Graphical Contrastive Losses for Scene Graph Parsing

Ji Zhang[1,2], Kevin J. Shih[2], Ahmed Elgammal[1], Andrew Tao[2], Bryan Catanzaro[2]

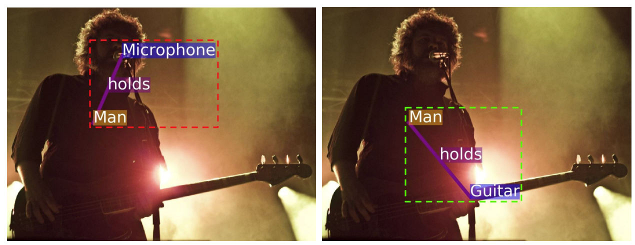
[1]Department of Computer Science, Rutgers University
[2]Nvidia Corporation

Abstract

Most scene graph parsers use a two-stage pipeline to detect visual relationships: the first stage detects entities, and the second predicts the predicate for each entity pair using a softmax distribution. We find that such pipelines, trained with only a cross entropy loss over predicate classes, suffer from two common errors. The first, Entity Instance Confusion, occurs when the model confuses multiple instances of the same type of entity (e.g. multiple cups). The second, Proximal Relationship Ambiguity, arises when multiple subject-predicate-object triplets appear in close proximity with the same predicate, and the model struggles to infer the correct subject-object pairings (e.g. mis-pairing musicians and their instruments). We propose a set of contrastive loss formulations that specifically target these types of errors within the scene graph parsing problem, collectively termed the Graphical Contrastive Losses. These losses explicitly force the model to disambiguate related and unrelated instances through margin constraints specific to each type of confusion. We further construct a relationship detector, called RelDN, using the aforementioned pipeline to demonstrate the efficacy of our proposed losses. Our model outperforms the winning method of the OpenImages Relationship Detection Challenge by 4.7% (16.5% relatively) on the test set. We also show improved results over the best previous methods on the Visual Genome and Visual Relationship Detection datasets.

1. Introduction

Given an image, the aim of scene graph parsing is to infer a visually grounded graph comprising localized entity categories, along with predicate edges denoting their pairwise relationships. This is often formulated as the detection of $\langle subject, predicate, object\rangle$ triplets within an image, e.g. $\langle man, holds, guitar\rangle$ in Figure 1b. Current state-of-the-art methods achieve this goal by a two-stage mechanism: first detecting entities, then predicting a predicate for each pair of entities.

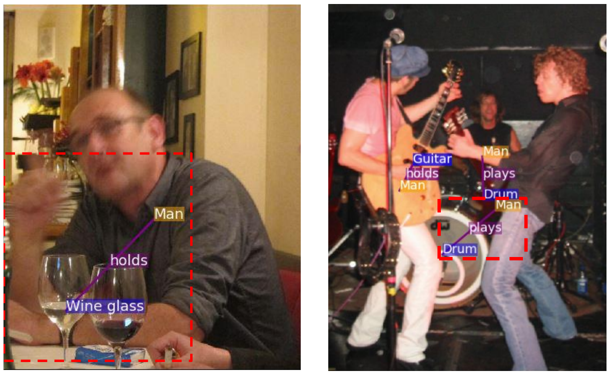


(a) without our losses (b) with our losses

Figure 1: Example of failure of models without our losses and success of our losses. (a) RelDN learned with only multi-class cross-entropy loss incorrectly relates the man with the microphone, while (b) RelDN learned with our *Graphical Contrastive Losses* detects the correct relationship $\langle man, holds, guitar\rangle$.

We find that scene graph parsing models using such pipelines tend to struggle with two types of errors. The first is **Entity Instance Confusion**, in which the subject or object is related to one of many instances of the same class, and the model fails to distinguish between the target instance and the others. We show an example in Figure 2a, in which the model identifies the man is holding a wine glass, but struggles to determine exactly which of the 3 visually similar wine glasses is being held. The incorrectly predicted wine glass is transparent and intersecting with the left arm, which makes it look like being held. The second type of error, **Proximal Relationship Ambiguity**, occurs when the image contains multiple subject-object pairs interacting in the same way, and the model fails to identify the correct pairing. An example can be seen in the multiple musicians "playing" their respective instruments in Figure 2b. Due to their close proximity, visual features for each musician-instrument pair overlap significantly, making it difficult for the scene graph models to identify the correct pairings.

The primary cause of these two failures lies in the inherent difficulty of inferring relationships such as "hold" and "play" from visual cues. For example, which glass is being



(a) Entity Instance Confusion (b) Proximal Rel Ambiguity

Figure 2: Examples of Entity Instance Confusion and Proximal Relationship Ambiguity. Red boxes highlight relationships our baseline model predicts incorrectly. (a) the man is not holding the predicted wine glass. (b) the guitar player on the right is not playing drum.

held is determined by the small part of the hand that covers the glass. Whether a player is playing the drum can only be inferred by very subtle visual cues such as his standing pose or finger placement. It is challenging for any model to learn to attend to these details precisely, and it would be impractical to specify which details to focus on for all kinds of relationships, let alone to learn all these details. These challenges motivate the need for a mechanism that can automatically learn fine details that determine visual relationships, and explicitly discriminate related entities from unrelated ones, for all types of relationships. This is the goal of our work.

In this paper we propose a set of *Graphical Contrastive Losses* to tackle these issues. The losses use the form of the margin-based triplet loss, but are specifically designed to address the two aforementioned errors. It adds additional supervision in the form of hard negatives specific to Entity Instance Confusion and Proximal Relationship Ambiguity. To demonstrate the effectiveness of our proposed losses, we design a relationship detection network named *RelDN* using the aforementioned pipeline with our losses. Figure 1 shows a result of RelDN with only the N-way cross-entropy loss *vs*. with our additional contrastive losses. Our best model achieves 0.328 on the Private set of the OpenImages Relationship Detection Challenge, outperforming the winning model by a significant 4.7% (16.5% relative) margin. It also attains state-of-the-art performance on the Visual Genome[10] and VRD[14] datasets.

