RETHINKING PSEUDO-LABELED SAMPLE MINING FOR SEMI-SUPERVISED OBJECT DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Consistency-based method has been proved effective for semi-supervised learning (SSL). However, the impact of the pseudo-labeled samples' quality as well as the mining strategies for high quality training sample have rarely been studied in SSL. An intuitive idea is to select pseudo-labeled training samples by threshold. We find it essential the selection of these thresholds to the final result of SSL. Following this discovery, we propose SEAT (**S**core **E**nsemble with **A**daptive **T**hreshold), a simple and efficient semi-supervised learning object detection method, in which the high confidence pseudo-labels are selected for self-training. Apart from confidence score as the indicator of the sample's quality, we also introduce the scores of temporal consistency and augmentation consistency. The scores provide a more comprehensive description to the quality of each sample. To cope with the data distribution difference among categories, the adaptive threshold strategy is used to automatically determine the sample mining threshold for each category. We conduct experiments on PASCAL-VOC and MSCOCO, extensive results show that our method is competitive and can be easily combined with consistency-based methods.

1 INTRODUCTION

Over the years, as the development of deep learning, large scale data are playing a more and more important role in promoting the performance of different machine learning tasks. How to make use of the huge amount of unlabeled data becomes a central topic among researchers in the machine learning society. Especially for computer vision tasks, large scale images are needed to train the model with hundreds of thousands of parameters.

Augmentation consistency has been proven effective for semi-supervised learning Miyato et al. (2018); Laine & Aila (2016); Tarvainen & Valpola (2017); Jeong et al. (2019); Sohn et al. (2020); Xie et al. (2020). The main idea is to add a regularization term to the generated pseudo-labels on unlabeled data by the model. The regularization term is defined as minimizing the pseudo-labels' dissimilarity between the teacher and student models. The images are processed using different type and strength of augmentations before fed into the teacher and student models correspondingly Jeong et al. (2019). Augmentation consistency works by stabilizing the training process on unlabeled data and improves generalization ability of the model. In the field of image classification, RandAugment Cubuk et al. (2019) and CTAugment Berthelot et al. (2019a) are good examples to show the power of augmentation consistency.

As for object detection, it also turns out to be a powerful strategy to adapt the detectors on unlabeled data with self-training. However, the refinery of the pseudo-labels are rarely noticed and studied. In the preliminary study, we show the importance of pseudo-label threshold for semi-supervised learning. One common coping strategy is to filter out less confident unlabeled samples Jeong et al. (2019)Sohn et al. (2020)Tang et al. (2020). It does help to improve the overall quality of the pseudo-labels, but it ignores the data distribution inconsistency among different classes. Given a static threshold, there exists a high risk to introduce noises to pseudo labels by filtering out many low confident positive samples and keeping high confident negative samples. Another challenge is how to make good use of middle confident unlabeled samples. Samples with confidence scores that are close to the threshold are essential for self-training, since these samples contribute to draw a precise

classification boundary. However, the pseudo-labels of these samples convey the most amount of noises than other samples.

In semi-supervised learning, unlabeled data are used for training together with pseudo-labels predicted by the teacher model. The quality of unlabel training samples with pseudo-labels play an essential role in semi-supervise learning. Usually, these training samples are selected according to the confidence score predicted by the detectors. Candidate samples with confidence scores higher than a given threshold like 0.8 are seen as high quality training sample. However, there are two major challenges. First, the confidence score should not be the only indicator of the quality of the training samples. In practice, even if the threshold is set as high as possible, like 0.9, there still exists some high confident false positive samples which significantly brings noise to semi-supervised training. More scores that reveals more detailed information from other aspects are needed for high quality sample mining. Second, the confidence scores of unlabled data does not follow the same distribution of different categories. Setting the same threshold for sample selection is an inaccurate way. For instance, in the same unlabeled dataset, there are more highly confident samples of the category CAR than PERSON. In this case, the threshold of the category CAR shall be set higher than that of the category PERSON.

