

# Introduction to The 1st Place Winning Model

Ji Zhang

zhangjiapply@gmail.com

## Abstract

*This article describes the model we built that achieved 1st place in the OpenImage Visual Relationship Detection Challenge on Kaggle. Three key factors contribute the most to our success: 1) language bias is a powerful baseline for this task. We build the empirical distribution  $P(\text{predicate}|\text{subject}, \text{object})$  in the training set and directly use that in testing. This baseline achieved the 2nd place when submitted; 2) spatial features are as important as visual features, especially for spatial relationships such as “under” and “inside of”; 3) It is a very effective way to fuse different features by first building separate modules for each of them, then adding their output logits before the final softmax layer. We show in ablation study that each factor can improve the performance to a non-trivial extent, and the model reaches optimal when all of them are combined.*

## 1. Model Description

The task of visual relationship detection can be defined as a mapping  $f$  from image  $I$  to 3 labels and 2 boxes  $l_S, l_P, l_O, b_S, b_O$

$$I \xrightarrow{f} l_S, l_P, l_O, b_S, b_O \quad (1)$$

where  $l, b$  stand for labels and boxes,  $S, P, O$  stand for subject, predicate, object. We decompose  $f$  into object detector  $f_{det}$  and relationship classifier  $f_{rel}$ :

$$I \xrightarrow{f_{det}} l_S, l_O, b_S, b_O, v_S, v_O \xrightarrow{f_{rel}} l_P \quad (2)$$

The decomposition means that we can run an object detector on the input image to obtain labels, boxes and visual features for subject and object, then use these as input features to the relationship classifier which only needs to output a label. There are two obvious advantages in this model: 1) learning complexity is dramatically reduced, since we can simply use an off-the-shelf object detector as  $f_{det}$  without the need for re-training, hence the learn-able weights exist only in the small subnet  $f_{rel}$ ; 2) We have much richer features for relationships, *i.e.*,  $l_S, l_O, b_S, b_O, v_S, v_O$  for  $f_{rel}$ , instead of only the image  $I$  for  $f$ .

We further assume that the semantic feature  $l_S, l_O$ , spatial feature  $b_S, b_O$  and visual feature  $v_S, v_O$  are independent from each other. So we can build 3 separate branches of sub-networks for them. This is the basic work flow of our model.

Figure1 shows our model in details. The network takes an input image and outputs the 6 aforementioned features, then each branch uses its corresponding feature to produce a confidence score for predicates, then all scores are added up and normalized by softmax. We now introduce each module’s design and their motivation.

### 1.1. Relationship Proposal

A relationship proposal is defined as a pair of objects that is very likely related[7]. In our model we first detect all meaningful objects by running an object detector, then we simply consider each pair of objects is a relationship proposal. The following modules learn to classify each pair as either “no relationship” or one of the 9 predicates, not including the “is” relationship.

### 1.2. Semantic Module

Zeller, et al.[6] introduced a frequency baseline that performs reasonably well on Visual Genome dataset[5] by counting frequencies of predicates given subject and object. Its motivation is that in general cases, the types of relationships between two objects are usually limited, *e.g.*, given the subject being person and object being horse, their relationship is highly likely to be ride, walk, feed, but less likely to be stand on, carry, wear, *etc.* In short, the  $\langle \text{subject}, \text{predicate}, \text{object} \rangle$  composition is usually biased. Furthermore, there are numerous types of possible relationships, and any relationship detection dataset can only contain a limited number of them, making the bias even stronger.

We improved this baseline by removing the background class of subject and object and used it as our baseline. Specifically, for each training image we count the occurrence of  $l_P$  given  $l_S, l_O$  in the ground truth annotations, and we end up with an empirical distribution  $p(P|S, O)$  for the whole training set. We do this under the assumption that the test set is also drawn from the same distribution. We