In this paper, we denote subject, predicate, object and attribute with $s, pred, o, a$. We use "entity" to describe individual detected objects to distinguish from "object" in the semantic sense, and use "relationships" to describe the entire $\langle s, pred, o \rangle$ tuple, not to be confused with "predicate," which is an element of said tuple.

2. Related Work

Scene Graph Parsing: A large number of scene graph parsing approaches have emerged during the last couple of years. They use the same pipeline that first either uses off-the-shelf detectors [14, 39, 36, 3, 33, 29] or detectors fine-tuned with relationship datasets [11, 27, 35, 37, 38, 30, 28] to detect entities, then predicts the predicate using proposed methods. Most of them [14, 39, 36, 3, 33, 30, 11, 27, 35, 37] model the second step as a classification task that takes features of each entity pair as input and output a label independently from other pairs. [38] instead learn embeddings for subjects, predicates and objects and use nearest neighbor searching during testing to predict predicates. Nevertheless, the prediction is still done on each entity pair individually. We show that this pipeline struggles with two major scenarios. We find that ignoring the intrinsic graph structure of relationships and predicting each predicate separately is the main cause. Our proposed losses compensate for such drawback by contrasting positive against negative edges for each node, providing global supervision to the classifier and significantly alleviating those two issues.

The scene graph parsing work most related to ours is Associative Embedding [20]. They use use a *push* and *pull* contrastive loss to train embeddings for entities within a visual genome scene graph. Our work differs in that we propose to have different sets of hard negatives to target specific error types within scene graph parsing.

Phrase Grounding and Referring Expressions: Phrase Grounding and Referring Expression models aim to localize the region described by a given expression, with the latter focusing more on cases of possible reference confusion [31, 16, 32, 19, 7, 15, 23, 25, 13, 2, 6, 21]. It can be abstracted as a bipartite graph matching problem, where nodes on the visual side are the regions and nodes on the language side are the expressions, and the goal is to find all matched pairs. There is no semantic meanings on each pair of region/expression except positive or negative. In contrast, scene graphs are arbitrarily connected, whose nodes are visual entities and edges are predicates with much richer semantic information. Our losses are designed to leverage that information to better discriminate between related and non-related entities.

Contrastive Training: Contrastive training using a triplet loss [8] has wide application in both computer vision and natural language processing. Representative work includes Negative Sampling [17] and Noise Contrastive Sampling [18]. More recent work also utilizes it to solve multi-modal tasks such as phrase grounding, image captioning and VQA, and vector embeddings [25, 5, 31, 20]. Our setting differs in that we define hard negative contrastive margins along the known structure of the annotated scene graph, allowing us to specifically target entity instance and proximal relationship confusion. By adding our losses as additional supervision on top of the N-way cross-entropy loss, we are able to improve the model by significant margins.

3. Graphical Contrastive Losses

Our Graphical Contrastive Losses comprises three losses, each addressing the two aforementioned issues in their own way: 1) **Class Agnostic**: contrasts positive/negative entity pairs regardless of their relation and adds contrastive supervision for generic cases; 2) **Entity Class Aware**: addresses the issue in Figure 2a by focusing on entities with the same class; 3) **Predicate Class Aware**: addresses the issue in Figure 2b by focusing on entity pairs with the same potential predicate. We define our contrastive losses over an affinity term $\Phi(s,o)$, which can be interpreted as the probability that subject s and object o have some relationship or interaction. Given a model that outputs the distribution over predicate classes conditioned on a subject and object pair $p(pred|s,o)$, we define $\Phi(s,o)$ as:

$$\Phi(s,o) = 1 - p(pred = \varnothing|s,o) \tag{1}$$

where $\varnothing$ is the class symbol representing `no_relationship`. This is equivalent to summing over all predicate classes except $\varnothing$.

3.1. Class Agnostic Loss

Our first contrastive loss term aims to maximize the affinity of the lowest scoring positive pairing and minimize the affinity of the highest scoring negative pairing. For a subject indexed by i and an object indexed by j, the margins we wish to maximize can be written as:

$$\begin{aligned} m_1^s(i) &= \min_{j \in \mathcal{V}_i^+} \Phi(s_i, o_j^+) - \max_{k \in \mathcal{V}_i^-} \Phi(s_i, o_k^-) \\ m_1^o(j) &= \min_{i \in \mathcal{V}_j^+} \Phi(s_i^+, o_j) - \max_{k \in \mathcal{V}_j^-} \Phi(s_k^-, o_j) \end{aligned} \tag{2}$$

where $\mathcal{V}_i^+$ and $\mathcal{V}_i^-$ represent sets of objects related to and not related to subject s_i; $\mathcal{V}_j^+$ and $\mathcal{V}_j^-$ are defined similarly for object j as the sets of subjects related to and not related to o_j.

The class agnostic loss for all sampled positive subjects and objects is written as:

$$\begin{aligned} L_1 =& \frac{1}{N} \sum_{i=1}^{N} \max(0, \alpha_1 - m_1^s(i)) \\ &+ \frac{1}{N} \sum_{j=1}^{N} \max(0, \alpha_1 - m_1^o(j)) \end{aligned} \tag{3}$$

where N is the number of annotated entities and α_1 is the margin threshold.

This loss tries to contrast positive and negative (s,o) pairs, ignoring any class information, and is similar to the triplet losses used referring expression and phrase-grounding literature. We found it works as well in our scenario and even better with the following class-aware losses, as shown in Table 1.

