Unbiased Teacher for Semi-Supervised Object Detection

Haowen Guan Xuan Zhao Qi Dong

Abstract

This is the project for NYU semi-supervised object detection competition. We implemented a semi-supervised object detection approach based on the Unbiased Teacher 2.0 (Liu et al., 2022). Our approach involved training in two steps: first train a supervised model using the 30000 images labeled dataset, then using it as the pre-trained weight to train an semi-supervised unbiased teacher model using both labeled and additional 512000 unlabeled images. Our approach was successful in improving the performance of the object detection model and leveraging the large amount of unlabeled data to improve the performance of the object detection model. Our group achieved a final AP of 25.2.

1. Introduction

1.1. Supervised Object Detection

Object detection is a task to localize and recognize objects of interest in an image. It inputs an image with one or more objects, and predicts a bounding box around each object with the corresponding category.

Supervised object detection uses only labeled data to do the training. There are two types of supervised object detection model: one is the two-stage approach, and the other is the single-stage approach.

Faster R-CNN (Ren et al., 2015) is a popular two-stage object detector. In stage one, it feeds features to the Region Proposal Network (RPN) to extract object proposals, and then uses Non-Maximum Suppression (NMS) to remove redundant and low-quality object proposals. In stage two, it extracts a pooled feature map for each proposal and feeds the pooled features into a Box Head or Region-of-Interest head to predict the object category.

Single-stage object detectors are generally faster and simpler than two-stage object detectors, at the expense of lower prediction quality. Single-stage detectors first predict objects at predefined locations, and then subsequently refine box locations and aspect ratios. DETR (Carion et al., 2020), is a transformer-based single-stage detector that has a much simpler architecture than Faster R-CNN. It starts from feeding an image to the backbone to extract features, and then feeds the features to the transformer encoder and decoder to output up to 100 predictions, which consist of bounding box location, object categories, and confidence score. The highest confidence predictions are returned. DETR removes the need for NMS, which is non-differentiable, by removing redundant detections.

There has been a recent trend to use Self-Supervised learning to pretrain the backbone on supervised detectors. There are predictive approaches, that re-predict the position of automatically generated ”ground-truth” crops. For example, in UP-DETR (Dai et al., 2021), it partitions the object queries into K groups, adds a different random crop to each group, and then feeds their corresponding features to the decoder input. The loss is computed by finding the optimal matching between the predicted boxes and the ”ground-truth” random crops. There are also contrastive approaches, that contrast backbone representation locally at feature or crop level. In ReSim (Xiao et al., 2021), two overlapping crops are generated from two different views of the same image. Then, a sliding window is moved across the overlapping regions, and the pooled representations are contrasted in the final convolutional layers.

1.2. Semi-Supervised Object Detection

Semi-Supervised Object Detection, different from Supervised Detection and Self-Supervised Pretraining, uses both labeled and unlabeled data when training. There are consistency-based method (Jeong et al., 2019), which enforces the predictions of an input image and its flipped version to be consistent, and pseudo-label-based method, which uses a small amount of labeled data to pretrain a detector and generates pseudo-labels on unlabeled data to fine-tune the pre-trained detector. In this project, we used the pseudo-label-based method.

1.3. Description of the competition dataset

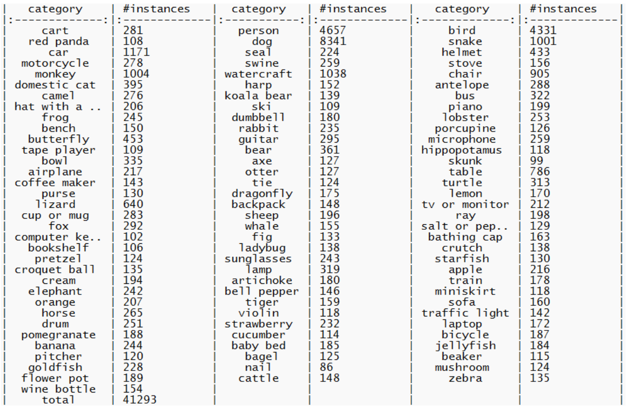
The competition dataset consists of 512,000 unlabeled images, 30,000 labeled training images, and 20,000 labeled validation images with a total of 100 classes. Since the number of labeled images is limited, it is important to make use of unlabeled dataset when training. We used the unlabeled and labeled training dataset for training, and the validation dataset for choosing the best model. Since we were using a custom dataset instead of the standard COCO dataset, we also needed to register the dataset when training and evaluating. Besides, because the model we used takes in an annotation json file, we needed to store the categories, images, and annotations information in a dictionary and convert it to a COCO format json file.