To cope with above challenges, we propose SEAT (**S**core **E**nsemble with **A**daptive **T**hreshold), a simple and efficient semi-supervised learning object detection method to deal with above two challenges. In SEAT, the high confidence pseudo-labels are refined for self-training. We use multiple scores apart from the confidence score predicted by the detector itself (SE, Score Ensemble). The threshold to filter in/out unlabeled samples is calculated dynamically according to the data distribution of each class (AT, Adaptive Threshold). First, to provide more information to guide the pseudo-labeled sample mining, we introduce the scores of temporal consistency and augmentation consistency. Second, to deal with the difference of confidence score distribution among categories, we propose an adaptive threshold calculation method. For categories where the confidence scores obeys long-tail distribution, we assume that there are more false positive samples mixed with true positive samples. The thresholds of these categories are set higher to filter out the FP samples. For categories where the confidence scores aggregates around 1 and 0, we assume that there are fewer FP samples mixed with TP samples. The corresponding thresholds are set lower.

In this paper, we implement SEAT both on single stage and two stage detectors (Yolov3 and Faster RCNN). We compare with state-of-the-art methods STAC, and CSD. Through experiments, we show that our framework is flexible to be combined with previous methods, and generates competitive results. We have the following contributions:

- As far as we know, we are the first to stress the importance of thresholding and conduct extensive experiments to study its properties.
- We propose SEAT, a simple and efficient semi-supervised learning object detection method. In SEAT, we design multiple scores to help pseudo-label sample selection from the candidates generated by the teacher model, and dynamic threshold to deal with classwise data distribution imbalance of the pseudo-label samples.
- SEAT is a flexible method which can be easily combined with other SSL methods such as Sohn et al. (2020) and Jeong et al. (2019). And we show SEAT helps to promote the performance of the state-of-the-art methods because it provides more reliable pseudo-label samples for semi-supervised training.

2 Related Work

Semi-Supervised Learning (SSL) aims to make use of large scale unlabel data to improve the model where the images are obtained almost for free. Consistency regularization becomes popular in this field because of its effectiveness and flexibility. VAT Miyato et al. (2018), Temporal Ensemble Laine & Aila (2016), and Mean-Teacher Tarvainen & Valpola (2017) study semi-supervised image classification with consistency regularization by adding different disturbs to the predictions on the unlabeled data. Proven its efficiency, consistency regularization is then promoted to semi-supervised object detection. In CSD Jeong et al. (2019), consistency regularization is applied to both the classification as well as localization branches. To avoid mismatch of bounding boxes, only horizontal flip is used as the augmentation method to evaluate the consistency on unlabeled images.

Another methodology to energize the unlabel data is self-training. The model is first trained on the labeled data to get relatively good representation ability. Then for each unlabeled sample x_u, the model predicts its pseudo-label y_u. Due to the ability limitation of the model, there is usually much noise in the pseudo-labels, which has a large impact on SSL. How to generate high quality pseudo-labeled samples from the unlabeled data becomes an essential problem. MixMatch Berthelot et al. (2019b) generates the pseudo-labels using the average predictions of unlabeled images on several augmentations. The unlabeled images are mixed with the labeled images and used for self-training. In Note-RCNN Gao et al. (2019), an ensemble of two classification head of Faster-RCNN is used to overcome the disturb from the noisy pseudo-labels.

Both consistency regularization and high quality pseudo-labeled sample mining can also be combined. In STAC Sohn et al. (2020), high confidence threshold is used to obtain high quality pseudo-labels. However, we argue that a fixed high threshold is not enough for SSL due to data distribution difference among categories.

Object Detection can be divided into two categories, anchor based and anchor-free. In this paper, we only concentrate on anchor based methods like Faster-RCNN Ren et al. (2015) and YOLOv3 Redmon & Farhadi (2018). Faster-RCNN is a two stage detector, where foreground proposals are first generated by the RPN (Region Proposal Network). These proposals are then classified and localized with a set of fully connected layers. YOLOv3 is a single stage detector, in which the bounding boxes are directly predicted with three independent tasks, bounding box localization, confidence of objectness, and classification. Both of the methods regress the bounding boxes from a set of anchor boxes, so they are called anchor-based detectors.

Unsupervised Domain Adaptation for Object Detection is different from SSL for Object Detection in two aspects. First, in the problem of DA, the labeled data and unlabeled data have obvious appearance dissimilarity. The main purpose of DA is to learn the domain irrelevant features for both domains Saito et al. (2019) Deng et al. (2020) Chen et al. (2018). While in the problem of SSL, the difference between labeled data and unlabeled data is not the concern. The purpose of SSL is to promote the performance of the model with help of unlabeled data. Second, the test set of DA is in the target domain. In SSL, we are concerned about the base data set. Due to above differences, methods for DA usually solves the problem of domain shift by decreasing the feature distribution among two domains Volpi et al. (2018)Pinheiro (2018). On the contrary, SSL methods promotes the performance of the original model by refining the pseudo-labels Cascante-Bonilla et al. (2020) for self-training or training with carefully designed data augmentation methods Verma et al. (2019)Arazo et al. (2019).

