RelTransformer architecture (shown in Fig. 2) consists of a local-context encoder, in which we construct the relationship features and its closely connected subject and object as a local context, and a scene-level encoder for constructing the global context which contains all the objects and relationships from the same image. We also design a novel memory module to learn the prior knowledge across the entire dataset.

Given an image, we first extract the object and relationship features via a faster R-CNN detection network [36], where a relationship bounding box is the minimum enclosing region of its referred subject and object. As shown in Fig. 2, we formalize each group of subject, relationship and object as a triplet $⟨s, r, o⟩$ and feed all the triplets from the same image into the scene encoder. Simultaneously, we individually feed each triplet into the local-context encoder. These two different-level features have the further interaction via a cross-scene attention module. Moreover, we design a memory-based self-attention module to augment the context-dependent relationship features with the encoded prior knowledge.

#### 3.1.1 Scene-level Encoder

Scene-level features can augment the relationship representation with additional contextual information, thus reducing the prediction bias caused by local context. We denote the relationship region features from a given image as $X$, containing the triplets $⟨X_1^s, X_1^o, X_1^r⟩, \ldots, ⟨X_N^s, X_N^o, X_N^r⟩$. We feed $X$ into the scene encoder. To have a consistent representation of each triplet, we organize the triplet into a concatenation representation as $V_{sro}^{sro}$ and feed all the triplets $V_{sro}^{sro} = (V_{sro}^{sro}, \ldots, V_{N_sro}^{sro})$ into the scene encoder, where $V_{sro}^{sro} \in \mathbb{R}^{N_sro \times T}$.

We define our scene encoder as a self-attention operation (e.g., [45]), where each triplet could attentively propagate its information into the other elements, and each triplet will thus have the contextualized representations. The self-attention is a permutation-invariant operation, which firstly
transfers the input into a set of query matrices $Q$, key matrices $K$, and value matrices $V$. Then, it calculates the scaled dot-product attention between $Q$ and $K$. The computed attention is finally weighted summed with $V$:

$$f_{sa}(Q, K, V) = \text{Softmax} \left( \frac{QK^\top}{\sqrt{d}} \right) V$$

(2)

where $Q$, $K$ and $V \in \mathcal{R}^{D_1 \times D_2}$, and $d$ is a scaling factor. $Q$, $K$ and $V$ have the same dimensionality. We feed each triplet $X_i^{sro}$ ($i \in N$) into the self-attention

$$X_{att} = f_{sa}(W_qX_i^{sro}, W_k X_i^{sro}, W_v X_i^{sro})$$

(3)

where $W_q$, $W_k$ and $W_v$ are learnable weight matrices.

The output of self-attention $X_{att}$ is then fed into a position-wise feed-forward network FFNN($\cdot$), which is a 2-layer affine transformation network. We define it as the follows:

$$\text{FFNN}(X)_i = W_2 \text{GELU}(W_1 X_i + b_1) + b_2$$

$$S^i = \text{FFNN}(X_{att})$$

(4)

where $W_1$ and $W_2$ are two learnable weight matrices, $b_1$ and $b_2$ are two bias terms. GELU [15] is Gaussian Error Linear Units activation function.

Additionally, $S^i$ denotes the $i$-th scene-encoder layer output, and in total there are $L$ parallel layers. We group them together as $(S = S^1, \ldots, S^L)$, where $S \in \mathcal{R}^{N \times T}$. Finally, we feed $S$ into the local-context encoder.

### 3.1.2 Local-Context Encoder

Relationship predictions are mainly determined by the local mutual spatial and semantic relationships with the referring subjects and objects. Thus, we designed a $Z$-layer local-context encoder to learn the local interaction. We treat each triplet $(X_i^q, X_i^k, X_i^v)$ as a local context and rely on the self-attention [45] to learn their pairwise relation. However, such self-attention has the intrinsic limitation that it merely depends on the input elements and cannot learn using context-free features such as prior knowledge. The prior knowledge also augment the relationship prediction by retrieving the similar context from the “history”. To incorporate the prior knowledge into the self-attention module, we design a novel memory-based self-attention operator.