In this project, we used the COCO metrics, namely Average Precision (AP) @ IoU=0.50:0.95 to evaluate the model. Precision is the proportion of True Positives over True Positives and False Positives, whereas Recall is the proportion of True Positives over True Positives and False Negatives. Average Precision(AP) is the Area under Precision-Recall Curve, where x-axis is Recall and y-axis is Precision. In the COCO setting, AP is averaged over all categories, which is traditionally called Mean Average Precision(mAP), but they make no distinction between AP and mAP. Intersection over Union(IoU) is simply a way to measure the amount of overlap between two bounding boxes. 0.5 IoU means that if a detection has an IoU less than 0.5, it is going to be treated as False Positives. AP @ IoU=0.50:0.95 means to calculate the AP from 0.5 IoU to 0.95 IoU at a step of 5%, and then average those 10 values.

2. Method

In this project, we used Unbiased Teacher 2.0 (Liu et al., 2022), as our Semi-Supervised Object detection training model. Unbiased Teacher consists of two training stages, Burn-In stage and Teacher-Student Mutual Learning stage. In Burn-In stage, the model trains the object detector using the available supervised data to initialize the detector and duplicate it to two models. In Teacher-Student mutual learning stage, the fixed teacher generates pseudo-labels to train the student, while teacher and student are given weakly and strongly augmented inputs respectively. The knowledge that the student learned is then transferred to the slowly progressing teacher via exponential moving average (EMA).

We first attempt to train the unbiased teacher model from scratch. However, the attempt fail as the model suffered from gradient exploding and often exited with an error. To address this issue, we decided to train a supervised Faster RCNN-R50 + FPN (Ren et al., 2015) model from scratch with the labeled data using Detectron2 (Wu et al., 2019). Then, we take the weights of supervised model as the pretrained weights and transfer into the unbiased teacher model.



category	#instances	category	#instances	category	#instances
cart	281	person	4657	bird	4331
red panda	108	dog	8341	snake	1001
car	1171	seal	224	helmet	433
motorcycle	278	swine	259	stove	156
monkey	1004	watercraft	1038	chair	905
domestic cat	395	harp	152	antelope	288
camel	276	koala bear	139	bus	322
hat with a ..	206	ski	109	piano	199
frog	245	dumbbell	180	lobster	253
bench	150	rabbit	235	porcupine	126
butterfly	453	guitar	295	microphone	259
tape player	109	bear	361	hippopotamus	118
bowl	335	axe	127	skunk	99
airplane	217	otter	127	table	786
coffee maker	143	tie	124	turtle	313
purse	130	dragonfly	175	lemon	170
lizard	640	backpack	148	tv or monitor	212
cup or mug	283	sheep	196	ray	198
fox	292	whale	155	salt or pep..	129
computer ke..	102	fig	133	bathing cap	163
bookshelf	106	ladybug	138	crutch	138
pretzel	124	sunglasses	243	starfish	130
croquet ball	135	lamp	319	apple	216
cream	194	artichoke	180	train	178
elephant	242	bell pepper	146	miniskirt	118
orange	207	tiger	159	sofa	160
horse	265	violin	118	traffic light	142
drum	251	strawberry	232	laptop	172
pomegranate	188	cucumber	114	bicycle	187
banana	244	baby bed	185	jellyfish	184
pitcher	120	bagel	125	beaker	115
goldfish	228	nail	86	mushroom	124
flower pot	189	cattle	148	zebra	135
wine bottle	154				
total	41293				