3.2. Entity Class Aware Loss

The Entity Class Aware loss deals with entity instance confusion, in which the model struggles to determine interactions between a subject (object) and multiple instances of a same-class object (subject). It can be viewed as an extension of the Class Agnostic loss where we further specify a class c when populating the positive and negative sets $\mathcal{V}^+$ and $\mathcal{V}^-$. We extend the formulation in equation (3) as:

$$\begin{aligned} m_2^s(i,c) &= \min_{j \in \mathcal{V}_i^{c+}} \Phi(s_i, o_j^+) - \max_{k \in \mathcal{V}_i^{c-}} \Phi(s_i, o_k^-) \\ m_2^o(j,c) &= \min_{i \in \mathcal{V}_j^{c+}} \Phi(s_i^+, o_j) - \max_{k \in \mathcal{V}_j^{c-}} \Phi(s_k^-, o_j) \end{aligned} \tag{4}$$

where $\mathcal{V}_i^{c+}$, $\mathcal{V}_i^{c-}$, $\mathcal{V}_j^{c+}$ and $\mathcal{V}_j^{c-}$ are now constrained to instances of class c.

The entity class aware loss for all sampled positive subjects and objects is defined as

$$\begin{aligned} L_2 =& \frac{1}{N} \sum_{i=1}^{N} \frac{1}{|\mathcal{C}(\mathcal{V}_i^+)|} \sum_{c \in \mathcal{C}(\mathcal{V}_i^+)} \max(0, \alpha_2 - m_2^s(i,c)) \\ &+ \frac{1}{N} \sum_{j=1}^{N} \frac{1}{|\mathcal{C}(\mathcal{V}_j^+)|} \sum_{c \in \mathcal{C}(\mathcal{V}_j^+)} \max(0, \alpha_2 - m_2^o(j,c)) \end{aligned} \tag{5}$$

where $\mathcal{C}()$ returns the set of unique classes of the sets $\mathcal{V}_i^+$ and $\mathcal{V}_j^+$ as defined in the class agnostic loss. Compared to the class agnostic loss which maximizes the margins across all instances, this loss maximizes the margins between instances of the same class. It forces a model to disentangle confusing entities illustrated in Figure 2a, where the subject has several potentially related objects with the same class.

3.3. Predicate Class Aware Loss

Similar to the entity class aware loss, this loss maximizes the margins within groups of instances determined by their associated predicates. It is designed to deal with the proximal relationship ambiguity as exemplified in Figure 2b, where instances joined by the same predicate class are within close proximity of each other. In the context of Figure 2b, this loss would encourage the correct pairing of who is playing which instrument by penalizing wrong pairing, *i.e.*, "man plays drum" in the red box. Replacing the class groupings in equation (4) with predicate groupings restricted to predicate class e, we define our margins to maxi-

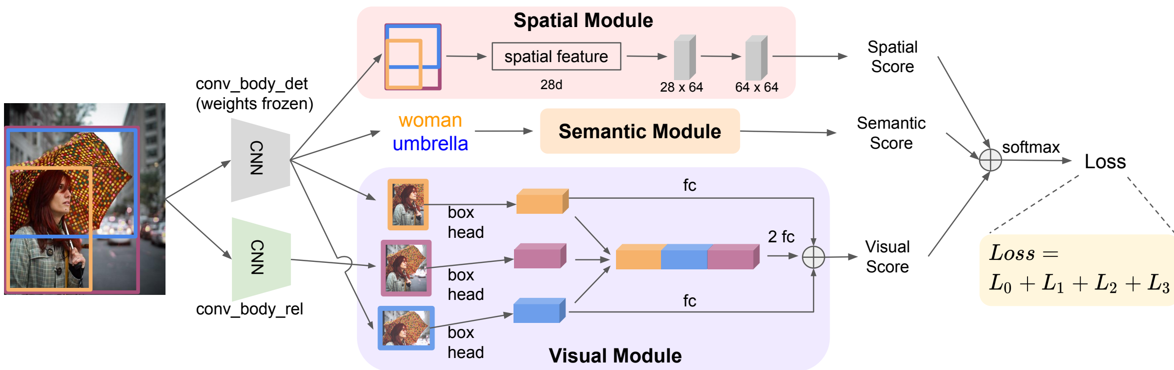


Figure 3: The RelDN model architecture. The structures of conv_body_det and conv_body_rel are identical. We freeze the weights of the former and only train the latter.

mize as:

$$
\begin{aligned}
m_3^s(i,e) &= \min_{j \in \mathcal{V}_i^{e+}} \Phi(s_i, o_j^+) - \max_{k \in \mathcal{V}_i^{e-}} \Phi(s_i, o_k^-) \\
m_3^o(j,e) &= \min_{i \in \mathcal{V}_j^{e+}} \Phi(s_i^+, o_j) - \max_{k \in \mathcal{V}_j^{e-}} \Phi(s_k^-, o_j)
\end{aligned}
\tag{6}
$$

Here, we define the sets $\mathcal{V}_i^{e+}$ and $\mathcal{V}_j^{e+}$ as the sets of subject-object pairs where the ground truth predicate between s_i and o_j is e, anchored with respect to subject i and object j respectively. We define the sets $\mathcal{V}_i^{e-}$ and $\mathcal{V}_j^{e-}$ as is the set of instances where the model *incorrectly predicts* (via argmax) the predicate to be e, anchored with respect to subject i and object j respectively.