**Memory-Based Self-Attention.** In the $z$-th encoding layer, we take the output $X_i^{z-1}$ from last layer as the current input, and feed it to the memory-based self-attention module (Fig. 3). This memory-based self-attention consists of 2 components: a regular self-attention and memory-attention, both sharing the same query. The outputs of these 2 modules are then combined, detailed as follows:

We firstly compute the query $Q_v$, key $K_v$ and value $V_v$ for the self-attention from the input $X_i^{z-1}$ and compute the memory key $K_m$ and memory value $V_m$ from a learnable memory $M^z$ in $z$-th layer.

$$Q_v, K_v, V_v = W_q q_i^{z-1}, W_k k_i^{z-1}, W_v v_i^{z-1}$$

$$K_m, V_m = W_m M^z$$

(5)

where $W_q$, $W_k$, $W_v$, $W_m$ and $W_{km}$ are learnable weight parameters. $M^z \in \mathcal{R}^{P \times T}$ is the learnable memory matrices containing $P$ randomly initialized vectors from the layer $z$, and we directly optimize it through SGD.

Then, the same query $Q_v$ is used to retrieve the self-attention and memory-attention blocks in parallel; see Eq 2

$$X_i^{att} = f_{sa}(Q_v, K_v, V_v)$$

$$X_i^{mem} = f_{sa}(Q_v, K_m, V_m)$$

(6)

$X_i^{att}$ and $X_i^{mem}$ are then combined together with a learnt attention matrix $\alpha$ as follows

$$\alpha = \sigma(W_m [X_i^{att}; X_i^{mem}] + b_m)$$

$$H_i^z = \alpha \odot X_i^{att} + (J - \alpha) \odot X_i^{mem}$$

(7)

where $W_m \in \mathcal{R}^{D \times 2D}$. $b_m$ is a bias term. $[; ;]$ denotes the concatenation operation, $\odot$ denotes the Hadamard product. $\alpha$ has the same size as $X_i^{att}$. $J$ is an all-one matrix with the same dimension as $\alpha$.

**Cross-Scene Attention.** Given the output of memory-based self-attention $H_i^z$ and the multi-layer output of scene-level encoder $S = (S^1, \ldots, S^L)$, we perform a cross-scene attention operation and attentively retrieve the scene features.
from each encoding layer. Each layer is finally weighted contributed to the relationship features.

We first compute the cross-attention between \( H^*_i \) with each scene encoder layer \( S^l \) using the same \( f_{sa} \), defined in Eq 2.

\[
\tilde{H}^*_i l = f_{sa}(\hat{W}_q H^*_i, \hat{W}_k S^l, \hat{W}_v S^l)
\]

where \( \hat{W}_q, \hat{W}_k \) and \( \hat{W}_v \) are learnable weights, and \( l \) is the \( l \)th layer output from the scene encoder.

\( H^*_i \) and \( \tilde{H}^*_i l \) are then weighted summed up together with another learned attention matrix \( \beta \), and we will aggregate each encoding layer’s contributions together as follows.

\[
\beta = \sigma(W_c[H^*_i; \tilde{H}^*_i l] + b_c)
\]

\[
\hat{H}^*_i = 1 \sum_{l=1}^{L}(\beta \odot H^*_i + (J - \beta) \odot \tilde{H}^*_i l))
\]

where \( W_c \in \mathbb{R}^{D \times 2D} \), \( b_c \) is a bias term. \( [:] \) denotes the concatenation operation. \( \odot \) denotes the Hadamard product. \( \beta \) has the same size as \( H^*_i \). \( J \) is an all-one matrix with the same dimension as \( \beta \).

We compute the final output with a position-wise feed forward network defined in Eq 4.

\[
X^*_i = FFNN(\hat{H}^*_i)
\]

The residual connection and layer normalization are shown in the Fig 2 and denoted as AddNorm term.

**Relationship Classification.** The final feature representation of triplet \((X^*_1, X^*_i, X^*_o)\) is \( X^*_2 \) from last layer of context-encoder. The subject, relationship and object features from \( X^*_2 \) are denoted as the \( X^Z_s, X^Z_r \) and \( X^Z_o \), respectively. To incorporate the direction information into \( R_{s \rightarrow o} \), we concatenate \( X^Z_s, X^Z_r \) and \( X^Z_o \) together for relationship prediction as follows

\[
P = \text{Softmax}(\hat{W}_2(\tilde{W}_1[X^Z_s; X^Z_r; X^Z_o] + \bar{b}_1) + \bar{b}_2)
\]

where \( \tilde{W}_1 \in \mathbb{R}^{D \times 3D} \) and \( \hat{W}_2 \in \mathbb{R}^{D \times D} \) are learning weights. \( \bar{b}_1 \) and \( \bar{b}_2 \) are bias terms, and \( [:] \) represents the concatenation operation.

