# DTG-SSOD: Dense Teacher Guidance for Semi-Supervised Object Detection

**Anonymous Author(s)**

## Abstract

The Mean-Teacher (MT) scheme is widely adopted in semi-supervised object detection (SSOD). In MT, the *sparse* pseudo labels, offered by the final predictions of the teacher (e.g., after Non Maximum Suppression (NMS) post-processing), are adopted for the *dense* supervision for the student via hand-crafted label assignment. However, the "sparse-to-dense" paradigm complicates the pipeline of SSOD, and simultaneously neglects the powerful direct, dense teacher supervision. In this paper, we attempt to directly leverage the dense guidance of teacher to supervise student training, i.e., the "dense-to-dense" paradigm. Specifically, we propose the Inverse NMS Clustering (INC) and Rank Matching (RM) to instantiate the dense supervision, without the widely used, conventional sparse pseudo labels. INC leads the student to group candidate boxes into clusters in NMS as the teacher does, which is implemented by learning grouping information revealed in NMS procedure of the teacher. After obtaining the same grouping scheme as the teacher via INC, the student further imitates the rank distribution of the teacher over clustered candidates through Rank Matching. With the proposed INC and RM, we integrate Dense Teacher Guidance into Semi-Supervised Object Detection (termed "DTG-SSOD"), successfully abandoning sparse pseudo labels and enabling more informative learning on unlabeled data. On COCO benchmark, our DTG-SSOD achieves state-of-the-art performance under various labelling ratios. For example, under 10% labelling ratio, DTG-SSOD improves the supervised baseline from 26.9 to 35.9 mAP, outperforming the previous best method Soft Teacher by 1.9 points.

## 1 Introduction

Thanks to plenty of annotated data, deep learning based computer vision has achieved great success in the past few years [1–3]. However, labeling accurate annotations is usually labour-consuming and expensive, especially for dense prediction tasks [4–7], such as object detection, where multiple bounding boxes with their corresponding category labels need to be annotated extensively for each image. Some works attempt to leverage easily accessible unlabeled data to promote object detection in a semi-supervised manner [8–11]. The model is first initialized by learning on the labeled data. Then it is adopted for generating pseudo labels (i.e., pseudo bounding boxes and category labels) on the unlabeled data for training.

Recently, advanced Semi-Supervised Object Detection (SSOD) methods follow the Mean-Teacher paradigm [13], where a stronger teacher model is built from the student model via Exponential Moving Average (EMA) to produce supervision signals over unlabeled data. Nevertheless, all of the previous methods [8–10, 12, 11, 14] adopt *sparse* pseudo labels after the post-processing (e.g., NMS and score filtering) as the supervision, passed from the teacher to student. To guarantee the precision of pseudo labels, a high score threshold (e.g., 0.9) is usually selected to filter coarse detection boxes and only a few pseudo boxes could survive. Through the hand-crafted label assignment strategy [15–17], the detector transfers *sparse* pseudo labels into *dense* supervision for student training, which is termed the "sparse-to-dense" paradigm. Such sparse pseudo labels neglect some important properties of object

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

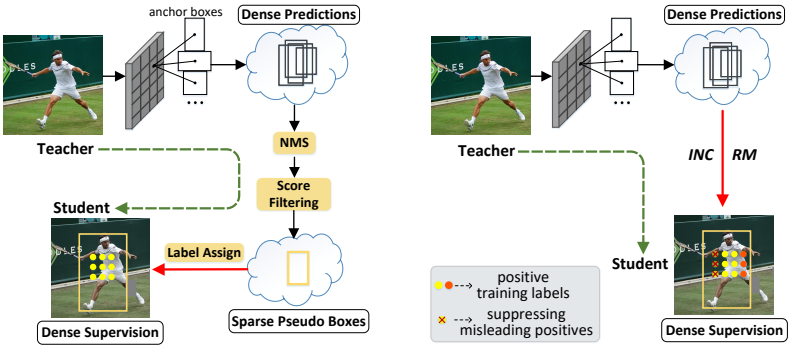


Figure 1: Comparison of two paradigms: (a) previous methods [8, 9, 12] perform NMS and score filtering on teacher's dense predictions to obtain *sparse* pseudo boxes, which are further converted to *dense* supervision of the student via the label assignment; (b) our DTG-SSOD directly converts teacher's *dense* predictions into the *dense* guidance for the student, by the proposed INC and RM.

