Introduction to The 5th Place Winning Solution

Mingyuan Zhang
Beihang University
xaphoenix@buaa.edu.cn

Jinyi Wu
National University of Singpapore
jinyi.wu@u.nus.edu

CunJun Yu
Nanyang Technological University
cyu002@e.ntu.edu.sg

Daisheng Jin
Beihang University
jindaisheng@buaa.edu.cn

Bairun Wang
Beihang University
wbrmk@buaa.edu.cn

Abstract

This article describes the method we used in the Open-Image Visual Relationship Detection Challenge on Kaggle, we achieved 5th place on the private leader board and 4th place on the public leader board. We transferred our knowledge and model structure from the task of human object interaction to this task. These two tasks are similar but also different in a lot of ways.

1. Introduction

The task of human-object interaction (HOI) detection aims to detect and classify the interactions between humans and objects in still images, such as "reading a book" or "riding a horse". HOI detection produces higher-level understanding of images than traditional computer vision tasks, such as object detection and semantic segmentation, and is a step forward from perception to comprehension. Recently, deep learning-based approaches for HOI prediction have witnessed remarkable progress. The most prevalent datasets in this field are V-COCO[1] and HICO-Det[3].

OpenImage Visual Relationship Detection(VRD) is similar to HOI as they both study relationships between two objects. However, while the subject in HOI is certainly human, the subjects in VRD are more diverse. For example, glass on table, Figure 1 is a valid relationship in VRD, but not HOI. Moreover, while most relationships in HOI are verbs, relationships in HOI include both verbs and prepositions like on, in and under. Therefore, the relationships in VRD are more spatially correlated.

In our baseline, we used a model structure we used on V-COCO and achieved a score of 0.25073 on public leader board. We then changed our method to cater to the differences between the two tasks and eventually improved our score to 0.40165.

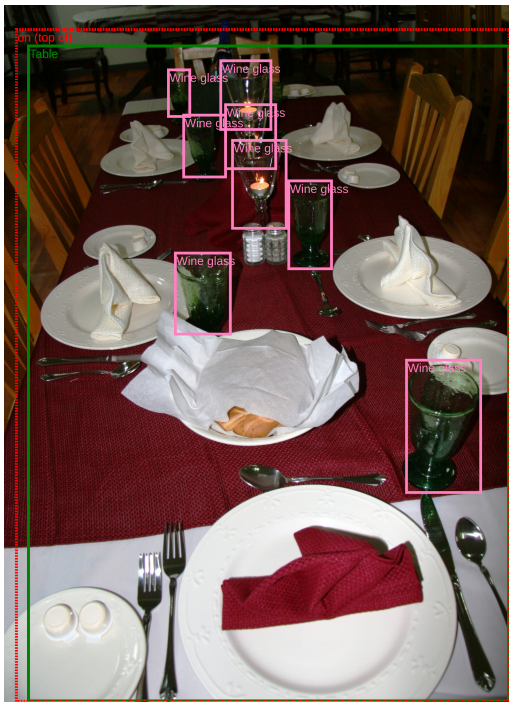

Figure 1. OpenImage Relationship.

2. Model Description

Our baseline module includes two branches, an object detection branch and a visual relationship branch. The object detection branch is a Faster-RCNN with resnet-50 as backbone and FPN. The visual relationship branch pairs bounding boxes from FPN exhaustively and concatenate their features which are roi-aligned from the backbone of the object detection branch. Then the features are passed through three fully connected layers, after which the relationship is classified. The attributes are similarly by the relationship branch. We simply consider attributes as a rela-

tionship between the thing and itself.

Our improvements in score can be mainly attributed to four modules. They are position embedding module in 2.1 which encodes the spatial relationship into the visual relationship branch, attribute head in 2.2 which separates attributes from visual relationships, a siamese attention module in 2.2 which improves the learning of visual relationships and a relationship proposal network 2.4.

2.1. Position Embedding

In our baseline the visual relationship branch contains no spatial information between the two bounding boxes. However, spatial information is very useful in inferring relationships like "on", "under" and "inside of". To solve this problem, we use positional embedding to encode the relative positions of the two bounding boxes into a $7 \times 7 \times 1$ tensor, which is concatenated with the objects feature maps and passed into the fully connected layers.

The technical details of how we embed spatial information to a tensor could be found in the paper [2].