### 4. Experiments

#### 4.1. Datasets

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

**GQA-LT.** This dataset contains 72,580 training, 2,573 validation and 7,722 test images. Overall, it contains 1,703 objects and 310 relationships. The class distribution in this dataset follows a heavy “long-tail” where the numbers of appearing objects and relations range from merely 1 to 1,692,068.

**VG8K-LT.** This dataset is a subset of Visual Genome (v1.4) [22] dataset, containing 97,623 training, 1,999 validation and 4,860 testing images. This dataset collects most frequent 5,330 object classes and 2,000 relationships. It also follows a long-tail distribution with least frequent objects/relationships showing 14 examples and the most frequent ones showing 618,687 examples.

**VG200.** This dataset has been popularly studied in many previous works [48, 54, 55]. It contains most frequent 150 objects and 50 relationships, and each 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 [54].

#### 4.2. Experimental settings

**GQA-LT & VG8K-LT Baselines.** We compared with several state-of-the-art long-tail approaches, including

- **Focal Loss** [27], which is a popular method for addressing the imbalanced object detection problem.
- **Weighted Cross Entropy (WCE),** which encourages weighing the tail more than the head. Here we compute the inverse class frequency and treat it as the weight for each class.
- **EQL** [41], which prevents updating the discouraging gradients of negative examples for infrequent classes.
- **Decoupling (DCPL)** [21], which decouples the learning procedure into representation learning and classification.
- **ViLHub** [1], which develops a vision & language hubless loss and encourages fair prediction over the frequent and infrequent classes in the batch-level.
- **Counterfactual** [42], which unbiases the biased scene graph generation via the counterfactual learning.
- **LSVRU** [54], which is a general framework for visual relationship understanding and it has Cross Entropy (CE) as objective loss function. The aforementioned baselines and our model are established based on this framework.

**VG200 Baselines.** We compare with several strong baselines including Visual Relationship Detection [30], Message Passing [48], Associative Embedding [32], MotifNet [53], Permutation Invariant Predication [16], LSVRU [54], relationship detection with graph contrastive loss (ReIDN) [55], and the more recent GPS-Net [28].

**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 [41, 21, 1]. Following the same evaluation setting in [1], we split the object/relationship classes into the many, medium and few respectively, where its splitting details are shown in Table 2.
### Subjects/Objects Relationships

| Datasets | Base Models | Learning Methods | Subjects | Objects | Relationships |
|----------|-------------|------------------|----------|---------|--------------|
| GQA-LT   | LSVRU [54]  | CE               | many     | medium  | few          | all | many | medium | few | all |
|          | RelTransformer | CE              | 68.3     | 37.0    | 6.9          | 14.5 | 62.6 | 15.5   | 6.8 | 11.0 |
|          | LSVRU       | CE               | 78.0     | 56.6    | 14.2         | 23.8 | 63.4 | 16.6   | 7.0 | 11.2 |
|          | LSVRU       | CE               | 66.8     | 44.0    | 10.3         | 18.3 | 63.6 | 17.6   | 7.2 | 11.7 |
|          | LSVRU       | CE               | 68.2     | 39.2    | 7.5          | 15.3 | 60.4 | 15.7   | 7.7 | 11.6 |
|          | LSVRU       | CE               | 68.9     | 43.7    | 10.0         | 18.0 | 63.5 | 15.0   | 8.2 | 12.1 |
|          | LSVRU       | CE               | 64.0     | 35.3    | 6.4          | 13.7 | 61.4 | 23.6   | 7.6 | 12.7 |
|          | LSVRU       | CE               | 68.3     | 37.0    | 6.9          | 14.5 | 38.6 | 38.0   | 9.4 | 15.2 |
|          | LSVRU       | CE               | 68.2     | 39.2    | 6.9          | 14.5 | 38.6 | 38.0   | 9.4 | 15.2 |
|          | LSVRU       | CE               | 57.3     | 11.1    | 8.5          | 11.4 | 22.2 | 15.5   | 12.6 | 13.5 |
|          | RelTransformer | CE              | 67.1     | 25.9    | 11.5         | 16.5 | 26.8 | 18.6   | 15.0 | 16.1 |
|          | LSVRU       | VC                | 61.6     | 20.3    | 10.1         | 14.2 | 27.5 | 17.4   | 14.6 | 15.7 |
|          | LSVRU       | Focal Loss [27]   | 58.1     | 13.9    | 8.9          | 12.1 | 24.5 | 16.2   | 13.7 | 14.7 |
|          | LSVRU       | EQL [41]         | 56.9     | 12.1    | 10.0         | 12.7 | 22.6 | 15.6   | 12.6 | 13.6 |
|          | LSVRU       | DCPL [21]        | 53.8     | 5.9     | 7.9          | 9.9  | 34.3 | 15.4   | 12.9 | 14.4 |
|          | LSVRU       | Counterfactual ‡ | 57.3     | 11.1    | 8.5          | 11.4 | 12.1 | 25.6   | 14.9 | 17.1 |
|          | LSVRU       | WCE               | 52.8     | 27.2    | 10.8         | 14.5 | 35.5 | 24.7   | 15.2 | 17.2 |
|          | RelTransformer | WCE              | 50.1     | 31.3    | 13.7         | 18.0 | 36.6 | 27.4   | 16.3 | 19.0 |