detection (e.g., dense predictions, contextual relations between neighbouring samples), and lack of informative and dense teacher guidance, leading to limited benefits for learning on the unlabeled data. In Fig. 1, we illustrate the "sparse-to-dense" paradigm, where the sparse pseudo labels originated from the dense predictions of teacher are adopted for the dense supervision of student. This process involves many handcrafted components (e.g., NMS, score thresholding, and label assignment), which inevitably introduce accumulated noise/bias to supervisory signals for student. Therefore, it is more intuitive to adopt the "dense-to-dense" paradigm, i.e., directly employing *dense* predictions of teacher to guide the training of student, whilst abandoning the intermediate operations.

In this work, to implement the "dense-to-dense" paradigm, we propose to enforce the student to mimic the behavior of the teacher model. Specifically, the behaviour is motivated from the procedure how dense samples perform NMS, an essential post-processing scheme for popular detectors. To regularize the consistency on NMS behaviour between the teacher and student, we introduce Inverse NMS Clustering (INC) and Rank Matching (RM). INC leads the student to group the candidate boxes into clusters in NMS as the teacher does. In the process of NMS, candidates are grouped into multiples clusters. Within each cluster, teacher candidates are supposed to predict the identical object, which will assign training labels to the corresponding student samples in turn. Different from the previous sparse-to-dense paradigm, where sparse pseudo labels are converted to student training labels via the hand-crafted label assignment, our dense-to-dense paradigm assigns dense training labels based on grouping information by reversing the NMS process of the teacher. Based on INC where the student obtains the same grouping scheme of NMS with the teacher, we further introduce Rank Matching to align the score rank over the clustered candidates between teacher and student. NMS suppresses the duplicated detection and reserves only one box for each cluster through the rank of clustered candidates. Therefore, the rank of the samples within each cluster could contain rich relationship learned by the teacher and it can behave as informative dense supervision which guides the student to reserve as the same candidates as the teacher during NMS.

Through the proposed INC and RM, the teacher model provides not only fine-grained and appropriate training labels, but also informative rank knowledge, realizing Dense Teacher Guidance for SSOD (DTG-SSOD). To validate the effectiveness of our method, we conduct extensive experiments on COCO benchmark under Partially Labeled Data and Fully Labeled Data settings. Under the Partially Labeled Data, where only 1%,2%,5%, and 10% images are labeled, the proposed DTG-SSOD achieves the state-of-the-art performance, for example, under 5% and 10% labelling ratios, DTG-SSOD surpasses the previous best method Soft Teacher by +1.2 and +1.9 mAP, respectively. Under the Fully Labeled Data, where the unlabeled2017 set is used as unlabeled data, the proposed DTG-SSOD not only achieves the best performance (45.7 mAP), but also improves the learning efficiency, e.g., DTG-SSOD halves the training iterations of Soft Teacher but achieves even better performance.

## 2 Related Work

**Semi-Supervised Image Classification (SSIC)** The pseudo labeling [18–20] is a popular pipeline, where unlabeled data is labeled by the model itself, and then combined with labeled data to act as

training examples in the cycle self-training manner. In the pseudo-labeling, hard pseudo labels are usually used with a confidence-based thresholding where only images with high confidence can be retained. Recently, some improvements [21–23] on pseudo-labeling are proposed, for example, Curriculum Labeling [21] proposes that restarting model parameters before each training cycle can promote self-training; Noisy Student Training [22] adopts an equal-or-larger model with noise injected to learn the pseudo labels, which are generated in the last self-training cycle. Apart from hard pseudo labels, Noisy Student Training also tried soft pseudo labels, observing that for in-domain unlabeled images, both hard and soft pseudo labels perform well, while for out-of-domain unlabeled data, soft pseudo labels can work better. On the other hand, based on the insight that the model should be robust to input perturbations, UDA [24] designs advanced data augmentations and enforces the model to predict the similar class distributions for the differently augmented inputs. Here, soft labels (i.e., distributions) are superior in measuring the distance between different model predictions over the hard ones. As another strong method, FixMatch [25] introduces a separate weak and strong augmentation and converts model predictions into hard labels, since the labels with low entropy are believed to be beneficial in their pipeline. To conclude, previous SSIC works investigate various forms of supervisory signals (including hard and soft pseudo labels), and observe the form of supervision signals matters in semi-supervised learning, which inspires us to investigate the optimal supervision form for semi-supervised object detection.