3 METHODOLOGY

3.1 PROBLEM FORMULATION

Given two sets of images $X = \{x_i; i = 1, 2, ..., M\}$ and $U = \{u_j; j = 1, 2, ..., N\}$, where X is the labeled dataset and U is the unlabeled dataset. $\{x_i, y_i\}$ corresponds to one pair of labeled training sample, where y_i is the labeled bounding boxes. In the scenario of self-training for SSL, pseudo-labels for U are generated by the teacher model. Assume that the teacher generates pseudo-labels $\hat{y}_j^T = T(u_j; \theta_T)$, and the student model has the predictions of the unlabeled data $\hat{y}_j^S = S(u_j; \theta_S)$. Unsupervised loss is then:

$$l_u = \sum_{u_j \in U} l(\hat{y}_j^T, \hat{y}_j^S) = \sum_{u_j \in U} l(T(u_j; \theta_T), S(u_j; \theta_S)) \tag{1}$$

where $l(\cdot)$ is the standard supervised loss function of the object detector. As stated in Sohn et al. (2020); Jeong et al. (2019), not all of the pseudo-labeled samples are suitable for unsupervised training. Low confident pseudo-labeled samples contain noises and false positive samples which is harmful for the student. Traditional method to get high quality training samples from the unlabeled data is to select samples with highly confident pseudo-label. A uniform threshold is usually used to filter out low confident pseudo-labeled samples, like 0.8.

We argue that there are two problems in this kind of rough division of pseudo-labeled samples, for two reasons. First, confidence score predicted by the detector is not the only indicator of the pseudo-

labeled samples. Confidence score itself is only one point view to look at the sample. There are many other ways to check the quality of a sample, for instance, resistance to image noise, similarity to other samples of the same class, consistency during training. Each aspect of the sample reveals only a small part of it. If we come up with a way to find out all the point views and combine all the information together, we get the most accurate judgement on the quality of the sample. Second, uniform threshold is too rough, which does not take category variation into consideration. Due to data distribution difference among categories, the threshold should also be decided separately. To solve above two problems, we propose Score Ensemble (SE) and Adaptive Threshold (AT) to mine high quality pseudo-labeled samples for semi-supervised training.

3.2 SCORE ENSEMBLE

Confidence score Confidence score is the most popular indicator used to mine high quality pseudo-labeled samples. In Yolov3, confidence score is the combination of the object-ness score which reveals the probability of a bounding box to be foreground and the classification score which reveals the probability that the content in the bounding box belongs to which class (Eq. 2). In Faster-RCNN, the box-head classifies the RPN generated proposals and outputs the confidence scores (Eq. 3). In object detection tasks, confidence score offers the model a basic ability to distinguish true positive samples (TP) from false positive samples (FP).

$$C_i^{FRCNN} = \arg\min_{p_i} L_{cls}(p_i, p_i^*), \tag{2}$$

$$C_i^{YOLO} = p_i * IOU_{pred}^{truth} \tag{3}$$

p_i^* is the ground-truth label. In previous work, the pseudo-labeled samples are separated into TP and FP with the uniform threshold for all categories (for instance, 0.9 in Sohn et al. (2020)). Differently, we use adaptive thresholds for each categories which will be introduced later.

Temporal Consistency Score We find that with the process of training, the predictions of the model changes over time. Mostly, the false predictions are less stable than the correct predictions. Following this discovery and inspired by Laine & Aila (2016), we propose the temporal consistency score as another indicator of the pseudo-labeled sample's quality. We record the model during its supervised training process. For the jth epoch, the model is saved as Π_j. The predictions of Π_i on the whole unlabeled sample u_i is written as $\hat{u}_i^j$. For sample u_i, temporal consistency score over multiple models $\{\Pi_j\}$ is calculated as follows:

$$TC_i = \frac{\sum_j C_i^j}{Var[\boldsymbol{C}_i]}, \tag{4}$$

where C_i^j is the confidence score of sample u_i predicted by model Π_j. In Wang et al. (2018), multi-model prediction consistency is also used for object detection training sample mining. Different from Wang et al. (2018), our temporal consistency does not require additional cut an paste. We are more concerned the agreement degree that multiple models have on each sample, other than the categorial prediction distribution.