The predicate class aware loss for all sampled positive subjects and objects is defined as

$$
\begin{aligned}
L_3 = &\frac{1}{N} \sum_{i=1}^{N} \frac{1}{|\mathcal{E}(\mathcal{V}_i^+)|} \sum_{e \in \mathcal{E}(\mathcal{V}_i^+)} \max(0, \alpha_3 - m_3^s(i,e)) \\
+ &\frac{1}{N} \sum_{j=1}^{N} \frac{1}{|\mathcal{E}(\mathcal{V}_j^+)|} \sum_{e \in \mathcal{E}(\mathcal{V}_j^+)} \max(0, \alpha_3 - m_3^o(j,e))
\end{aligned}
\tag{7}
$$

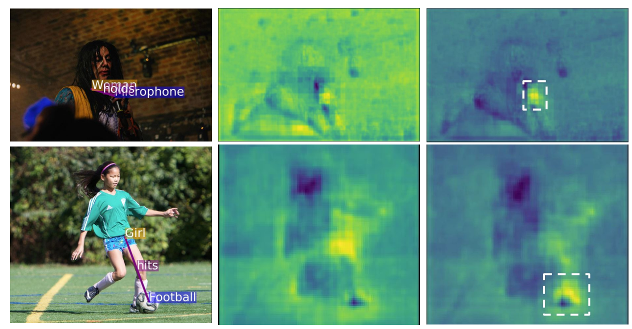
where $\mathcal{E}()$ returns the set of unique predicates associated with the input (excluding $\varnothing$). The final loss is expressed as:

$$
L = L_0 + \lambda_1 L_1 + \lambda_2 L_2 + \lambda_3 L_3 \tag{8}
$$

where L_0 is the cross-entropy loss over predicate classes.

4. RelDN

We demonstrate efficacy of our proposed losses with our Relationship Detection Network (RelDN). The RelDN follows a two stage pipeline: it first identifies a proposal set of likely subject-object relationship pairs, then extracts features from these candidate regions to perform a fine-grained classification into a predicate class. We build a separate CNN branch for predicates (conv_body_rel) with the same structure as that of entity detector CNN (conv_body_det) to extract predicate features. The intuition for having a separate branch is that we want visual features for predicates to focus on the interactive areas of subjects and objects as opposed to individual entities. As Figure 4 illustrates, the predicate CNN clearly learns better features which concentrate on regions that strongly imply relationships.



(a) ground truth (b) conv_body_det (c) conv_body_rel

Figure 4: Visualization of CNN features by averaging over the channel dimension of convolution feature maps [34]. (a) shows the image ground truth relationships, (b) shows the convolution feature from the entity detector backbone, and (c) shows the feature from the predicate backbone. In all the three examples there are clear shifts of salience from large entities to small areas that strongly indicate the predicates (highlighted in white boxes).

The first stage of the RelDN exhaustively returns bounding box regions containing every pair. In the second stage, it computes three types of features for each relationship proposal: semantic, visual, and spatial. Each feature is used to output a set of class logits, which we combine via element-wise addition, and apply softmax normalization to attain a probability distribution over predicate classes. See Figure 3 for our model pipeline.

Semantic Module: The semantic module conditions the predicate class prediction on subject-object class co-occurrence frequencies. It is inspired by Zeller, et al. [35]

which introduced a frequency baseline that performs reasonably well on Visual Genome by counting frequencies of predicates given subject and object. Its motivation is that in general, the combination of relationships between two entities is usually very limited, *e.g.*, the relationship between a person-horse subject-object pairing is most likely to be ride, walk, or feed, and unlikely to be stand on or wear. For each training image, we count the occurrences of predicate class $pred$ given subject and object classes s and o in the ground truth annotations. This gives us an empirical distribution $p(pred|s, o)$. We assume that the test set is also drawn from the same distribution.

Spatial Module: The spatial module conditions the predicate class predictions on the relative positions of the subject and object. One of the major predicate types are about positions, for example, "on", "under", or "inside_of." These predicate types can often be inferred using only relative spatial information. We capture spatial information by encoding the box coordinates of subjects and objects using the box delta [22] and normalized coordinates.

We define the delta feature between two sets of bounding box coordinates as follows:

$$\Delta(b_1, b_2) = \langle \frac{x_1 - x_2}{w_2}, \frac{y_1 - y_2}{h_2}, \log \frac{w_1}{w_2}, \log h_1 h_2 \rangle \quad (9)$$

where b_1 and b_2 are two coordinate tuples in the form of (x, y, w, h).

We then compute the normalized coordinate features for a bounding box b as follows:

$$c(b) = \langle \frac{x}{w_{img}}, \frac{y}{h_{img}}, \frac{x + w}{w_{img}}, \frac{y + h}{h_{img}}, \frac{wh}{w_{img} h_{img}} \rangle \quad (10)$$

where w_{img} and h_{img} are the width and height dimensions of the image. Our spatial feature vector for the subject, object, and predicate bounding boxes b_s, b_o, b_{pred} is represented as:

$$\langle \Delta(b_s, b_o), \Delta(b_s, b_{pred}), \Delta(b_{pred}, b_o), c(b_s), c(b_o) \rangle \quad (11)$$

Note that b_{pred} is the tightest bounding box around b_s and b_o. This feature vector is fed through an MLP to attain predicate class logit scores.

Visual Module: The visual module produces a set of class logits conditioned ROI feature maps, as in the fast-RCNN pipeline. We extract subject and object ROI features from the entity detector's convolution layers (conv_body_det in Figure 3) and extract predicate ROI features from the relationship convolution layers (conv_body_rel in Figure 3). The subject, object, and predicate feature vectors are concatenated and passed through an MLP to attain the predicate class logits.

We also include two skip-connections projecting subject-only and object-only ROI features to the predicate class logits. These skip connections are inspired by the observation that many relationships, such as human interactions [4], can be accurately inferred by the appearance of only the subjects or objects. We show an improvement from adding these skip connections in 6.3.

Module Fusion: As illustrated in Figure 3, we obtain the final probability distribution over predicate classes by adding the three scores followed by softmax normalization:

$$\mathbf{p}^{pred} = softmax(\mathbf{f}_{vis} + \mathbf{f}_{spt} + \mathbf{f}_{sem}) \quad (12)$$

where $\mathbf{f}_{vis}, \mathbf{f}_{spt}, \mathbf{f}_{sem}$ are unnormalized class logits from the visual, spatial, semantic modules.

5. Implementation Details

We train the entity detector CNN (conv_body_det) independently using entity annotations, then fix it when training our model. While previous works [11, 3, 30] claim it is beneficial to fine-tune the entity detector end-to-end with the second stage of the pipeline, we opt to freeze our entity detector weights for simplicity. We initialize the predicate CNN (conv_body_rel) with the entity detector's weights and fine-tune it end-to-end with the second stage.