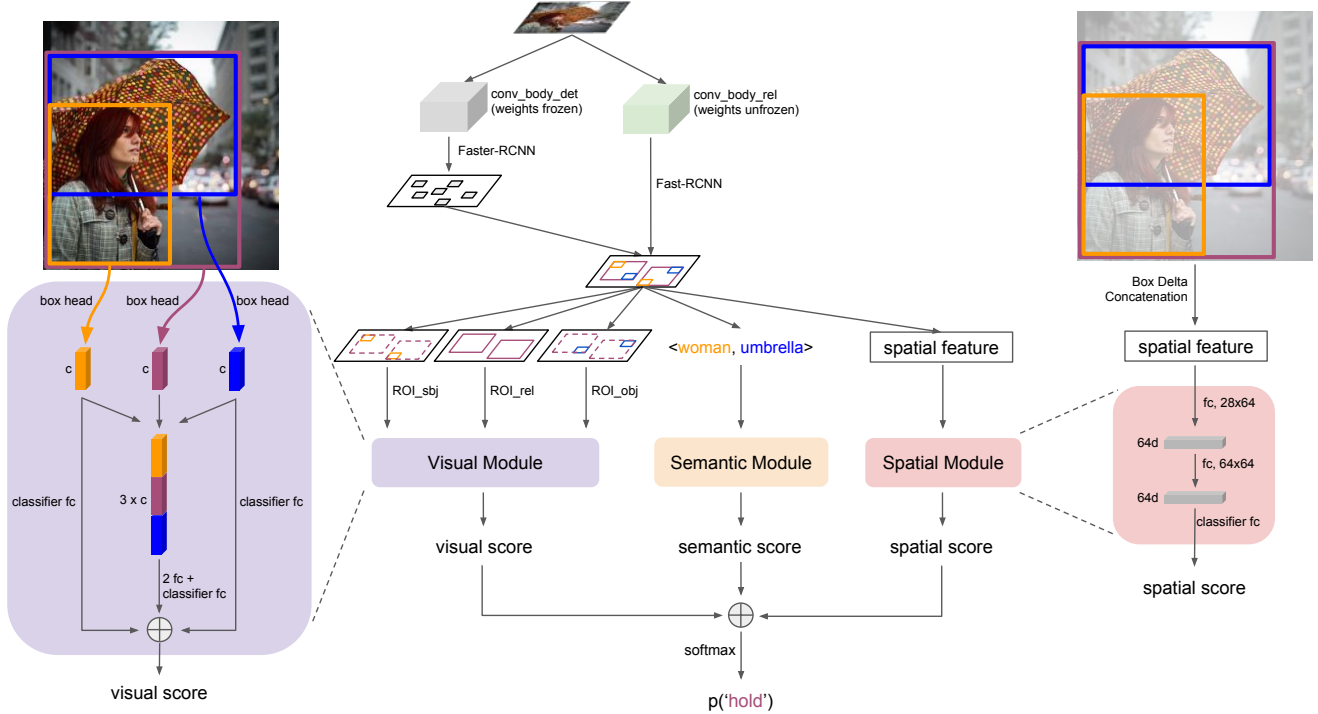


Figure 1: Model Architecture

then build the remaining modules to learn a complementary residual on top of the output of this baseline.

### 1.3. Spatial Module

In the challenge dataset, the three predicates “on”, “under”, “inside\_of” indicate purely spatial relationships *i.e.*, the relative locations of subject and object are sufficient to tell the relationship. A common solution, as applied in Faster-RCNN[4], is to learn a mapping from visual features to location offsets. However, the learning becomes significantly hard when the distance of two objects are very far[2], which is often the case for relationships. We capture spatial information by encoding the box coordinates of subjects and objects using box delta[4] and normalized coordinates:

$$\langle \Delta(b_S, b_O), \Delta(b_S, b_P), \Delta(b_P, b_O), c(b_S), c(b_O) \rangle \quad (3)$$

where  $\Delta(b_1, b_2)$  are box delta of two boxes  $b_1, b_2$ , and  $c(b)$  are normalized coordinates of box  $b$ , which are defined as:

$$\Delta(b_1, b_2) = \left\langle \frac{x_1 - x_2}{w_2}, \frac{y_1 - y_2}{h_2}, \log \frac{w_1}{w_2}, \log \frac{h_1}{h_2} \right\rangle \quad (4)$$

$$c(b) = \left\langle \frac{x_{min}}{w}, \frac{y_{min}}{h}, \frac{x_{max}}{w}, \frac{y_{max}}{h}, \frac{a_{box}}{a_{img}} \right\rangle \quad (5)$$

where  $b_1 = (x_1, y_1, w_1, h_1)$  and  $b_2 = (x_2, y_2, w_2, h_2)$ ,  $w, h$  are width and height of the image,  $a_{box}$  and  $a_{img}$  are areas of the box and image.