Table 1. The main results on GQA-LT and VG8K-LT datasets. We separately evaluate the average per-class accuracy for many, medium, few and all the classes. The best performance is indicated in **Bold** font. ‡ denotes our reproduction.

Figure 4. Per-class accuracy comparisons of our RelTransformer (left) and RelTransformer (WCE) (right) with LSVRU [54] baseline on GQA-LT dataset. The green bars indicate the improvement, 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.

| Datasets | Subjects & Objects | Relationships |
|----------|--------------------|---------------|
| GQA-LT [1] | 86 255 1362 | 16 46 248 |
| VG8K-LT [1] | 267 799 4264 | 100 300 1600 |

Table 2. Class count for each category in GQA-LT & VG8K-LT dataset.

For VG200 dataset, following previous evaluation setting in [54], we evaluate on: (1) **Scene Graph Classification (SGCLS):** given the ground truth subject and object boxes, the goal is to predict the subject, object and relationship labels. (2) **Predicate Classification (PRDCLS):** given the ground truth boxes and labels of the subject and object, the goal is to predict the relationship labels.

**Image and Word Features** In order to fairly compare with previous works [1, 54], we also apply the Faster R-CNN [36] with VGG16 backbone to extract the object proposal features for GQA-LT, VG8K-LT and VG200 datasets. We apply the pretrained word2vec [31] embeddings as the category word representation. The learnable positional embedding is employed in the local-context encoder.

### 4.3 Quantitative Results

**GQA-LT and VG8K-LT Evaluation.** The results for GQA-LT and VG8k-LT datasets are presented in Table 1. Our RelTransformer (with CE and WCE loss) achieve the best performance among all the compared baselines on both
| Dataset | Models | SGCLS | PRDCLS | SGCLS | PRDCLS |
|---------|--------|-------|--------|-------|--------|
| VG200   | Recall at | 20   | 50    | 100   | 20    | 50    | 100   |
|         | VRD [30] |       |        |       | 11.8  | 14.1  |       |
|         | Message Passing [48] | 31.7 | 34.6  | 35.4  | 27.9  | 35.0  |       |
|         | Associative Embedding [32] | 18.2 | 21.8  | 22.6  | 47.9  | 54.1  | 55.4  |
|         | MotifNet (Left to Right) [53] | 32.9 | 35.8  | 36.5  | 58.5  | 65.2  | 67.1  |
|         | Permutation Invariant [16] |       | 36.5  | 38.8  | 56.1  | 66.9  |       |
|         | LSVRU [54] | 36.0 | 36.7  | 36.7  | 66.8  | 68.4  | 68.4  |
|         | RelDN [55] | 36.1 | 36.8  | 36.8  | 66.9  | 68.4  | 68.4  |
|         | Message Passing [48] | 31.7 | 34.6  | 35.4  | 52.7  | 59.3  | 61.3  |
|         | Associative Embedding [32] | 18.2 | 21.8  | 22.6  | 47.9  | 54.1  | 55.4  |
|         | MotifNet (Left to Right) [53] | 32.9 | 35.8  | 36.5  | 58.5  | 65.2  | 67.1  |
|         | Permutation Invariant [16] |       | 36.5  | 38.8  | 64.8  | 66.9  |       |
|         | LSVRU [54] | 36.0 | 36.7  | 36.7  | 66.8  | 68.4  | 68.4  |
|         | RelDN [55] | 36.1 | 36.8  | 36.8  | 66.9  | 68.4  | 68.4  |

Table 3. The main results on VG200 dataset. The best performance is indicated in **Bold** font.

| Dataset | Base Models | Learning Methods | SO | SR | OR | SO | SR | OR |
|---------|-------------|------------------|----|----|----|----|----|----|
| GQA-LT  | LSVRU[54]   | CE               | 38.6| 30.3| 31.5| 11.3| 10.8| 7.5 |
|         | RelTransformer | CE             | **54.2**| **46.6**| **47.2**| **37.4**| **20.8**| **21.2**| **16.1**| **8.6**| **7.7**|
|         | 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.