During training, we independently sample positive and negative pairs for each loss, subject to their respective constraints. For L_0, we sample 512 pairs in total where 128 of them are positive. For our class-agnostic loss, we sample 128 positive subjects, then for each of them sample the two closet contrastive pairs according to Eq.2; we do the sampling symmetrically for objects. For our entity and predicate aware losses, we sample in the same way with class-agnostic except that negative pairs are grouped by entity and predicate classes, as described in Eq.4,6. We set $\lambda_1 = 1.0, \lambda_2 = 0.5, \lambda_3 = 0.1$, determined by cross-validations, for all experiments.

During testing, we take up to 100 outputs from the entity detector and exhaustively group all pairs as relationship proposals/entity pairs. We rank relationship proposals by multiplying the predicted subject, object, predicate probabilities as $\mathbf{p}^{det}(s) \cdot \mathbf{p}^{pred}(pred) \cdot \mathbf{p}^{det}(o)$ where $\mathbf{p}^{det}(s), \mathbf{p}^{det}(o)$ are the probabilities of the predicted subject and object classes from the entity detector, and $\mathbf{p}^{pred}(pred)$ is the probability of the predicted predicate class from the result of Eq.12.

To match the architectures of previous state-of-the-art methods, We use ResNeXt-101-FPN [26, 12] as our OpenImages backbone and VGG-16 on Visual Genome (VG) and Visual Relationship Detection (VRD).

6. Experiments

We present experimental results on three datasets: OpenImages (OI) [9], Visual Genome (VG) [10] and Visual Relationship Detection (VRD) [14]. We first report evaluation settings, followed by ablation studies and finally external comparisons.

L_0	L_1	L_2	L_3	R@50	wmAP$_{rel}$	wmAP$_{phr}$	score$_{wtd}$	AP$_{rel}$ per class at	on	holds	plays	interacts_with	wears	inside_of	*under*	*hits*
✓				74.67	34.63	37.89	43.94	32.40	36.51	41.84	36.04	40.43	5.70	44.17	*25.00*	*55.40*
✓	✓			75.06	35.25	38.37	44.46	32.78	36.96	42.93	37.55	43.30	**9.01**	44.15	*100.00*	*50.95*
✓		✓		74.64	35.03	38.18	44.21	32.76	36.82	42.24	37.17	40.47	8.53	44.71	*33.33*	*49.68*
✓			✓	74.88	35.19	38.27	44.36	32.88	36.73	42.38	38.03	43.53	6.71	44.18	*16.67*	*52.06*
✓	✓	✓		75.03	35.38	38.50	44.56	**32.95**	**37.10**	42.82	38.58	43.66	6.79	43.72	*20.00*	*50.24*
✓	✓		✓	**75.30**	35.30	38.27	44.49	32.92	36.73	42.58	38.81	44.13	6.35	42.74	*100.00*	*51.40*
✓		✓	✓	75.00	35.12	38.34	44.39	32.79	36.47	42.31	39.74	41.35	6.11	43.57	*25.00*	*55.12*
✓	✓	✓	✓	74.94	**35.54**	**38.52**	**44.61**	32.92	37.00	**43.09**	**41.04**	**44.16**	7.83	**44.72**	*50.00*	*51.04*

Table 1: Ablation Study on our losses. We report a frequency-balanced wmAP instead of mAP, as the test set is extremely imbalanced and would fluctuate wildly otherwise (see fluctuations in columns "under" and "hits"). We also report score$_{wtd}$, which is the official OI scoring formula but with wmAP in place of mAP. "Under" and "hits" are not highlighted due to having too few instances.

	R@50	wmAP$_{rel}$	wmAP$_{phr}$	score$_{wtd}$	AP$_{rel}$ per class at	holds	plays	interacts_with	wears	AP$_{phr}$ per class at	holds	plays	interacts_with	wears
L_0	61.72	25.80	33.15	35.92	14.77	26.34	42.51	21.33	21.03	21.76	35.88	**48.57**	38.74	31.92
$L_0 + L_1 + L_2 + L_3$	**62.65**	**27.37**	**34.58**	**37.31**	**16.18**	**30.39**	**42.73**	**22.40**	**22.14**	**22.67**	**39.60**	48.09	**40.96**	**32.64**

Table 2: Comparison of our model with Graphical Contrastive Loss *vs*. without the loss on 100 images containing the 5 classes that suffer from the two aforementioned confusions, selected via visual inspection on a random set of images.

	R@50	wmAP$_{rel}$	wmAP$_{phr}$	score$_{wtd}$	AP$_{rel}$ per class at	on	holds	plays	interacts_with	wears	inside_of	*under*	*hits*
sem only	72.98	28.73	33.07	39.32	28.62	24.52	37.04	27.33	38.37	3.16	16.34	*25.00*	*38.45*
sem + ⟨S,P,O⟩	74.97	34.70	37.96	44.06	32.26	36.26	42.44	38.47	41.63	6.50	40.97	*20.00*	*54.38*
sem + vis	**75.12**	35.22	38.33	44.44	32.68	36.83	42.09	**41.53**	42.58	**8.49**	42.31	*33.33*	*53.95*
sem + vis + spt	74.94	**35.54**	**38.52**	**44.61**	**32.92**	**37.00**	**43.09**	41.04	**44.16**	7.83	**44.72**	*50.00*	*51.04*

Table 3: Ablation study on RelDN modules. *sem only* means using only the semantic module without training any model; ⟨*S,P,O*⟩ means using only the ⟨S,P,O⟩ concatenation without the separate S,O layers in the visual module; *vis* means our full visual module, and *spt* means spatial module. "Under" and "hits" are not highlighted due to having too few instances.