Figure 1. Labeled Training Dataset Number of Instance

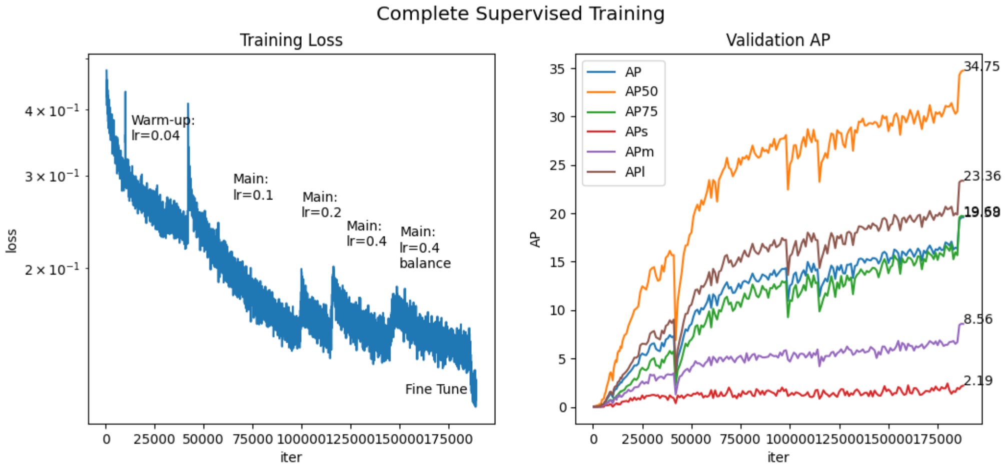


Figure 2. Supervised Training Loss and Validation AP

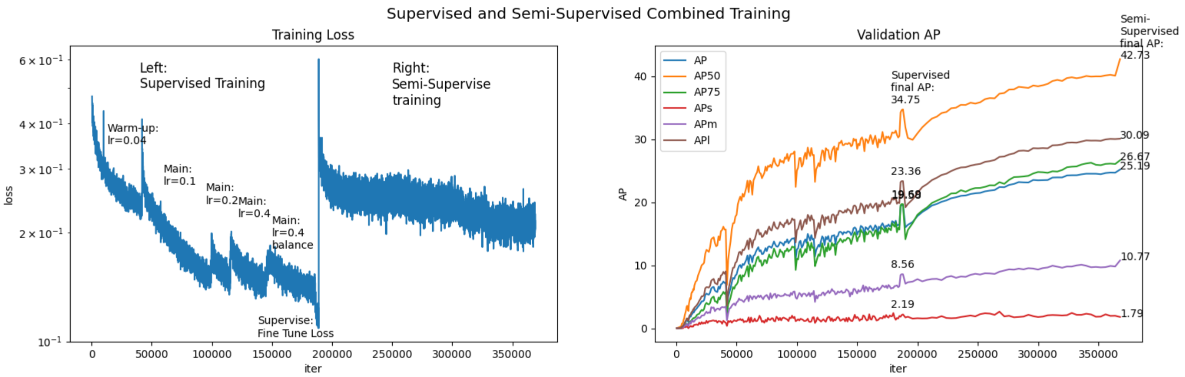


Figure 3. Supervised + Semi-supervised Combined Training Loss and Validation AP

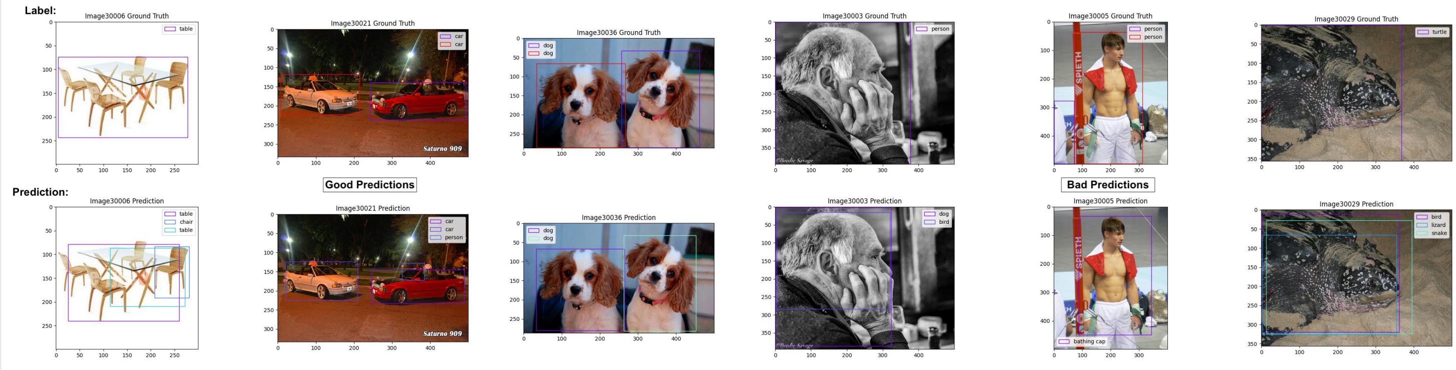


Figure 4. Good and Bad Prediction Example

2.1. Training the Supervised Model

We first proceeded the supervised training on 30000 labeled images using 6 GPUs, batch size of 60, and initialized the weight from scratch. For learning rate, we started from a very small learning rate of 0.04 combined with 3000 warm-up iteration to prevent gradient exploding. Then we gradually raised learning rate to 0.1, 0.2, 0.4 to increase the convergence speed. We achieved an initial AP of 15.5 after the main training. During late period of training, we discovered that the labeled dataset was imbalanced (Figure 1): some categories, such as dog, person, and bird, have over 4000 instances in the training data, whereas other categories, such as laptop, train, and nail, only have less than 200 instances. Therefore, we experimented and applied `RepeatFactorTrainingSampler` dataset balance technique, and it improved the AP to 16.5. Finally, we fine tuned the model by gradually decreasing the learning rate. Our supervised model achieved a final AP of 19.5. Complete supervised training logs is in Figure 2).