Augmentation Consistency Score It has been proven effective in images classification Miyato et al. (2018)Laine & Aila (2016)Tarvainen & Valpola (2017) that consistency regularization with different augmentations improves the semi-supervised learning. Instead of directly adding a regularization term, we generate the augmentation consistency scores for the pseudo-labeled samples. Thus, we create another indicator to help distinguishing TPs and FPs. Since some of the samples are not tolerant to the augmentation disturbing, and may produce different predictions, augmentation consistency scores reflect the degree of resistance to different augmentations. Assume there are K types of augmentation written as $f_k(), k = 1, 2, ..., K$. Augmentation consistency score for sample u_i is calculated as follows:

$$AC_i = \frac{\sum_k f_k(C_i)}{Var[\boldsymbol{f}(C_i)]} \tag{5}$$

In this work, we use data augmentation strategies like images flip, resize and color transform.

Having gotten scores for each sample u_i, training samples are selected from the pseudo-labeled samples by ensembling the scores. Each time we filter our samples by setting a unique threshold for

Algorithm 1 SEAT for semi-supervised object detection

```
Train the model Π on labeled data X.
Generate pseudo-labels Y^T = ŷ_j^T, j = 1, 2, ..., N for the samples in the unlabeled data set U using the model Π.
Calculate the adaptive thresholds for each categories following Eq. 6.
for sample s_i do
    if s_i ∈ U then
        Get pseudo-label y_i of s_i. Check the adaptive threshold t̂_k of sample s_i's category.
        if scores_i > t̂_k then
            Unsupervised loss calculated by Eq. 1
        end if
    else
        Supervised loss
    end if
end for
```

each of the scores, the overall quality of the training samples is improved. To combine the ability of all scores, we ensemble the scores in a boosting way. The samples are first filtered by the confidence score, then filtered by the temporal consistency score, and finally by the augmentation consistency scores. The thresholds for each scores are set adaptively instead of static for all categories.

3.3 Adaptive Thresholds for Each Category

Since the decision boundary is the key for training data selection in SSL, a more accurate thresholding method is essential other than the uniform threshold. With the observation that the data distribution of pseudo-labeled samples for each class are different, we assume that the ideal thresholds for each class should also be different. One challenge is how to figure out the ideal thresholds automatically according to the data distribution of each classes.

Since the detectors are usually pre-trained on labeled data, the confidence score is regressed with the supervision of GT labels. The distribution of the confidence scores on the training data tend to gather around 0 or 1. This phenomenon can be explained as follows: when the training process converges, the confidence score prediction loss is small, so they must be around 0 or 1, otherwise the loss will become large. When we calculate the distribution of the confidence scores on the pseudo-labeled data, however, they may not gather around 0 or 1 as close as on the labeled data. It is because the detector has not been trained on the unlabeled data set in supervised manner. We call this the score distribution shift phenomenon. The score distribution shift can be defined as the entropy of the confidence score distribution. Intuitively, when the larger the score shift is, the more noise the pseudo-labeled samples have. The threshold of the category with large score distribution shift should be set higher than that with smaller score distribution shift. We adaptively calculate the threshold for each class with different confidence score shift as follows:

$$\hat{t}_k = \arg\min_t S_k(t) - \gamma S_k(T), \tag{6}$$

where k is the index of the categories. $S_k(t)$ represents the total number of samples with confidence scores larger than t. γ controls the percentage of samples used to determine the variational threshold. Pseudo-labeled samples with confidence scores higher than T are seen as true positive (TP) samples.

4 Experiments

We evaluate our method on public datasets MS-COCO and Pascal VOC, the most popular public datasets for object detection. Following Sohn et al. (2020) and Tang et al. (2020), we conduct experiments based on three different settings. The MS-COCO data set contains 118k labeled images and 123k unlabeled images. Following Tang et al. (2020), we split the labeled images into two parts. 1%, 2%, 5%, and 10% images are randomly sampled from the labeled images and they are used as the labeled set. The other 99%, 98%, 95%, and 90% images are used as the unlabeled set. Following Sohn et al. (2020) and Tang et al. (2020), we use the 118k labeled images as the labeled set and the 123k unlabeled images as the unlabeled set. The semi-supervised object detection methods are

Table 1: Comparison on MS-COCO with COCO-unlabel. Experiment is done on Faster-RCNN Resnet-50. Evaluated with IOU=0.5.