	R@50	wmAP$_{rel}$	wmAP$_{phr}$	score$_{wtd}$
m = 0.1	**75.09**	35.29	38.43	44.51
m = 0.2	74.94	**35.54**	**38.52**	**44.61**
m = 0.5	74.64	35.14	38.39	44.34
m = 1.0	74.28	34.17	37.75	43.62

Table 4: Ablation study on the margin threshold m. We use $m = 0.2$ for all other experiments.

6.1. Evaluation Settings

OpenImages: The full train and val sets contains 53,953 and 3,234 images, which takes our model 2 days to train. For quick comparisons, we sample a "mini" subset of 4,500 train and 1,000 validation images where predicate classes are sampled proportionally with a minimum of one instance per class in train and val. We first conduct parameter searches on the mini set, then train and compare with the top model of the OpenImages VRD Challenge [1] on the full set. We show two types of results, one using the same entity detector from the top model, and the other using a detector trained by our own initialized by COCO pre-trained weights.

In the OpenImages Challenge, results are evaluated by calculating Recall@50 (R@50), mean AP of relationships (mAP$_{rel}$), and mean AP of phrases (mAP$_{phr}$). The final score is obtained by score $= 0.2 \times R@50 + 0.4 \times mAP_{rel} + 0.4 \times mAP_{phr}$. The mAP$_{rel}$ evaluates AP of $s, pred, o$ triplets where *both* the subject and object boxes have an IOU of at least 0.5 with ground truth. The mAP$_{phr}$ is similar, but applied to the enclosing relationship box*. In practice, we find mAP$_{rel}$ and mAP$_{phr}$ to suffer from extreme predicate class imbalance. For example, 64.48% of the relationships in val have the predicate "at," while only 0.03% of them are "under." This means a single "under" relationship is worth much more than the more common "at" relationships. We address this by scaling each predicate category by their relative ratios in the val set, which we refer to as the weighted mAP (wmAP). We use wmAP in all of our ablation studies (Table 1-4), in addition to reporting score$_{wtd}$ which replaces mAP with wmAP in the score formula.

We compare with other top models on the official evaluation server. The official test set is split into a Public and Private set with a 30%/70% split. The Public set is used as a dev set. We present individual results for both, as well as their weighted average under Overall in Table 7.

Visual Genome: We follow the same train/val splits and evaluation metrics as [35]. We train our entity detector ini-

*More details of evaluation can be found on the official page: https://storage.googleapis.com/openimages/web/vrd$_{d}etection_{m}etric.html$

L_0 only | with our losses | L_0 only | with our losses

(a) Entity Instance Confusion | (b) Proximal Relationship Ambiguity

Figure 5: Example results of RelDN with L_0 only and with our losses. The top row shows RelDN outputs and the bottom row visualizes the learned predicate CNN features of the two models. Red and green boxes highlight the wrong and right outputs (the first row) or feature saliency (the second row). As it shows, our losses force the model to attend to the representative regions that discriminate the correct relationships against unrelated entity pairs, thus is able to disentangle entity instance confusion and proximal relationship ambiguity.

tialized by COCO pre-trained weights. Following [35], we conduct three evaluations: scene graph detection(SGDET), scene graph classification (SGCLS), and predicate classification (PRDCLS). We report results for these tasks with and without the Graphical Contrastive Losses.

VRD: We evaluate our model with entity detectors initialized by ImageNet and COCO pre-trained weights. We use the same evaluation metrics as in [33], which reports R@50 and R@100 for relationship predictions at 1, 10, and 70 predicates per entity pair.

6.2. Loss Analysis

Loss Combinations: We now look at whether our proposed losses reduce two aforementioned errors without affecting the overall performance, and whether all three losses are necessary. Results in Table 1 show that the combination of all three losses plus L_0 ($L_0 + L_1 + L_2 + L_3$) consistently outperforms L_0 alone. Notably, AP$_{rel}$ on "holds" improves by from 41.84 to 43.09 (+1.3). It improves even more significantly from 36.04 to 41.04 (+5.0) on "plays" and from 40.43 to 44.16 (+3.7) on "interacts_with" respectively. These three classes suffer the most from the two aforementioned problems. Our results also show that any subset of the losses is worse than the entire ensemble. We see that $L_0 + L_1$, $L_0 + L_2$ and $L_0 + L_3$ are inferior to $L_0 + L_1 + L_2 + L_3$, especially on "holds", "plays", and "interacts_with", where the largest margin is 3.87 ($L_0 + L_2$ *vs.* $L_0 + L_1 + L_2 + L_3$ on "play").

To better verify the isolated impact of our losses, we carefully sample a subset of 100 images containing five predicates that significantly suffer from the two aforementioned problems, selected via visual inspection on a random set of images. The five predicates are "at", "holds", "plays", "interacts_with", and "wears". We sample them by looking at the raw images and select those with either entity instance confusion or proximal relationship ambiguity. Table 2 compares our losses with L_0 only on this subset. The overall gap is 1.4 and the largest gap is 4.1 at AP$_{rel}$ on "holds".

Figure 5 shows two examples from this subset, one containing entity instance confusion and the other containing proximal relationship ambiguity. In Figure 5a the model with only L_0 fails to identify the wine glass being held, while by adding our losses, the area surrounding the correct wine glass lights up. In Figure 5b $\langle woman, plays, drum\rangle$ is incorrectly predicted since the L_0-only model mistakenly pairs the unplayed drum with the singer – a reasonable error considering the amount of person-play-drum examples as well as the relative proximities between the singer and the drum. Our losses successfully suppress that region and attend to the correct microphone being held, demonstrating the effectiveness of our hard-negative sampling strategies.

Margin Thresholds: We study the effects of various values of the margin thresholds $\alpha_1, \alpha_2, \alpha_3$ used in Eq.3,5,7. For each experiment, we set $\alpha_1 = \alpha_2 = \alpha_3 = m$ while varying m. As shown in Table 4, we observe similar results with previous work [8, 24] that $m = 0.1$ or $m = 0.2$ achieves the best performance. Note that $m = 1.0$ is the largest possible margin, as our affinity scores range from 0 to 1.

