3rd Place Solution for Open Images 2019 - Visual Relationship

Roman Solovyev
Institute for Design Problems in Microelectronics of Russian Academy of Sciences
3, Sovetskaya Street, Moscow 124365, Russian Federation
turbo@ippm.ru

Weimin Wang
National University of Singapore
21 Lower Kent Ridge Rd, Singapore 119077
wangweimin777@gmail.com

Vladislav Golubev
Moscow State University of Railway Engineering
2, Minayevskiy Pereulok, Moscow 127055, Russian Federation
qnt.pro@gmail.com

Arthur Stsepanenka
Pavel Sukhoi Technical University
48, October ave., Gomel, 246746, Belarus
arthurstp@icloud.com

Nikolay Sergievskiy
Moscow Aviation Institute (National Research University)
4, Volokolamskoye st, Moscow 125993, Russian Federation
dereyly@gmail.com

Abstract

In this technical report, we discuss our 3rd place solution for the Visual Relationship track of Open Images 2019 competition. Specially, we demonstrate the effectiveness of using two models - 'Relationship Model' and 'Attribute Model' - to solve the task. In particular, for the 'Relationship Model', we have shown in our results that leveraging on the high quality bounding boxes of individual objects, a gradient boost based model can achieve very good performance of predicting pairs of relations between objects based on carefully designed geometric features extracted from those individual boxes.

1. Introduction

In this VR track of the Open Image Challenge 2019 [2], we are asked to detect pairs of objects and the relationships that connect them. The training set contains 329 relationship triplets with 375k training samples. These include both human-object relationships (e.g. "woman playing guitar", "man holding microphone"), object-object relationships (e.g. "beer on table", "dog inside car"), and also considers object-attribute relationships (e.g."handbag is made of leather" and "bench is wooden").

2. Dataset

The dataset for VR track is a subset of 1.7 million images provided for Object Detection track (57 out of 500 bounding boxes in total), with additional ground truth labels for Relationships and Attributes.

3. Methodology

Our solution largely consists of two models – a 'Relationship' model, and an 'Attribute' model. The final prediction is simply a combination of both outputs, as both outputs are non-overlapping.

3.1. Relationship Model

Our 'Relationship' model is based on the LightGBM framework [5], which takes around 100 geometric features extracted from pair of bounding boxes as its input. It then gives the probability of such pair being in a particular relationship.

During inference, for each test image, we will collect all predicted bounding boxes belonging to one of those 57 existing classes, and from those bounding boxes, we will prepare the exclusive list of all possible 'Relationship' pairs. Based on each pair of boxes, we will then run our algorithm to extract around 100 geometric features. These features are hand-crafted features which quantify the characteristics and statistics of each pair. Some of the features are given below as examples:

- Image aspect ratios
- angles
- areas
- box perimeters
- IOU of both boxes
- Absolute intersections
- Location of centers

The input bounding boxes we used are from our predictions in Object Detection track [1], and we found out that the quality of the Object Detection prediction directly affects our performance on Visual Relationship LB score.

To prepare for training and validation data, we created all possible pairs of bounding boxes for each relationship, based on bounding boxes ground truth files provided in Visual Relationship track. This exclusive list of bounding box pairs includes both positive (true relationship) and negative (false relationship) data, which we can use for training.

For true relationship training data, we simply use the provided Visual Relationship ground truth files, namely challenge-2019-train-vrd.csv, challenge-2019-val-vrd.csv and test-annotations-vrd.csv [3]. From these files, we will get all the positive training and validation examples. We subtract those positive pairs from the exclusive list of bounding box pairs to get negative pairs of training data.

3.2. Attribute Model

Attribute model is different from *Relationship* model in the way that each prediction is based on a single bounding box.

In total there are 42 different attribute labels, such as *Handbag is Textile*, *Ski is Wooden*, etc. Our Attribute model is a typical object detection models trained using only the 42 labeled bounding boxes as targets.

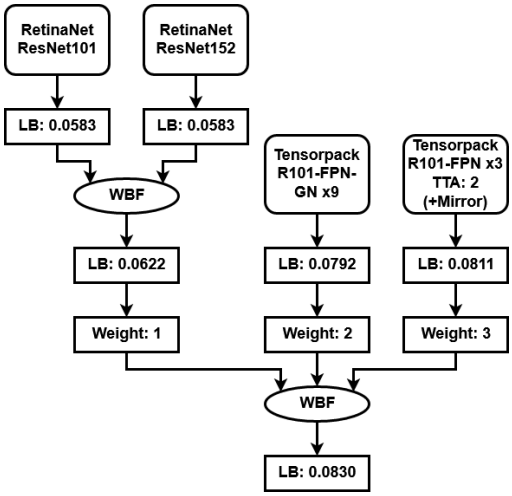

Figure 1. Attribute model illustration.

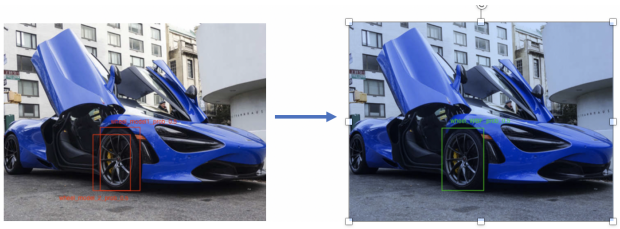

Figure 2. Example of ensembling two models' bounding box predictions using WBF. The red boxes are predicted boxes from two models, and the predicted probabilities are 0.5 for each; the green box is the final ensembled box using our algorithm

Our overall solution architecture is shown in Figure 1 above. In total, it has four single models – two RetinaNet [6] with ResNet101 and ResNet152 backbones [4], as well as two Faster RCNNs [7] – both using ResNet101 as backbones and with FPN, and one is Cascade x3 and another is Cascade x9 with GroupNorm. The ensemble approach is using Weighted Box Fusion which will be discussed in the next Section.

3.3. Weighted Box Fusion

When ensembling bounding boxes, there will be many overlapping boxes that have high IOU values (i.e. IOU $>$0.5) with each other. These boxes are from different models whose information may be lost if the boxes are simply removed. Therefore, instead of directly removing those boxes with lower probabilities like in algorithm of NMS, we will weighted average the boxes based on their coordinates and probabilities.

In our proposed WBF algorithm, for each class in each image, we will first find all overlapping bounding boxes with IOU larger than a pre-defined threshold. Second,

within those selected boxes, we will weighted average each of the four coordinates across all boxes. The weighted averaging is done based on each box's prediction score. Figure 2 shows an example of using WBF to ensemble two boxes.

4. Conclusion

To get a high accuracy in Visual Relationship task you need strong Object Detection model. Even a small improvement in OD gives large step on LB. Also it's better to separate Attribute and Relationship models. Relationship model is strongly depends on accuracy of OD boxes and geometric relationships between them. We got more than 0.98 AUC on validation for our LightGBM model to find correct relationship.

References

[1] Kaggle competition: Open images 2019 - object detection. https://www.kaggle.com/c/open-images-2019-object-detection, 2019.

[2] Kaggle competition: Open images 2019 - visual relationship. https://www.kaggle.com/c/open-images-2019-visual-relationship, 2019.

[3] Visual relationships detection track annotations. https://storage.googleapis.com/openimages/web/challenge2019_downloads.html, 2019.

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[5] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: A highly efficient gradient boosting decision tree. In *NIPS*, 2017.

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

[7] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. *CoRR*, abs/1506.01497, 2015.