| Models | Subjects/Objects | Relationships |
|--------|------------------|---------------|
|        | many med few all | many med few all |
| Base (LSVRU) [54] | 68.3 | 37.0 | 6.9 | 14.5 | 62.6 | 15.5 | 6.8 | 11.0 |
| RelTransformer (w/ 1 layers) | 77.3 | 56.0 | 13.8 | 23.3 | **63.9** | 16.2 | 7.0 | 11.2 |
| RelTransformer (w/ 2 layers) | 78.0 | 56.6 | 14.2 | 23.8 | 63.4 | 16.6 | 7.0 | 11.2 |
| RelTransformer (mem* [40]) | 77.7 | 56.4 | 14.0 | 23.6 | 63.5 | 16.5 | 6.5 | 11.0 |
| RelTransformer | 78.0 | 56.6 | 14.2 | 23.8 | 63.4 | 16.6 | 7.0 | 11.2 |
| RelTransformer (w/o scene) | 74.1 | 50.7 | 11.1 | 20.2 | 61.9 | 15.1 | 7.0 | 11.0 |
| RelTransformer (w/o mem) | 78.1 | 56.3 | 13.7 | 23.4 | 63.8 | 16.3 | 7.2 | 11.4 |
| RelTransformer (w/o scene) | 51.9 | 42.4 | 18.0 | 23.4 | 60.6 | 56.1 | 37.1 | 41.4 |
| RelTransformer (w/o mem) | 50.0 | 46.1 | 28.2 | 31.9 | 60.8 | 57.3 | 38.3 | 42.3 |
| RelTransformer WCE | 50.3 | 46.2 | **28.7** | **32.4** | 63.6 | **59.1** | **43.1** | **46.5** |

Table 5. Ablation Study of RelTransformer on GQA-LT dataset. RelTransformer WCE denotes its combination with Weighted Cross Entropy. mem* represents the persistent memory proposed by Sukhbaatar et al. [40]

Datasets and especially show their strength on tail prediction (medium and few). RelTransformer can generally improve the performance over each category. With the combination of WCE, it significantly improves the tail, yielding the improvement over the best baseline from 14.0% to 28.7% on few subjects & objects and from 15.8% to 43.1% on few relationships with GQA-LT dataset. VG8K-LT is a more challenging dataset since it contains more relationship and object types, whereas it is encouraging to see RelTransformer still improve over all the baselines.

To analyze our results in more depth, we quantify our model’s improvement per each class and visualize them in Fig 4. It shows that RelTransformer and RelTransformer (WCE) improve the majority of classes with mostly gaining from the medium and tail classes. In particular, RelTransformer (WCE) improves 865 objects classes with only worsening 194 ones, and it improves 209 relationships with only worsening 22 ones. We notice that most harmed classes in RelTransformer (WCE) are from the head, which we think is influenced by the WCE loss as we observed a similar phenomenon in LSVRU (WCE) baseline.

**VG200 Evaluation.** Table 3 suggests that our model can also perform well on the VG200 dataset, where the classes are much more balanced compared to GQA-LT and VG8K-LT datasets. RelTransformer outperforms all the baselines on the Predicate Classification (PRDCLS). Our model also improves the previous best on Scene Graph Classification (SGCLS) Recall@20 by 1.3%. Those performance gains imply that our model framework is better than the others based on GCN or LSTM, and incorporating multi-level features could augment the discriminative relationship features not only constrained to the long-tail classes but in general.

**Ablation Studies.** To quantify the contributions of each
LSVRU vs. RelTransformer (WCE)

Figure 5. Relationship Detection Visualization. In each pair of 4 demonstrated images, left part is shows the relationship generated from LSVRU and the right part shows the relationship from our RelTransformer (WCE)

component to the whole model architecture, we ablate our RelTrasnformer in different versions and evaluate them on GQA-LT dataset as shown in Table 5.

Number of Local-Context Encoder Layers. We investigate the effect of the number of local-context encoding layers by experimenting with one and two layers. The results shown in Table 5 suggest that two layers local-context encoder performs slightly better. With this observation, all subsequent experiments follow the two layers local-context encoder setting.