	Graph Constraint									No Graph Constraint					
	SGDET			SGCLS			PRDCLS			SGDET		SGCLS		PRDCLS	
Recall at	20	50	100	20	50	100	20	50	100	50	100	50	100	50	100
Frequency+Overlap	20.1	26.2	30.1	29.3	32.3	32.9	53.6	60.6	62.2	28.6	34.4	39.0	43.4	75.7	82.9
MotifNet-LeftRight	**21.4**	27.2	30.3	32.9	35.8	36.5	58.5	65.2	67.1	**30.5**	35.8	44.5	47.7	81.1	88.3
RelDN	21.1	**28.3**	**32.7**	**36.1**	**36.8**	**36.8**	**66.9**	**68.4**	**68.4**	30.4	**36.7**	**48.9**	**50.8**	**93.8**	**97.8**

Table 5: Comparison with state-of-the-arts on VG. We only show the best two previous methods due to space limit. Full results can be found in the supplementary materials.

	Relationship		Phrase		Relationship Detection						Phrase Detection					
	free k				k = 1		k = 10		k = 70		k = 1		k = 10		k = 70	
Recall at	50	100	50	100	50	100	50	100	50	100	50	100	50	100	50	100
KL distilation[33]	22.68	31.89	26.47	29.76	19.17	21.34	22.56	29.89	22.68	31.89	23.14	24.03	26.47	29.76	26.32	29.43
Zoom-Net[30]	21.37	27.30	29.05	37.34	18.92	21.41	-	-	21.37	27.30	24.82	28.09	-	-	29.05	37.34
CAI + SCA-M[30]	22.34	28.52	29.64	38.39	19.54	22.39	-	-	22.34	28.52	25.21	28.89	-	-	29.64	38.39
RelDN (ImageNet)	21.52	26.38	28.24	35.44	19.82	22.96	21.52	26.38	21.52	26.38	26.37	31.42	28.24	35.44	28.24	35.44
RelDN (COCO)	**28.15**	**33.91**	**34.45**	**42.12**	**25.29**	**28.62**	**28.15**	**33.91**	**28.15**	**33.91**	**31.34**	**36.42**	**34.45**	**42.12**	**34.45**	**42.12**

Table 6: Comparison with state-of-the-art on VRD (− means unavailable / unknown). We only show the best three previous methods due to space constraints. Full results in supplementary materials.

Team ID	Public	Private	Overall
radek	0.289	0.201	0.227
toshif	0.256	0.228	0.237
tito	0.256	0.237	0.243
Kyle	0.280	0.235	0.249
Seiji	0.332	0.285	0.299
RelDN*	0.327	0.299	0.308
RelDN	0.320	0.332	**0.328**

Table 7: Comparison with models from OpenImages Challenge. RelDN* means using the same entity detector from *Seiji*, the champion model. Overall is computed as 0.3*Public+0.7*Private. Note that this table uses the official mAP$_{rel}$ and mAP$_{phr}$ metrics.

6.3. Model Analysis

We conduct effectiveness evaluation on the three modules of RelDN. For the visual module we also investigate the two skip-connections. As Table 3 shows, the semantic module alone cannot solve relationship detection by using language bias only. By adding the basic visual feature, *i.e*., the $\langle$S,P,O$\rangle$ concatenation, we see a significant 4.7 gain, which is further improved by adding additional separate S,O skip-connections, especially at "plays" (+3.1), "interacts_with" (+1.0), "wears" (+2.0) where subjects' or objects' appearance and poses are highly representative of the interactions. Finally, adding the spatial module gives the best results, and the most obvious gaps are at spatial relationships, *i.e*., "at" (+0.2), "on" (+0.2), "inside_of" (+2.4).

6.4. Comparison to State of the Art

OpenImages: We present results compared with top 5 models from the Challenge in Table 7. We surpass the 1st place *Seiji* by 4.7% on Private set and 2.9% on the full set, which is in fact a significant margin considering the low absolute scores and the large amount of test images (99,999 in total). Even using the same entity detector with *Seiji*, we still achieve healthy gaps (1.4% and 0.8%) on the two sets.

Visual Genome: Table 5 shows that our model is better than state-of-the-arts on all metrics. It outperforms the previous best, MotifNet-LeftRight, by a 2.4% gap on Scene Graph Detection (SGDET) with Recall@100 and by a 12.7% gap on Predicate Classification (PRDCLS) with Recall@50. Note that although our entity detector is better than MotifNet-LeftRight on mAP at 50% IoU (25.5 *vs*. 20.0), our implementation of Frequency+Overlap baseline (Recall@20: 16.2, Recall@50: 19.8, Recall@100: 21.5) is not better than their version (Recall@20: 21.0, Recall@50: 26.2, Recall@100: 30.1), indicating that our better relationship performance mostly comes from our model design.

VRD: Table 6 presents results on VRD compared with state-of-the-art methods. Note that only [30] specifically states that they use ImageNet pre-trained weights while others remain unknown. Therefore we show both results pre-trained on ImageNet and COCO. Our model is competitive to those methods when pre-trained on ImageNet, but significantly outperforms when pre-trained on COCO.

7. Conclusion

Our work presents methods to overcome two major issues in scene graph parsing: Entity Instance Confusion and Proximal Relationship Ambiguity. We show that softmax classification losses over predicate classes alone cannot leverage the scene graph's structure to adequately handle these two issues. To address this, we propose Graphical Contrastive Losses which effectively utilize semantic properties of scene graphs to contrast positive relationships against hard negatives. We carefully design three types of losses to solve the issues in three aspects. We demonstrate efficacy of our losses by adding it to a model built with the same pipeline, and we achieve state-of-the-art results on three datasets.

References

[1] Openimages visual relationship detection challenge. https://storage.googleapis.com/openimages/web/challenge.html.

[2] K. Chen, R. Kovvuri, and R. Nevatia. Query-guided regression network with context policy for phrase grounding. In *ICCV*, 2017.