2.2. Attribute Head

Although it is convenient to consider attribute as a form of visual relationship, such method does not achieve very good results. We reflect that attribute depends more upon the visual feature of an object. Hence it is more appropriate to put it in the object detection branch, rather than the visual relationship branch. We create a new head, which is parallel to the classification and regression branch in RCNN. The separate attribute head brings around 2 points of improvement.

2.3. Siamese attention

For better exploiting the relationship between objects, we design a new architecture called siamese attention, which is similar to the classical self-attention mechanism. To be specific, we use a 1x1 conv to process the appearance feature of the object, serving as the "Key" branch in self-attention. Also another 1x1 conv is used to transform the appearance feature of the subject as "Query" branch. Then we do matrix multiplication on these two results in the next stage with a following softmax operation to generate an attention mask. At last, we element-wise multiply two feature maps to generate the "Value" feature map and do multiplication again with the former result to acquire the final output. After that, we concatenate this output with the original appearance feature from subject, object and the union region of two bounding boxes in channel dimension. Finally, we use three res-block to fuse all these information and use a GAP layer to categorize. This method can bring us nearly 1.5 points of improvement.

2.4. Relationship Proposal Network

For the visual relationship branch, if we simply sample any pair of objects as negative samples. The negative samples would be too easy and affect the learning of relationships. Therefore we borrow the insights of Faster-RCNN and add a Relationship Proposal Network, which functions similarly as the Region Proposal Network in Faster-RCNN. The Relationship Proposal Network outputs a binary classification of a pair of objects, signaling if there is a relationship between the pair. Only false pairs with high scores in this network will be used as negative samples. Therefore this module serves not only to reduce inference time, but also help with hard negative samples mining.

3. Tricks and observations

Besides the above modules, there are also tricks that help us improve our scores. In this section, we will introduce the tricks and challenges we discover during our one month of competition.

3.1. Softmax vs Sigmoid

In our baseline, softmax is used for the classification of visual relationships. However, while we visualize the dataset, we find that there can be more than one relationship between the same pair of objects. For example one can hold and play a guitar at the same time. Similarly for the attributes, an object can be plastic and transparent. Therefore we decide to replace softmax with binary sigmoid, and obtained an improvement of 0.5.

3.2. "under" Filter

The data is very unbalanced and there are only 34 instances of the relationship "under". We achieved very low AP for the "under" category on validation set. We discovered the reason to be that we only kept the top 100 relationships in a picture and score for "under" is so low due to its rareness in training set that it is often dumped. To solve this we keep the top 50 relationships for each relationship category. Moreover, we handcrafted a function that filters pairs of objects with "under" according to their positions. The filter improves our score by two points in public leader board. We further tightened the criteria and achieved a 0.6 point increase in public, but our score dropped by 2 points in private leader board. This failed attempt shows the hazard of overfitting public test set.

3.3. Ensemble

Near the end of the competition, we trained several models with different backbones including resnext 152 and senet. To ensemble these results, we first try to use voting, and we find our score has decreased. We then chose

to merge all results and use NMS, and saw a significant increase of our score.

3.4. Unimplemented ideas

Data Cleaning. From the visualization of our model on validation set, we noticed that many annotations in the validation set are problematic. This posed quite a problem that we cannot testify our ideas on the validation set. We also suspect similar problems with the training set and planned to calculate mAP of our model on each of the training picture, rank them, find out which picture has problematic annotation and remove it. However, we did not implement this due to the workload of the task and lack of time.

Expert Model on boy, man, girl, woman Since the evaluation metric requires both object detections to have correct classification, many wrong predictions of our models involve girl misidentified as boy or girl misidentified as woman. Therefore, training an expert model with outside data should be able to improve our scores. However, we did not have time to try this.

4. Ablation study

	Public Score	Private Score
1	0.25071	0.21085
2	0.30638	0.28628
3	0.37376	0.32534
4	0.34779	0.39900

Table 1. results of models at different stages.

We put our model results at different stages into the above table. Model 1 is the baseline we mentioned above. Model 2 includes RPN, attention and attribute head. Model 3 adds sigmoid loss and ensemble to Model 2. Model 4 is our final submission with everything.

References

[1] S. Gupta and J. Malik. Visual semantic role labeling. In *arXiv preprint arXiv:1505.04474*, 2015.

[2] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, 2017.

[3] X. Liu H. Zeng Y. W. Chao, Y. Liu and J. Deng. Learning to detect human-object interactions. In *WACV*, 2018.