**Semi-Supervised Object Detection (SSOD)** CSD [26] is an early attempt to construct consistency learning for horizontally flipped image pairs, where only the flip transformation can be used as the augmentation, which greatly limits its performance. STAC [27] borrows the separate weak and strong augmentation from FixMatch [25] and applies it to SSOD, obtaining promising performance. After that, [9, 10, 12, 8] abandon the conventional multi-phase training and update the teacher model from the student via EMA after each training iteration. Apart from the framework, previous works promote SSOD from two main perspectives: designing advanced data augmentation and improving the precision of pseudo labels. For the former one, Instant-Teaching [9] introduces more complex augmentations for unlabeled data, including Mixup and Mosaic; MUM [14] designs a Mix/UnMix data augmentation, which enforces the student to reconstruct unmixed feature for the mixed input images. As for the latter, through ensembling the teacher model on input images and prediction head, respectively, Humble Teacher [10] and Instant-Teaching [9] reduce confirmation bias of the pseudo labels; Soft Teacher [8] selects reliable pseudo boxes by averaging regression results from multiple jittered boxes; Rethinking Pse [11] converts the box regression to a classification task, making localization quality easy to estimate. Different from these methods, which all adopt sparse pseudo boxes with category labels as the supervisory signal, our method investigates the new form of supervision for the first time and proposes to replace sparse pseudo labels with more informative dense supervision.

# 3 Method

The mechanism of weak-strong augmentation pairs shows promising performance in SSOD. Based on it, we first build an end-to-end pseudo-labeling baseline, described in Sec. 3.1, through which we explain the task formulation of SSOD and the conventional sparse-to-dense paradigm. Then in Sec. 3.2, we introduce the proposed Inverse NMS Clustering (INC) and Rank Matching (RM), i.e., the key two components of the proposed DTG-SSOD.

## 3.1 Sparse-to-dense Paradigm

**Task Formulation.** The framework of SSOD is illustrated in Fig. 2(a). The Mean-Teacher scheme is a common practice for previous works [8, 9, 12, 10] and realizes the end-to-end training, where the teacher is built from the student via EMA after each training iteration. The teacher takes weakly augmented (e.g., flip and resize) images as input to generate pseudo labels, while the student is applied with strong augmentations (e.g., cutout, geometry transformation) for training. Strong and appropriate data augmentations play an important role, which not only increase the difficulty of student tasks and alleviate the over-confidence issue [24], but also enable the student to be invariant to various input perturbations, leading to robust representation learning [22, 25]. In this work, we borrow the data augmentations from Soft Teacher [8], and more details can be found in Appendix. Each training batch consists of both labeled data $\{x_i^l, y_i^l\}_{i=1}^{N_l}$ and unlabeled data $\{x_i^u\}_{i=1}^{N_u}$, which are mixed according to the ratio $r = \frac{N_l}{N_u}$, where $N_l$ and $N_u$ are the number of labeled and unlabeled