2.2. Training the Unbiased Teacher 2.0 Model

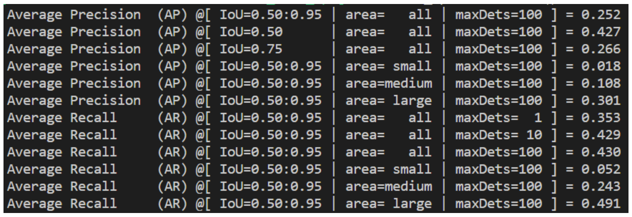
With supervised model weight as the pretrained weight for teacher and student, we started semi-supervised training using the unbiased teacher 2.0 approach. Based on the original paper, the unbiased teacher model is typically trained for 16 coco dataset epochs to reach its final model. Since the NYU dataset used in this competition has 512,000 images, with a batch size of 18 + 18, one NYU dataset epoch is equivalent to 14,000 training iterations. Therefore, to train the unbiased teacher for the equivalent of 16 epochs, we aimed to train for a total of 225,000 iterations. During main training, we used a learning rate of 0.1, which improved AP from supervise model's 19.5 to 23.5. Then, we proceeded fine tuning of unbiased teacher by gradually decreasing the learning rate. Different from unbiased teacher 2.0 (Liu et al., 2022) which didn't include a fine tuning step, we observed the fine tuning does benefit in certain degree and it elevates the AP to 24.7. We noticed that the supervised fine tuning method has more potential than unbiased teacher 2.0, so, we transferred the teacher's weight back to the supervised model to perform fine-tuning again. Finally, Our model achieved a AP of 25.2 on the validation set. Complete supervised + semi-supervised training logs is in Figure 3.

3. Result and Evaluation

Our final model achieved an AP of 25.2. Figure 5 shows the overall evaluation statistics. Figure 6 shows the AP by each category. Some categories that have over 8000 instances in the training data, such as dog, also have an AP as high as 66.44, and some categories that have less than 100 instances, such as nail, have an AP as low as 3.20. Exceptions exist as well. For example, person has over 4000 instances in the training dataset, but it only has an AP of 15.11; train has less than 200 instances, but it has an AP of 41.41. As a result, besides the number of instances, object representations, such as whether or not this category is easy to be classified or not, is also an important factor that determines the accuracy of this category. We provide some good example and bad example of our model's prediction in Figure 4.

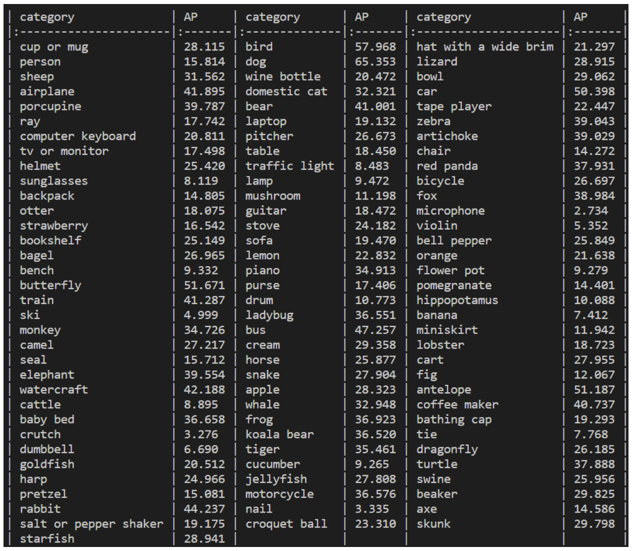
4. Conclusion

In this report, we presented a semi-supervised object detection approach based on the Unbiased Teacher 2.0 (Liu et al., 2022). Our approach involved training a supervised model using a labeled dataset, and then using that model as a pretrained weight to train an unbiased teacher model using both labeled and unlabeled data. We found that our approach was successful in improving the performance of the object detection model. The final unbiased teacher model outperformed the supervised model, achieving an AP of 24.7 compared to 19.5 for the supervised model. This suggests that our approach was effective in leveraging the large amount of unlabeled data to improve the performance of the object detection model. To make a final breakthrough on AP, we propose a novel fine-tune approach by transferring the semi-supervised weight back to supervised model, and we are able to elevate the AP even more from 24.7 to 25.2. Our approach provides a promising solution for semi-supervised object detection, and could be applied to a variety of object detection tasks. Further research could be focusing on exploring different architectures and training strategies to further improve the performance of the model.



```
Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.252
Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.427
Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.266
Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = 0.018
Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.108
Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.301
Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.353
Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.429
Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.430
Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = 0.052
Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.243
Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.491
```