Detector	Methods	COCO-train	COCO-unlabel	AP	$AP^{0.5}$
Faster-RCNN	Supervised	√	-	37.3	59.2
	Proposal	√	√	38.4	59.7
	CSD$^{\diamond}$	√	√	40.2	60.5
	STAC$^{\diamond}$	√	√	39.8	60.2
	SEAT (ours)	√	√	38.5	59.9
	CSD + SEAT (ours)	√	√	**41.0**	**61.3**
YOLO-v3	Supervised	√	-	27.5	50.1
	CSD$^{\diamond}$	√	√	28.7	52.4
	STAC$^{\diamond}$	√	√	29.0	52.6
	SEAT (ours)	√	√	28.5	51.8
	CSD + SEAT (ours)	√	√	**30.3**	**52.9**

Table 2: Comparison on MS-COCO. Experiment is done on Faster-RCNN.

Methods	1% COCO	2% COCO	5% COCO	10% COCO
Supervised	9.2	12.2	17.6	21.1
CSD + SEAT (ours)	13.8	16.9	21.5	26.4

evaluated with mAP over 80 classes on the MS-COCO test set. Following the standard MS-COCO evaluation method, the mean mAP over IOUs of 0.5 to 0.95 (0.05 gap) is reported.

In Pascal VOC, there are two sets VOC07 and VOC12 which do not contain same images. We use the VOC07 as the labeled set, and VOC12 as the unlabeled set. The testset of VOC07 is used for evaluation. The evaluation metric is the mAP over 20 classes over IOU of 0.5.

4.1 Implementation Details

We implement SEAT both on single stage and two stage detectors (Yolov3 and Faster RCNN). For Yolov3, we use Darknet19 as the backbone. For Faster-RCNN, we use ResNet-50 as the backbone. We use MMDetection to construct the networks and design the training process. The models are pretrained on ImageNet.

There are two ways to generate pseudo labels for the unlabeled images, online and offline. Online generation refers to predict pseudo labels while training. Offline generation refers to predict pseudo labels altogether and store the labels in the file, which can be loaded into the memory for training later on. To save calculation resources, we conduct the offline generation in all the experiments for LWDT. We train object detectors with 2 NVIDIA Tesla V100 in 30 epochs with batch size of 8, in which 4 labeled and 4 unlabeled images are sampled randomly from the training set. We use SGD with initial learning rate of 0.0005 to optimize the network. The learning rate is stepped after 15 and 25 epochs.

4.2 Results

The main purpose of SSL is to promote the performance of the model trained on the labeled data. We compare the results of the model trained with SEAT in SSL manner with the model trained on labeled data in SL manner. In additions, we compare with consistency based semi-supervised object detection methods Sohn et al. (2020)Jeong et al. (2019)Tang et al. (2020).

Table 3: Results on PASCAL VOC2007. Experiment is done on Faster-RCNN. Score 1 is confidence score. Score 2 is temporal consistency score. Score 3 is augmentation consistency score.

Methods	Labeled	Unlabeled	Score 1	Score 2	Score 3	AT	AP	$AP^{0.5}$
Supervised	VOC07	-	—-	—-	—-	—-	42.60	74.80
CSD$^{\diamondsuit}$	VOC07	VOC12	$\surd$	—-	—-	—-	43.42	78.40
STAC	VOC07	VOC12	$\surd$	—-	—-	—-	44.64	79.08
CSD + SEAT (ours)	VOC07	VOC12	$\surd$	$\surd$	—-	—-	43.50	79.32
CSD + SEAT (ours)	VOC07	VOC12	$\surd$	$\surd$	$\surd$	—-	44.06	79.20
CSD + SEAT (ours)	VOC07	VOC12	$\surd$	$\surd$	$\surd$	$\surd$	45.30	80.05
CSD + SEAT (ours)	VOC07	VOC12 + COCO	$\surd$	$\surd$	$\surd$	$\surd$	**45.80**	**80.85**

Table 4: Results on PASCAL VOC2007 to study the influence of threshold. Experiment is done on Faster-RCNN.