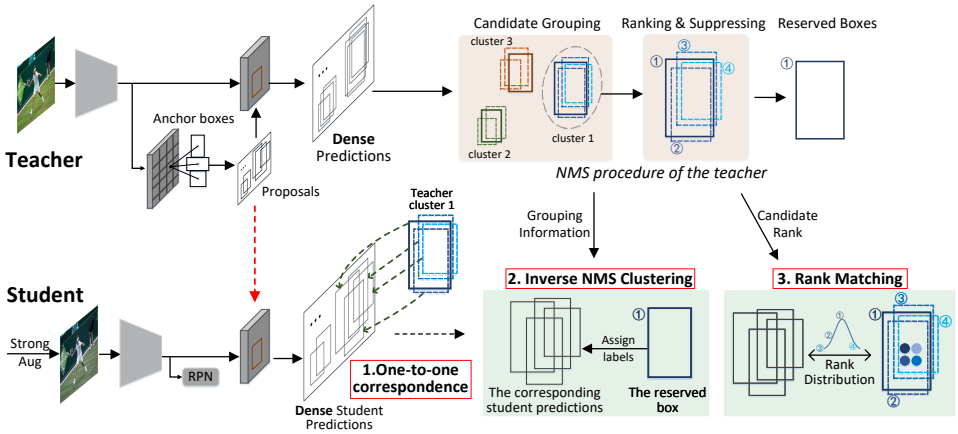


Figure 2: The pipeline of our DTG-SSOD on **unlabeled data**. The training process at R-CNN stage is illustrated, and it can be easily applied to RPN stage with minor modifications. For dense predictions of the teacher, NMS is performed, consisting of candidate grouping, ranking, and suppressing. We design three steps to implement the "dense-to-dense" paradigm. Firstly, one-to-one correspondence between samples of teacher and student is identified. Next, INC assigns training labels for the student based on grouping information revealed in the teacher's NMS; At last, RM regularizes consistency in candidate rank between two models. These three steps are described successively in Sec. 3.2

images in each batch. The overall loss function of SSOD is formulated as:

$$\mathcal{L} = \mathcal{L}^l + \alpha \mathcal{L}^u, \tag{1}$$

where $\alpha$ controls the contribution of losses on labeled data $\mathcal{L}^l$ and unlabeled data $\mathcal{L}^u$.

**Sparse-to-dense Baseline.** All previous SSOD approaches [8, 9, 12, 10] are based on the sparse-to-dense mechanism, where sparse pseudo boxes with category labels are generated to behave as ground-truths for the student training. It comes with confidence-based thresholding where only pseudo labels with high confidence (e.g., larger than 0.9) are retained. This makes foreground supervision on unlabeled data much sparser than that on labeled data, therefore, the class imbalance problem is amplified in SSOD and impedes the detector training severely. To alleviate this issue, we borrow some strengths of previous works: Soft Teacher [8] sets the mixing ratio $r$ to 1/4 to sample more unlabeled data in each training batch, which makes the number of foreground samples on unlabeled data approach that on labeled data; Unbiased Teacher [12] replaces the cross-entropy loss with the Focal loss [15], reducing the gradient contribution from easy examples. Both of these two improvements, i.e., the appropriate mixing ratio $r$ (1/4) and Focal loss, are adopted in the sparse-to-dense baseline and our dense-to-dense DTG method. Because the teacher only provides *sparse* pseudo labels, which are further converted into the *dense* supervision for the student training, these methods are termed "sparse-to-dense" paradigm. In theory, our SSOD method is independent of the detection framework and can be applicable to both one-stage and two-stage detectors. For a fair comparison with previous works, we use the Faster RCNN [16] as the default detection framework.

### 3.2 Our method: Dense-to-dense paradigm

Pseudo labels in SSOD contain both bounding boxes and category labels. Considering that classification scores are not strongly correlated with localization quality [28, 3, 29, 30], although confidence-based thresholding can filter out many false positives, the localization quality of pseudo boxes is not guaranteed. Poorly-localized pseudo boxes will mislead the label assignment and confuse the decision boundary. Therefore, previous methods [11, 8] design two separate sets of indicators to measure the confidence of categories and box locations, which complicates the SSOD pipeline. Moreover, the sparse-to-dense paradigm neglects some important properties of object detection and lacks of powerful direct, dense teacher guidance. Motivated by these, we propose to directly employ teacher's dense predictions to supervise the student's training, termed "dense-to-dense" paradigm, abandoning the explicit sparse pseudo boxes.

Dense predictions widely exist in both RPN and RCNN stages. Specifically, in the RPN stage, multiple anchors are usually tiled on each feature map grid [16, 15]; while in the RCNN stage, taking

2,000 proposals as training samples is a common practice [16]. Compared against the sparse pseudo labels, dense predictions of the teacher contain rich knowledge, and we attempt to pass it to the student by enforcing the student to mimic the dense behaviour of the teacher. Specifically, we analyze how teacher's dense samples perform to eliminate redundancy in the process of Non Maximum Suppression (NMS). Based on the analysis, we design Inverse NMS Clustering and Rank Matching to regularize consistency on the NMS procedure between the teacher and student, from which the student can obtain precise training labels and informative supervision signals.