Effectiveness of the Memory-Based Self-Attention. We evaluate our memory-based self-attention in two steps: (1) We contrast our memory with another persistent memory [7] which is also applicable in Transformer. They treat a persistent memory as a extended key matrices and value matrices in self-attention operation. The results indicate that our designed memory is superior to theirs. (2) We evaluate its contributions to the whole model architecture under two configurations: Without WCE, our memory-based self-attention can improve the few subjects/objects by 0.4% but it decreases the accuracy of few relationships by 0.2%; With WCE, it improves performance by 0.5% on tail subject/object and by 4.8% on tail relationships. We hypothesis that such different behaviour (~0.2% and +4.8%) on tail relationship prediction is because our memory is directly optimized by SGD gradients and tail relationship features receive more activation in memory under WCE setting.

Effectiveness of the Scene-Level Encoder. To investigate the role of scene-level encoder, we conduct the experiments with the removal of it from the whole architecture. Without WCE, we can observe that removing scene encoder brings the performance drop by 3.6% on subjects/objects and 0.2% on relationships in overall performance. With WCE, the removal of scene encoder brings the performance drop by 9.0% on subjects/objects and 5.1% on relationships, which underlies its usefulness.

Relationship Triplet Performance. The compositional prediction is a more challenging task since it requires to precisely detect multiple objects and relationships together, and it can cause a more heavier long-tail distribution in prediction due to its combinatorial nature. To evaluate our model’s compositional learning behaviour, we test our model and all the benchmark models on the relationship triplet accuracy [1], which is to classifying $\langle S, R, O \rangle$ altogether, and we group the results by the pairs of (S, O), (S, R), (O, R). The results are shown in Table 4, demonstrating the contrast between our RelTransformer with LSVRU. Our model is consistently better than LSVRU in compositional learning. Due to the space limit, other comparison results are provided in supplementary.

4.4. Qualitative Results

Relationship Detection Visualization We randomly sample 4 different images consisting of long-tail relationships and evaluate them with LSVRU [54] and our RelTransformer (WCE). The contrasting demonstrations are shown in Fig. 5. We can observe that LSVRU only tends to predict very trivial relationships like “to the left of”, “on” and “near” which all belong to the head relationships. Those trivial relationships convey a vague and inaccurate understanding of the image content. On the other hand, our RelTransformer detects the “chewing”, “towing” and “chained to” infrequent relationships, which are more likely to abide by the human understanding.

5. Conclusion

In this work, we presented a novel visual relation detection architecture, RelTransformer, which is designed for incorporating multi-level features from local, scene and a proposed memory module. Our model is able to produce
more discriminative relation representation. Experimental results demonstrate its effectiveness on long-tail relations and objects prediction. Furthermore, we achieve the state-of-the-art results on three different datasets.

References

[1] Sherif Abdelkarim, Aniket Agarwal, Panos Achlioptas, Jun Chen, Jiajii Huang, Boyang Li, Kenneth Church, and Mohamed Elhoseiny. Long tail visual relationship recognition with hubless regularized relmix. arXiv preprint arXiv:2004.00436, 2020. 2, 5, 6, 8, 12, 13

[2] Samy Bengio. The battle against the long tail. 2011. 2

[3] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, Advances in Neural Information Processing Systems, volume 33, pages 1877–1901. Curran Associates, Inc., 2020. 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. 2

[5] Long Chen, Hanwang Zhang, Jun Xiao, Xiangnan He, Shiliang Pu, and Shih-Fu Chang. Counterfactual critic multi-agent training for scene graph generation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 4613–4623, 2019. 2

[6] Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewon Jun, David Luan, and Ilya Sutskever. Generative pretraining from pixels. In International Conference on Machine Learning, pages 1691–1703. PMLR, 2020. 2

[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. 8

[8] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime G Carbonell, Quoc Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 2978–2988, 2019. 2

[9] 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. 2

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929, 2020. 2

[11] 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. 2

[12] David Ha, Andrew Dai, and Quoc V Le. Hypernetworks. arXiv preprint arXiv:1609.09106, 2016. 2

[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. 2

[14] Tao He, Lianli Gao, Jingkuan Song, Jianfei Cai, and Yuanfang Li. Learning from the scene and borrowing from the rich: Tackling the long tail in scene graph generation. In Christian Bessiere, editor, Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20, pages 587–593. International Joint Conferences on Artificial Intelligence Organization, 7 2020. Main track. 2

[15] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). arXiv preprint arXiv:1606.08415, 2016. 4

[16] Roei Herzig, Moshiko Raboh, Gal Chechik, Jonathan Bentin, and Amir Globerson. Mapping images to scene graphs with permutation-invariant structured prediction. Advances in Neural Information Processing Systems, 31:7211–7221, 2018. 5, 7