[3] B. Dai, Y. Zhang, and D. Lin. Detecting visual relationships with deep relational networks. In *CVPR*, 2017.

[4] G. Gkioxari, R. Girshick, P. Dollár, and K. He. Detecting and recognizing human-object intaractions. *CVPR*, 2018.

[5] T. Gupta, K. J. Shih, S. Singh, and D. Hoiem. Aligned image-word representations improve inductive transfer across vision-language tasks. In *ICCV*, 2017.

[6] R. Hu, M. Rohrbach, J. Andreas, T. Darrell, and K. Saenko. Modeling relationships in referential expressions with compositional modular networks. In *CVPR*, 2017.

[7] R. Hu, H. Xu, M. Rohrbach, J. Feng, K. Saenko, and T. Darrell. Natural language object retrieval. In *CVPR*, 2016.

[8] R. Kiros, R. Salakhutdinov, R. S. Zemel, and et al. Unifying visual-semantic embeddings with multimodal neural language models. *TACL*, 2015.

[9] I. Krasin, T. Duerig, N. Alldrin, V. Ferrari, S. Abu-El-Haija, A. Kuznetsova, H. Rom, J. Uijlings, S. Popov, S. Kamali, M. Malloci, J. Pont-Tuset, A. Veit, S. Belongie, V. Gomes, A. Gupta, C. Sun, G. Chechik, D. Cai, Z. Feng, D. Narayanan, and K. Murphy. Openimages: A public dataset for large-scale multi-label and multi-class image classification. *Dataset available from https://storage.googleapis.com/openimages/web/index.html*, 2017.

[10] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *IJCV*, 2017.

[11] Y. Li, W. Ouyang, and X. Wang. Vip-cnn: A visual phrase reasoning convolutional neural network for visual relationship detection. In *CVPR*, 2017.

[12] T.-Y. Lin, P. Dollár, R. B. Girshick, K. He, B. Hariharan, and S. J. Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017.

[13] J. Liu, L. Wang, M.-H. Yang, et al. Referring expression generation and comprehension via attributes. In *CVPR*, 2017.

[14] C. Lu, R. Krishna, M. Bernstein, and L. Fei-Fei. Visual relationship detection with language priors. In *ECCV*, 2016.

[15] R. Luo and G. Shakhnarovich. Comprehension-guided referring expressions. In *CVPR*, 2017.

[16] J. Mao, J. Huang, A. Toshev, O. Camburu, A. L. Yuille, and K. Murphy. Generation and comprehension of unambiguous object descriptions. In *CVPR*, 2016.

[17] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, 2013.

[18] A. Mnih and K. Kavukcuoglu. Learning word embeddings efficiently with noise-contrastive estimation. In *NIPS*, 2013.

[19] V. K. Nagaraja, V. I. Morariu, and L. S. Davis. Modeling context between objects for referring expression understanding. In *ECCV*, 2016.

[20] A. Newell and J. Deng. Pixels to graphs by associative embedding. In *NIPS*, 2017.

[21] B. Plummer, L. Wang, C. Cervantes, J. Caicedo, J. Hockenmaier, and S. Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *ICCV*, 2015.

[22] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, 2015.

[23] A. Rohrbach, M. Rohrbach, R. Hu, T. Darrell, and B. Schiele. Grounding of textual phrases in images by reconstruction. In *ECCV*, 2016.

[24] I. Vendrov, R. Kiros, S. Fidler, and R. Urtasun. Order-embeddings of images and language. In *ICLR*, 2016.

[25] L. Wang, Y. Li, and S. Lazebnik. Learning deep structure-preserving image-text embeddings. In *CVPR*, 2016.

[26] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He. Aggregated residual transformations for deep neural networks. In *CVPR*, 2017.

[27] D. Xu, Y. Zhu, C. Choy, and L. Fei-Fei. Scene graph generation by iterative message passing. In *CVPR*, 2017.

[28] J. Yang, J. Lu, S. Lee, D. Batra, and D. Parikh. Graph r-cnn for scene graph generation. In *ECCV*, 2018.

[29] X. Yang, H. Zhang, and J. Cai. Shuffle-then-assemble: Learning object-agnostic visual relationship features. In *ECCV*, 2018.

[30] G. Yin, L. Sheng, B. Liu, N. Yu, X. Wang, J. Shao, and C. Change Loy. Zoom-net: Mining deep feature interactions for visual relationship recognition. In *ECCV*, 2018.

[31] L. Yu, Z. Lin, X. Shen, J. Yang, X. Lu, M. Bansal, and T. L. Berg. Mattnet: Modular attention network for referring expression comprehension. In *CVPR*, 2018.

[32] L. Yu, P. Poirson, S. Yang, A. C. Berg, and T. L. Berg. Modeling context in referring expressions. In *ECCV*, 2016.

[33] R. Yu, A. Li, V. I. Morariu, and L. S. Davis. Visual relationship detection with internal and external linguistic knowledge distillation. In *ICCV*, 2017.

[34] S. Zagoruyko and N. Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. In *ICLR*, 2017.

[35] R. Zellers, M. Yatskar, S. Thomson, and Y. Choi. Neural motifs: Scene graph parsing with global context. In *CVPR*, 2018.

[36] H. Zhang, Z. Kyaw, S.-F. Chang, and T.-S. Chua. Visual translation embedding network for visual relation detection. In *CVPR*, 2017.

[37] J. Zhang, M. Elhoseiny, S. Cohen, W. Chang, and A. Elgammal. Relationship proposal networks. In *CVPR*, 2017.

[38] J. Zhang, Y. Kalantidis, M. Rohrbach, M. Paluri, A. Elgammal, and M. Elhoseiny. Large-scale visual relationship understanding. In *AAAI*, 2019.

[39] B. Zhuang, L. Liu, C. Shen, and I. Reid. Towards context-aware interaction recognition for visual relationship detection. In *ICCV*, 2017.