However, implementing dense mimic is not trivial. Due to the separate weak-strong augmentations, one-to-one correspondence between the samples of the teacher and student can not be easily identified. We design a simple yet effective matching strategy to address this. For the R-CNN stage, we directly copy the teacher proposals to the student RCNN, realizing one-to-one correspondence. For the RPN stage, since the feature map sizes and pre-set anchors differ between teacher and student, we first transfer the teacher's anchors to the space of student through box transformation. Then we assign the teacher's anchors to the corresponding FPN level based on the anchor box scale. The index of target FPN level $m_i$ can be calculated by: $m_i = \log_2 \frac{area_i}{S}$, where $area_i$ refers to the area of the teacher anchor box $t_i$, and $S$ is a hyper-parameter and set to 32. Once the target FPN level $m_i$ is obtained, we calculate the Intersection-over-Unions (IoUs) between the teacher anchor $t_i$ and all student anchors $\{s_j\}^{j \in m_i}$ belonging to the FPN level $m_i$, and set the student anchor with the largest IoU as the corresponding one: $s_i = \arg\max IoU(t_i, s_j), j \in m_i$. Through the proposed matching strategy, one-to-one correspondence between training samples of the teacher and student is identified, making dense behaviour imitation possible.

**Inverse NMS Clustering**. We first revisit the mechanism of NMS by taking RCNN stage as the example. Before NMS, predicted bounding box offsets are applied on proposals to derive regressed boxes. Then a pre-defined score threshold $\tau$ filters out clear background detection boxes. For the remaining detection boxes (or candidates), NMS conducts clustering on them based on their categories and locations. As a result, candidates are grouped into multiple clusters and in each cluster, candidates are supposed to predict the same object. At last, only one candidate with the highest confidence can be retained and others will be suppressed. The process is illustrated in Fig. 2(b).

To regularize consistency on the NMS behaviour of the teacher and student, we enforce the student to perform the same candidate grouping as the teacher. In the NMS procedure of the teacher, candidates will be grouped into multiple clusters, and the cluster indexed by $j$ can be defined as $C_j^t = \{b_i^t, c_i^t\}_{i=1}^{N_j}$, where $b$ and $c$ refer to the bounding box and category, respectively, and $N_j$ is the number of candidates belonging to $C_j$. In each cluster of the teacher (taking $C_j^t = \{b_i^t, c_i^t\}_{i=1}^{N_j}$ as an example), the candidates own the same category $c$ (i.e., $c_1^t = \cdots = c_i^t = c$), and highly-overlapped boxes $\{b_i\}_{i=1}^{N_j}$, thus they are supposed to predict the identical object. Here, we take the finally reserved box $\{b_k^t, c\}$ to represent the detection target of $C_j$. Then the target $\{b_k^t, c\}$ will inversely assign training labels to the corresponding student samples $\{b_i^s, c_i^s\}_{i=1}^{N_j}$. To be specific, the category label $c$ is employed to supervise the classification predictions of the student: $\{c_i^s\}_{i=1}^{N_j}$, and the classification loss can be formulated as:

$$\mathcal{L}_{cls}^u = \frac{\sum_{i=1}^{N_j} f_{cls}(c_i^s, c)}{N_j}, \tag{2}$$

where $f_{cls}$ refers to the Focal loss [15]. Besides, for those teacher samples, whose scores are lower than the NMS threshold $\tau$, they are clear background ones and excluded from the NMS, thus we assign background labels to their corresponding student samples. For the regression task, we directly take the reserved box $b_k^t$ as the regression target for $\{b_i^s\}_{i=1}^{N_j}$. We describe the regression loss as:

$$\mathcal{L}_{reg}^u = \frac{\sum_{i=1}^{N_j} f_{reg}(b_i, b_k^{t'})}{N_j}, \tag{3}$$

where $b_k^{t'}$ is obtained by transforming the $b_k^t$ to the student space; $f_{reg}$ refers to regression loss, e.g., Smooth L1 Loss [16]. There is a clear difference between our method and the traditional sparse-to-dense paradigm. Our method calculates classification and regression labels according to the grouping information revealed in the NMS procedure of the teacher, whilst the sparse-to-dense paradigm employs some hand-crafted label assignment strategies [17, 15, 16] to convert sparse pseudo labels into student's dense training labels, which loses rich teacher guidance.