Methods	Labeled Data	Unlabeled Data	Threshold	AP	$AP^{0.5}$
CSD + SE (ours)	VOC07	VOC12	0.4	42.80	75.33
CSD + SE (ours)	VOC07	VOC12	0.8	43.98	78.01
CSD + SE (ours)	VOC07	VOC12	0.9	44.21	79.45
CSD + SE + AT (ours)	VOC07	VOC12	AT	**45.30**	**80.05**

Table 1 shows the comparison between different methods on MSCOCO. Same as Sohn et al. (2020)Tang et al. (2020), the coco-train2017 data set is used as the labeled data, and the coco-unlabeled is used as the unlabeled data. We conduct experiments both on Faster-RCNN (ResNet-50) and Yolo-v3 (Darknet19). Although only applying SEAT does not produce competitive results, when combined with CSD, our method outperforms other methods. We get an improvement of 3.7 mAP on Faster-RCNN and 2.8 mAP on Yolo-v3.

Table 2 shows the results on split COCO data set. We only use a small part of the samples in COCO-train2017 as the labeled data and the rest as the unlabeled data. We randomly split the data set into 1% over 99%, 2% over 98%, 5% over 95% and 10% over 90%. For 1 and 2 % protocols, we improve 2 mAP. For 5 and 10 % protocols, we improve 4 mAP. As for the experiment on Pascal-VOC, we also outperform previous methods.

4.3 Ablation Study

Understanding Score Ensemble Table 3 shows the ablation study of scores used in our method. Confidence score used in CSD shows beneficial for SSL. The improvement over baseline is 0.8 mAP. Together with temporal consistency score, we achieve the improvement of 0.9 mAP. When ensembled with augmentation consistency score, the advantage rises to 1.4 mAP. For the above experiments, the thresholds are set to be 0.8 for all categories. After applying adaptive thresholds, we achieve 45.3 mAP and get 2.7 mAP over baseline. We also notice that more unlabeled data is helpful. When adding more unlabeled COCO data into training, the result further rises to 45.8 mAP.

In our proposed method, we only use three scores, confidence score, augmentation consistency, and temporal consistency. We have proven these scores are all beneficial for the whole system. While a problem naturally arises: what kind of scores are helpful? In our opinion, the scores should provide information from different aspects. Compared with distinguishing accuracy using a score, we are more concerned whether it can provide information other than the confidence score already has.

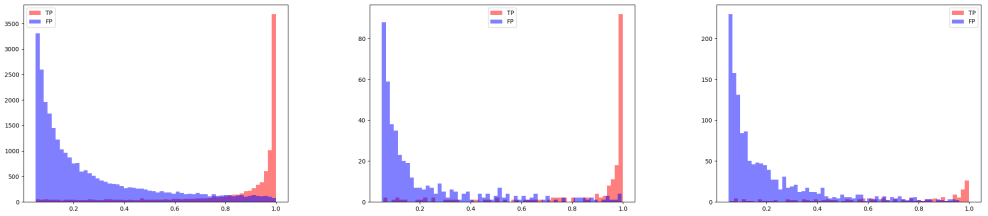

Figure 1: Data distribution of TP/FP samples on COCO-test. By changing the threshold to select pseudo-labeled samples for training, the TP/FP ratio also changes. The data distribution difference among categories has an influence on the ideal threshold to distinguish the TPs and FPs.

Understanding Adaptive Thresholds Threshold is important for SSL, because it is the simplest way to extract high quality pseudo-labeled training samples. In previous works, a static threshold is usually offered for all categories, which causes two side effects. One is that the same threshold for all the categories assumes that the data distributions of each categories are the same, which is not the case. This will cause the mis-split of the pseudo-labeled samples. If the threshold is set too high for one category, the training data is too little. If the threshold is set too low for one category, the noises in the training data are too much. Although the thresholds for each categories can be manually set, it requires plenty of time to figure out all the hyper parameters. Our adaptive thresholds solves above problems by adaptively select suitable thresholds for each class according to the data distribution of the unlabeled data. This method does not require heavy work to adjust parameters, while improves the overall quality of the pseudo-labeled samples.

Apart from the proposed adaptive thresholds by calculating the data distribution on unlabeled data, there is another way to figure out a set of thresholds. In the case where the data distribution on labeled data has little difference to that on unlabeled data, it is fine to calculate the thresholds on labeled data. Then we know whether each pseudo-labeled samples are positive or negative, and can determine more precise adaptive thresholds for each categories. But the assumption of same data distribution on labeled and unlabeled data is not always true. The adaptive thresholds calculated on labeled data has not much difference on our proposed adaptive thresholds on unlabeled data.