[17] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. Neural computation, 9(8):1735–1780, 1997. 2

[18] Kexin Huang, Jaan Altosaar, and Rajesh Ranganath. Clinicalbert: Modeling clinical notes and predicting hospital readmission. arXiv preprint arXiv:1904.05342, 2019. 2

[19] 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

[20] 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

[21] 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. 2, 5, 6, 13

[22] 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. 1, 2, 5

[23] J Lee, W Yoon, S Kim, D Kim, S Kim, CH So, and J Kang. Biobert: a pre-trained biomedical language representation
model for biomedical text mining. Bioinformatics (Oxford, England), 36(4):1234, 2020. 2

[24] Yikang Li, Wanli Ouyang, Xiaogang Wang, and Xiao’ou Tang. Vip-cnn: Visual phrase guided convolutional neural network. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1347–1356, 2017. 2

[25] 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. 1

[26] Yikang Li, Wanli Ouyang, Bolei Zhou, Kun Wang, and Xi-aogang 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

[27] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 2980–2988, 2017. 2, 5, 6, 13

[28] 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. 1, 5, 7

[29] 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. 2

[30] 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. 2, 5, 7

[31] 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, 6

[32] Alejandro Newell and Jia Deng. Pix2graph: From pixels to graphs by associative embedding. In Proceedings of the 31st International Conference on Neural Information Processing Systems, pages 2168–2177, 2017. 5, 7

[33] Wanli Ouyang, Xiaogang Wang, Cong Zhang, and Xiaokang Yang. Factors in finetuning deep model for object detection with long-tail distribution. In CVPR, pages 864–873, 2016. 2

[34] 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

[35] 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. 1

[36] 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. 1, 3, 6

[37] Philippe Schwaller, Teodoro Laino, Théophile Gaudin, Petter Bolgar, Christopher A Hunter, Costas Bekas, and Alp A Lee. Molecular transformer: a model for uncertainty-calibrated chemical reaction prediction. ACS central science, 5(9):1572–1583, 2019. 2

[38] Li Shen, Zhouchen Lin, and Qingming Huang. Retrival propagation for effective learning of deep convolutional neural networks. In European conference on computer vision, pages 467–482. Springer, 2016. 2

[39] Sainbayar Sukhbaatar, Édouard Grave, Piotr Bojanowski, and Armand Joulin. Adaptive attention span in transformers. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 331–335, 2019. 2

[40] Sainbayar Sukhbaatar, Édouard Grave, Guillaume Lample, Herve Jégou, and Armand Joulin. Augmenting self-attention with persistent memory. arXiv preprint arXiv:1907.01470, 2019. 7

[41] Jingru Tan, Changbao Wang, Buyu Li, Quanquan Li, Wanli Ouyang, Chenguang 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, 6

[42] 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. 2, 5, 6, 13

[43] 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

[44] 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

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

[46] Weitao Wang, Meng Wang, Sen Wang, Guodong Long, Lina Yao, Guilin Qi, and Yang Chen. One-shot learning for long-tail visual relation detection. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 34, pages 12225–12232, 2020. 2

[47] Yu-Xiong Wang, Deva Ramanan, and Martial Hebert. Learning to model the tail. In Advances in Neural Information Processing Systems, pages 7029–7039, 2017. 2

[48] 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, 7
[49] 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, 7

[50] 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

[51] 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

[52] 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. 2

[53] 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, 7

[54] 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. 1, 2, 5, 6, 7, 8, 12, 13

[55] 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. 1, 2, 5, 7
A. Supplementary material

A.1. Training and Implementation Details

To have fair and consistent comparisons with all the baseline models, we follow the same experimental setup as [1] for GQA-LT and VG8K-LT evaluation and same experimental setup as [54] for VG200 evaluation.

In GQA-LT and VG8K-LT, we train all the variants of our RelTransformer and baseline models with 8 V100 gpus. We use the batch size of 8 and sample 1 image in each batch. Initially, it starts with the learning rate (lr) 0.01, and the lr will be decreased into 0.001 and 0.0001 at the step of 967, 74 and 129, 032 in GQA-LT training and at the step of 130, 162 and 173, 550 in VG8K-LT training. We train all the models with 12 epochs on GQA-LT and 8 epochs on VG8K-LT respectively.

In VG200, we train our models for 7 epochs using 8 GPUs. The lr is 0.001 for the first 5 epochs and it will be decreased to 0.0001 in the last 2 epochs.