**Rank Matching.** Through the Inverse NMS Clustering, the student can obtain the same grouping scheme as the teacher. Based on it, we further introduce Rank Matching to align the score rank over clustered candidates between the teacher and student. In each NMS cluster, the candidate boxes are ranked according to the classification scores, then only one candidate can be retained regarding the candidate rank. Therefore, the rank of candidates could contain rich relationship information. In object detection, the candidate rank is validated to be a key element and can affect detection performance much [28, 3, 29, 31]. For example, GFL [28] proposes that taking IoUs between predicted boxes and ground-truths as classification targets contributes to a more precise rank. Different from these methods, our Rank Matching models the score rank over clustered candidates to align the NMS behaviour of the teacher and student. Exactly, we model the score distribution of the teacher and student over the cluster $C_j = \{p_i\}_{i=1}^{N_j}$ as:

$$\mathcal{D}(p_i^t) = \frac{exp(p_i^t/T)}{\sum_{k=1}^{N_j} exp(p_k^t/T)}, \;\; D(p_i^s) = \frac{exp(p_i^s/T)}{\sum_{k=1}^{N_j} exp(p_k^s/T)}, \tag{4}$$

where $p_i$ indicates the logits from the target category of the $i$th candidate, and $T$ is the temperature coefficient. The Kullback-Leibler (KL) divergence is adopted to optimize the distance between $\mathcal{D}(p_i^t)$ and $\mathcal{D}(p_i^s)$:

$$\mathcal{L}_{RM} = -\sum_{i=1}^{N_j} D(p_i^t) \log \frac{D(p_i^s)}{D(p_i^t)}. \tag{5}$$

Then $\mathcal{L}_{RM}$ is averaged over all NMS clusters. Rank Matching not only provides relationship information but also enables the student to reserve the identical candidates in each NMS cluster as the teacher. The overall loss function on unlabeled data is defined as:

$$\mathcal{L}^u = \mathcal{L}^u_{cls} + \mathcal{L}^u_{reg} + \beta \mathcal{L}^u_{RM}, \tag{6}$$

where $\beta$ controls contribution of rank matching loss.

# 4 Experiments

## 4.1 Dataset and Evaluation Metric.

We benchmark our proposed method on the challenging dataset, MS COCO [32]. Following the convention [27, 8, 12, 9], two training settings are provided, namely **Partially Labeled Data** and **Fully Labeled Data** settings, which validate the method with limited and abundant labeled data, respectively. The val2017 set with 5k images is used as the validation set. We use $AP_{50:97}$ (denoted as mAP) as the evaluation metric. We describe two training settings as follows:

**Partially Labeled Data.** The train2017 set, consisting of 118k labeled images, is used as the training dataset, from which, we randomly sample 1%,2%,5%, and 10% images as labeled data, and set the remaining unsampled images as unlabeled data. Following the practice of previous methods [27, 8, 12], for each labelling ratio, 5 different folds are provided and the final result is the average of these 5 folds.

**Fully Labeled Data.** Here, the train2017 set is employed as the labeled data, and the additional unlabeled2017 set, consisting of 123k unlabeled images, is used as unlabeled data. This setting is to validate SSOD methods can still promote the detector even when labeled data is abundant.

## 4.2 Implementation Details

Faster RCNN [16] with FPN [35] performs as default detection framework in this paper, and ResNet-50 [36] is adopted as the backbone. Moreover, we use SGD as the optimizer. At the beginning of NMS, a score threshold $\tau$ is leveraged to filter clear background samples. In this work, we set $\tau$ to a relatively high value to exclude noisy teacher samples, whose behaviour is ambiguous and hardly provides helpful guidance. Exactly, we set $\tau$ to 0.9 in NMS of the RPN stage, and 0.45 in NMS of the R-CNN stage, empirically. To stabilize the training, we also warm up the learning on unlabeled data for the first 4k iterations [12]. The training schedules for two training settings are a bit different:

**Partially Labeled Data.** The model is trained for 180k iterations on 8 V100 GPUs with an initial learning rate of 0.01, which is then divided by 10 at 120k iteration and again at 160k iteration.