Figure 5. Overall Results of the Final Model



```
| category              | AP     | category      | AP     | category             | AP     |
|:----------------------|:-------|:--------------|:-------|:---------------------|:-------|
| cup or mug            | 28.115 | bird          | 57.968 | hat with a wide brim | 21.297 |
| person                | 15.814 | dog           | 65.353 | lizard               | 28.915 |
| sheep                 | 31.562 | wine bottle   | 20.472 | bowl                 | 29.062 |
| airplane              | 41.895 | domestic cat  | 32.321 | car                  | 50.398 |
| porcupine             | 39.787 | bear          | 41.001 | tape player          | 22.447 |
| ray                   | 17.742 | laptop        | 19.132 | zebra                | 39.043 |
| computer keyboard     | 20.811 | pitcher       | 26.673 | artichoke            | 39.029 |
| tv or monitor         | 17.498 | table         | 18.450 | chair                | 14.272 |
| helmet                | 25.420 | traffic light | 8.483  | red panda            | 37.931 |
| sunglasses            | 8.119  | lamp          | 9.472  | bicycle              | 26.697 |
| backpack              | 14.805 | mushroom      | 11.198 | fox                  | 38.984 |
| otter                 | 18.075 | guitar        | 18.472 | microphone           | 2.734  |
| strawberry            | 16.542 | stove         | 24.182 | violin               | 5.352  |
| bookshelf             | 25.149 | sofa          | 19.470 | bell pepper          | 25.849 |
| bagel                 | 26.965 | lemon         | 22.832 | orange               | 21.638 |
| bench                 | 9.332  | piano         | 34.913 | flower pot           | 9.279  |
| butterfly             | 51.671 | purse         | 17.406 | pomegranate          | 14.401 |
| train                 | 41.287 | drum          | 10.773 | hippopotamus         | 10.088 |
| ski                   | 4.999  | ladybug       | 36.551 | banana               | 7.412  |
| monkey                | 34.726 | bus           | 47.257 | miniskirt            | 11.942 |
| camel                 | 27.217 | cream         | 29.358 | lobster              | 18.723 |
| seal                  | 15.712 | horse         | 25.877 | cart                 | 27.955 |
| elephant              | 39.554 | snake         | 27.904 | fig                  | 12.067 |
| watercraft            | 42.188 | apple         | 28.323 | antelope             | 51.187 |
| cattle                | 8.895  | whale         | 32.948 | coffee maker         | 40.737 |
| baby bed              | 36.658 | frog          | 36.923 | bathing cap          | 19.293 |
| crutch                | 3.276  | koala bear    | 36.520 | tie                  | 7.768  |
| dumbbell              | 6.690  | tiger         | 35.461 | dragonfly            | 26.185 |
| goldfish              | 20.512 | cucumber      | 9.265  | turtle               | 37.888 |
| harp                  | 24.966 | jellyfish     | 27.808 | swine                | 25.956 |
| pretzel               | 15.081 | motorcycle    | 36.576 | beaker               | 29.825 |
| rabbit                | 44.237 | nail          | 3.335  | axe                  | 14.586 |
| salt or pepper shaker | 19.175 | croquet ball  | 23.310 | skunk                | 29.798 |
| starfish              | 28.941 |               |        |                      |        |
```

Figure 6. Average Precision by Category

References

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., and Zagoruyko, S. End-to-end object detection with transformers. In *European conference on computer vision*, pp. 213–229. Springer, 2020.

Dai, Z., Cai, B., Lin, Y., and Chen, J. Up-detr: Unsupervised pre-training for object detection with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 1601–1610, 2021.

Jeong, J., Lee, S., Kim, J., and Kwak, N. Consistency-based semi-supervised learning for object detection. In *Neural Information Processing Systems*, 2019.

Liu, Y.-C., Ma, C.-Y., and Kira, Z. Unbiased teacher v2: Semi-supervised object detection for anchor-free and anchor-based detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9819–9828, June 2022.

Ren, S., He, K., Girshick, R., and Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.

Wu, Y., Kirillov, A., Massa, F., Lo, W.-Y., and Girshick, R. Detectron2. https://github.com/facebookresearch/detectron2, 2019.

Xiao, T., Reed, C. J., Wang, X., Keutzer, K., and Darrell, T. Region similarity representation learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10539–10548, 2021.