### 1.4. Visual Module

Visual Module is useful mainly for three reasons: 1) it accounts for all other types of relationships that spatial features can hardly predict, *e.g.*, interactions such as “man play guitar” and “woman wear handbag”; 2) it solves relationship reference problems[3], *i.e.*, when there are multiple subjects or objects that belong to a same category, we need to know which subject is related to which object; 3) for some specific interactions, *e.g.*, “throw”, “eat” “ride”, the visual appearance of the subject or object alone is very informative about the predicate. With these motivations, we feed subject, predicate, object ROIs into the backbone and get the feature vectors from its last fc layer as our visual features, then we concatenate these three features and feed them into 2 additional randomly initialized fc layers followed by an extra fc layer to get a logit, *i.e.*, unnormalized score. We also add one fc layer on top of the subject feature and another fc layer on top of the object feature to get two scores. These two scores are the predictions made solely by the subject/object feature according to the third reason mentioned above.

### 1.5. The “is” Relationship

In this challenge, “ $\langle object \rangle$  is  $\langle attribute \rangle$ ” is also considered as relationships, where there is only one object involved. We achieve this sub-task by using a completely sep-



Figure 2: Qualitative results

arate, single-branch, Fast-RCNN based model. We use the same object detector to get proposals for this model, then for each proposal the model produces a probability distribution over all attributes with the Fast-RCNN pipeline.

## 2. Implementation

### 2.1. Training

We train the regular relationship model and the “is” model separately. For the former, we train for 8 epochs using the default hyper-parameter settings from Detectron[1]. We copy layers from the first conv layer to the last fc layer as the relationship feature extractor. We freeze the the object detector’s weights but set the relationship branch learnable. We also tried setting object detector free for fine-tuning, and found that it over-fits to the few objects that appear in relationship annotations and loses generalization ability during testing. We set the ratio of negative and positive as 3 during ROI sampling. For the latter, we train for 8 epochs as well using the same default settings. We set the negative and positive ratio as 1 as we found it optimal.

### 2.2. Testing

For non-attribute relationships, we obtain a relationship score by

$$S_{SPO} = S_S \times S_P \times S_O \quad (6)$$

where  $S_S$  and  $S_O$  are obtained from object detector,  $S_P$  is the output of our relationship model. For attribute relationships, we obtain the score by

$$S_{OA} = S_O \times S_A \quad (7)$$

	R@50	mAP <sub>rel</sub>	mAP <sub>phr</sub>	Score on val	Score on public
Baseline	72.98	26.54	32.77	38.32	22.21 (2nd)
$\langle S, P, O \rangle$	74.13	32.41	39.55	43.61	-
$\langle S, P, O \rangle + S + O$	74.46	34.16	39.59	44.39	-
$\langle S, P, O \rangle + S + O + spt$	74.40	34.96	40.70	45.14	33.21 (1st)

Table 1: Ablation Study

where  $S_O$  and  $S_A$  are scores of objects and attributes. We use  $S_{SPO}$  and  $S_{OA}$  to rank all predictions and get the top 200 as final results.

## 3. Experiments

### 3.1. Ablation Study

We show performance of four models with the following settings: 1) **baseline**: only the semantic module. 2)  $\langle S, P, O \rangle$ : using semantic module and visual module without the direct predictions from subject/object. 3)  $\langle S, P, O \rangle + S + O$ : using semantic module and the complete visual module 4)  $\langle S, P, O \rangle + S + O + spt$ : our complete model.

### 3.2. Qualitative Results

We show several example outputs of our model. We can see from Figure2 that we are able to correctly refer relationships, *i.e.*, when there are multiple people playing multiple guitars, our model accurately points to the truly related pairs. One major failure case of our model is on the predicate “hold”, where the model usually needs to focus on the small area of the intersection of a human hand and the object, which our model is currently not expert at.

## References

- [1] R. Girshick, I. Radosavovic, G. Gkioxari, P. Dollár, and K. He. Detectron. <https://github.com/facebookresearch/detectron>, 2018.
- [2] G. Gkioxari, R. Girshick, P. Dollár, and K. He. Detecting and recognizing human-object interactions. *CVPR*, 2018.
- [3] R. Krishna, I. Chami, M. Bernstein, and L. Fei-Fei. Referring relationships. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.
- [4] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.
- [5] D. Xu, Y. Zhu, C. B. Choy, and L. Fei-Fei. Scene graph generation by iterative message passing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, 2017.
- [6] R. Zellers, M. Yatskar, S. Thomson, and Y. Choi. Neural motifs: Scene graph parsing with global context. In *Conference on Computer Vision and Pattern Recognition*, 2018.
- [7] J. Zhang, M. Elhoseiny, S. Cohen, W. Chang, and A. Elgammal. Relationship proposal networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5678–5686, 2017.