Table 1: Comparison with state-of-the-art methods on COCO benchmark. All results are reported on val2017 set. Under the **Partially Labeled Data** setting, results are the average of all five folds and numbers behind $\pm$ indicate the standard deviation. Under the **Fully Labeled Data** setting, numbers in font of the arrow indicate the supervised baseline.

| Methods | Partially Labeled Data | | | | Fully Labeled Data |
|---|---|---|---|---|---|
| | 1% | 2% | 5% | 10% | |
| Supervised Baseline | 12.20±0.29 | 16.53±0.12 | 21.17±0.17 | 26.90±0.08 | 40.9 |
| STAC [27] | 13.97±0.35 | 18.25±0.25 | 24.38±0.12 | 28.64±0.21 | 39.5 $\xrightarrow{-0.3}$ 39.2 |
| Humble Teacher [10] | 16.96±0.35 | 21.74±0.24 | 27.70±0.15 | 31.61±0.28 | 37.6 $\xrightarrow{+4.8}$ 42.4 |
| ISMT [33] | 18.88±0.74 | 22.43±0.56 | 26.37±0.24 | 30.53±0.52 | 37.8 $\xrightarrow{+1.8}$ 39.6 |
| Instant-Teaching [9] | 18.05±0.15 | 22.45±0.15 | 26.75±0.05 | 30.40±0.05 | 37.6 $\xrightarrow{+2.6}$ 40.2 |
| Unbiased Teacher [12] | 20.75±0.12 | 24.30±0.07 | 28.27±0.11 | 31.50±0.10 | 40.2 $\xrightarrow{+1.1}$ 41.3 |
| SED [34] | - | - | 29.01 | 34.02 | 40.2 $\xrightarrow{+3.2}$ 43.4 |
| Rethinking Pse [11] | 19.02±0.25 | 23.34±0.18 | 28.40±0.15 | 32.23±0.14 | 41.0 $\xrightarrow{+2.3}$ 43.3 |
| MUM [14] | **21.88**±0.12 | 24.84±0.10 | 28.52±0.09 | 31.87±0.30 | 37.6 $\xrightarrow{+4.5}$ 42.1 |
| Soft Teacher [8] | 20.46±0.39 | - | 30.74±0.08 | 34.04±0.14 | 40.9 $\xrightarrow{+3.6}$ 44.5 |
| Ours (DTG-SSOD) | 21.27±0.12 | **26.84**±0.25 | **31.90**±0.08 | **35.92**±0.26 | 40.9 $\xrightarrow{\mathbf{+4.8}}$ **45.7** |

Mini-batch per GPU is 5, with 1 labeled image and 4 unlabeled images. The loss weight of unlabeled images $\alpha$ is set to 4.0.

**Fully Labeled Data.** The model is trained on 8 GPUs with mini-batch as 8 per GPU, where 4 labeled images are combined with 4 unlabeled images. We adopt 720k iterations for training, and the initial learning rate is set to 0.01, which is divided by 10 at 480k and 680k iterations. We adopt $\alpha = 2.0$.

### 4.3 Main Results

We compare the proposed DTG-SSOD to the state-of-the-art methods on MS COCO val2017 set in Tab. 1. We first evaluate our method using the **Partially Labeled Data** setting, and experimental results demonstrate our method achieves the state-of-the-art performance under various labelling ratios. Specifically, on 5% and 10% labelling ratios, DTG-SSOD improves the supervised baseline by +10.73 and +9.02 mAP, reaching 31.90 and 35.92 mAP, and it also surpasses the previous best method Soft Teacher [8] by +1.16 and +1.92 mAP, respectively. When the labelling ratio is 2%, our DTG-SSOD yields 26.84 mAP, which is even comparable to the 26.90 mAP supervised baseline using 10% labeled data. When the labeled data is extremely scarce, i.e., 1% labelling ratio, our DTG-SSOD achieves a comparable performance to the previous best method MUM [14], 21.27 vs. 21.88 mAP. MUM [14] designs a strong data augmentation to expand data space, which mixes original image tiles to construct new images. It is worth noting that when the number of labeled data increases (2%,5%, and 10% labelling ratios), our DTD-SSOD outperforms MUM by at least +2 mAP, moreover, our method is orthogonal to these advanced data augmentations [14, 9] and can benefit from them.