A.2. Further Experimental Results

We conduct additional ablation study on the feature input of scene encoder. Previously, we concatenate each triplet \(S, R, O\) as one embedding to produce consistent triplet representation, but it might lose the individual information of each object and relationship. To justify our design choice, we conduct additional experiments by directly feeding all the subjects, objects and relationships into the scene encoder as a sequence. We evaluate it on GQA-LT dataset and the results are shown in Table 6.

We denote our previous concatenation version as RelTransformer (concat), and the ablated version as RelTransformer (separate). We can observe that RelTransformer (concat) outperforms RelTransformer (separate) by 0.8% on subjects/objects but it worsens the relationship overall average accuracy by 0.8%. Under WCE setting, we can see RelTransformer (separate) worsens the subject/objects overall average accuracy by 1.7% and also worsens the relationship overall average accuracy by 9.6%. RelTransformer (concat) shows more robust and superior performance in both setup.

A.3. Full Triplet Performance Results

The full relationship triplet performance result on GQA-LT dataset is shown in Table 7. We can observe that our RelTransformer (with CE and WCE) can consistently outperform other baselines.

A.4. Per-Class Accuracy Evaluation on VG8K dataset

We quantify our model’s improvement per each class on VG8K-LT dataset and visualize them in Fig 6. It shows that RelTransformer (CE and WCE) gains the performance improvement from the medium and tail classes. In comparison with LSVRU [54], RelTransformer (WCE) improves 713 objects classes with only worsening 146 ones, and it improves 233 relationships with only worsening 10 ones, underlying the superiority of our method.

A.5. Additional Qualitative Examples

We show the relation prediction results on VG200 dataset in Fig. 7 and provide more long-tail relation prediction results on GQA-LT and VG8K-LT dataset in Fig. 8 and 9.
Table 6. Ablation results of scene encoder input. “concat” means that we concatenate the triplet \((S, R, O)\) as one consistent embedding in scene encoder, and “separate” means that all the subjects, relations and objects are directly fed into the scene encoder as a sequence. RelTransformer\textsuperscript{WCE} denotes its combination with weighted cross entropy.

| Models                  | Subjects/Objects | | Relationships | | |
|-------------------------|------------------|------------------|------------------|------------------|------------------|
|                         | many | medium | few | all | many | medium | few | all | |
| RelTransformer (concat) | 78.0 | 56.6  | 14.2 | 23.8 | 63.4 | 16.6  | 7.0 | 11.2 |
| RelTransformer (separate)| 76.6 | 55.3  | 13.4 | 22.9 | 63.5 | 16.1  | 7.9 | 12.0 |
| RelTransformer\textsuperscript{WCE} (concat) | 50.3 | 46.2  | 28.7 | 32.4 | 63.6 | 59.1  | 43.1 | 46.5 |
| RelTransformer\textsuperscript{WCE} (separate) | 50.2 | 45.9  | 26.6 | 30.7 | 59.1 | 55.6  | 32.0 | 36.9 |

Table 7. Full Relationship Triplet Performance on GQA-LT dataset.

| Dataset | Base Models | Learning Method | Many | Medium | Few |
|---------|-------------|-----------------|------|--------|-----|
|         |             |                 | SO    | SR     | OR  |
| GQA-LT  | LSVRU [54] | CE              | 38.6 | 30.3  | 31.5 |
|         | RelTransformer | CE         | 38.6 | 30.3  | 31.5 |
|         | LSVRU       | VilHub [1]     | 40.5 | 32.8  | 33.7 |
|         | LSVRU       | Focal Loss [27]| 39.2 | 31.1  | 32.3 |
|         | LSVRU       | DCPL [21]      | 30.2 | 24.7  | 25.1 |
|         | LSVRU       | Counterfactual [42]| 19.6 | 15.8  | 15.1 |
|         | LSVRU       | WCE            | 18.3 | 17.3  | 17.2 |
|         | RelTransformer | WCE         | 19.2 | 20.0  | 19.5 |

Table 7. Full Relationship Triplet Performance on GQA-LT dataset.

Figure 6. Per-class accuracy comparisons of our RelTransformer (left) and RelTransformer (WCE) (right) with LSVRU [54] baseline on VG8K-LT dataset. The green bars indicate the improvement, red bars indicate worsening and no bars means no change. The left-side y-axis represents the number of examples per class. The right-side y-axis shows the absolute accuracy improvement.
Figure 7. Qualitative results on VG200 dataset
Figure 8. More long-tail relation prediction examples on GQA-LT dataset
Figure 9. More long-tail relation prediction examples on VG8K-LT dataset