Training Samples: Quantity and Quality To select high quality training data from the pseudo-labeled samples, a high threshold is usually used in SSL. The higher the threshold, the purer is the pseudo-labeled samples. But the threshold cannot be set as high as possible, because number of effective samples decreases as the threshold rises. So the quantity and quality of the training data is contradictory.

5 Conclusion

The quality of pseudo-labels matter a lot for semi-supervised object detection. In this paper, we investigate the relationship between the pseudo-label quality and SSL. Following the discovery, we design a general framework for semi-supervised object detection, SEAT ((**S**core **E**nsemble with **A**daptive **T**hreshold)). In the framework, the high confidence pseudo-labels are refined for self-training. Several indicators are used to distinguish true positive samples (TP) from false negative samples (FP). These indicators convey information of the quality of a sample from different aspects. They provide a more comprehensive description to the quality of each sample. In this paper, apart from confidence score as the indicator of the sample's quality, we also introduce the scores of temporal consistency and augmentation consistency. To cope with the data distribution difference among categories, the adaptive threshold strategy is used to automatically determine the sample mining threshold for each category. This framework is compatible with consistency-based SSL methods. Extensive results on PASCAL-VOC and MSCOCO show the flexibility and efficiency of this framework. In future work, more reliable indicators and better combination methods are needed to improve SEAT.

REFERENCES

Eric Arazo, Diego Ortego, Paul Albert, Noel E O'Connor, and Kevin McGuinness. Pseudo-labeling and confirmation bias in deep semi-supervised learning. *arXiv preprint arXiv:1908.02983*, 2019.

David Berthelot, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Kihyuk Sohn, Han Zhang, and Colin Raffel. Remixmatch: Semi-supervised learning with distribution matching and augmentation anchoring. In *International Conference on Learning Representations*, 2019a.

David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. In *Advances in Neural Information Processing Systems*, 2019b.

Paola Cascante-Bonilla, Fuwen Tan, Yanjun Qi, and Vicente Ordonez. Curriculum labeling: Self-paced pseudo-labeling for semi-supervised learning. *arXiv preprint arXiv:2001.06001*, 2020.

Yuhua Chen, Wen Li, Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Domain adaptive faster r-cnn for object detection in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018.

Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical data augmentation with no separate search. *arXiv preprint arXiv:1909.13719*, 2019.

Jinhong Deng, Wen Li, Yuhua Chen, and Lixin Duan. Unbiased mean teacher for cross domain object detection. *arXiv preprint arXiv:2003.00707*, 2020.

Jiyang Gao, Jiang Wang, Shengyang Dai, Li-Jia Li, and Ram Nevatia. Note-rcnn: Noise tolerant ensemble rcnn for semi-supervised object detection. In *Proceedings of the IEEE international conference on computer vision*, 2019.

Jisoo Jeong, Seungeui Lee, Jeesoo Kim, and Nojun Kwak. Consistency-based semi-supervised learning for object detection. In *Advances in neural information processing systems*, 2019.

Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. *arXiv preprint arXiv:1610.02242*, 2016.

Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 2018.

Pedro O Pinheiro. Unsupervised domain adaptation with similarity learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, 2015.

Kuniaki Saito, Yoshitaka Ushiku, Tatsuya Harada, and Kate Saenko. Strong-weak distribution alignment for adaptive object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

Kihyuk Sohn, Zizhao Zhang, Chun-Liang Li, Han Zhang, Chen-Yu Lee, and Tomas Pfister. A simple semi-supervised learning framework for object detection. *arXiv preprint arXiv:2005.04757*, 2020.

Peng Tang, Chetan Ramaiah, Ran Xu, and Caiming Xiong. Proposal learning for semi-supervised object detection. *arXiv preprint arXiv:2001.05086*, 2020.

Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in neural information processing systems*, 2017.

Vikas Verma, Alex Lamb, Juho Kannala, Yoshua Bengio, and David Lopez-Paz. Interpolation consistency training for semi-supervised learning. *arXiv preprint arXiv:1903.03825*, 2019.

Riccardo Volpi, Pietro Morerio, Silvio Savarese, and Vittorio Murino. Adversarial feature augmentation for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

Keze Wang, Xiaopeng Yan, Dongyu Zhang, Lei Zhang, and Liang Lin. Towards human-machine cooperation: Self-supervised sample mining for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1605–1613, 2018.

Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

A APPENDIX

You may include other additional sections here.