As Tab. 1 shows, under the **Fully Labeled Data** setting, our DTG-SSOD surpasses previous methods by a large margin, at least 1.2 mAP. Following the practice of [8, 11], we also apply weak augmentations to labeled data and obtain a strong supervised baseline of 40.9 mAP. Even based on the such strong baseline, DTG-SSOD still attains the biggest improvements of +4.8 mAP, reaching 45.7 mAP, which validates the effectiveness of our method when the amount of labeled data is relatively large.

### 4.4 Ablation Experiments

In this section, we conduct extensive ablation studies to validate the key designs. Unless specified, all experiments adopt a single data fold of the 10% labelling ratio as the training dataset.

**Effects of Dense Teacher Guidance.** Instead of conventional pseudo boxes with category labels provided, our method offers dense supervisory signals. We first validate that Dense Teacher Guidance (DTG) is a feasible learning paradigm for SSOD in Tab. 2(a), where we apply DTG to RPN and R-CNN stages step by step. On unlabeled data, when only RPN is optimized with the DTG and R-CNN is excluded from training, it can bring 1.7 mAP gains to the supervised baseline. Furthermore, we apply DTG to both RPN and RCNN, and it yields 36.3 mAP, outperforming the baseline by +9.5 mAP. Significant improvements against the supervised baseline indicate dense teacher guidance can

Table 2: Ablation studies related to effectiveness, efficiency and comparison of two paradigms.

(a) **Effectiveness of Dense Teacher Guidance** in the RPN and RCNN stages.

| RPN | RCNN | mAP | $AP_{50}$ | $AP_{75}$ |
|---|---|---|---|---|
| | | 26.8 | 44.9 | 28.4 |
| ✓ | | 28.5 (+1.7) | 47.7 | 29.5 |
| ✓ | ✓ | **36.3 (+9.5)** | **56.4** | **38.8** |

(b) **Effectiveness of each component of DTG**. Validation is conducted in R-CNN stage.

| INC | | RM | mAP | $AP_{50}$ | $AP_{75}$ |
|---|---|---|---|---|---|
| Cls Label | Reg Label | | | | |
| | | | 28.5 | 47.7 | 29.5 |
| ✓ | | | 32.9(+4.4) | 54.0 | 34.9 |
| ✓ | ✓ | | 35.1(+6.6) | 55.0 | 37.6 |
| ✓ | ✓ | ✓ | **36.3(+7.8)** | **56.4** | **38.8** |

(c) **Comparison of two paradigms** in RPN and R-CNN stages.

| RPN | RCNN | mAP | $AP_{50}$ | $AP_{75}$ |
|---|---|---|---|---|
| spa-den | spa-den | 34.5 | 53.8 | 37.1 |
| spa-den | **den**-den | 35.5 | 55.2 | 38.5 |
| **den**-den | **den**-den | **36.3** | **56.4** | **38.8** |

(d) **Comparison of two paradigms** under various labelling ratios.

| Methods | 1% | 5% | 10% |
|---|---|---|---|
| sparse-dense | 21.1 | 31.1 | 34.5 |
| dense-dense | **21.3** | **31.9** | **36.3** |

(e) Study on **training efficiency**. 90k is iterations.

| Methods | 10% ratio | | 100% ratio | |
|---|---|---|---|---|
| | 90k | 180k | 360k | 720k |
| Soft Teacher [8] | - | 34.0 | - | 44.5 |
| Our DTG-SSOD | 33.7 | **36.3** | 44.7 | **45.7** |

Table 3: Ablation studies on hyper-parameters for the proposed DTG-SSOD.

(a) Study on **temperature T**.

| T | mAP | $AP_{50}$ | $AP_{75}$ |
|---|---|---|---|
| 0.5 | 35.6 | 55.1 | 38.4 |
| 1.0 | **35.8** | **55.5** | **38.5** |
| 5.0 | 35.6 | 55.3 | 38.4 |

(b) Study on **loss weight** $\beta$

| $\beta$ | mAP | $AP_{50}$ | $AP_{75}$ |
|---|---|---|---|
| 1.0 | 35.8 | 55.5 | 38.5 |
| 3.0 | **36.3** | **56.4** | **38.8** |
| 5.0 | 36.1 | 55.9 | 38.8 |

(c) Study on **target boxes in INC**.

| Methods | mAP | $AP_{50}$ | $AP_{75}$ |
|---|---|---|---|
| w/o reg | 32.9 | 54.0 | 34.9 |
| all boxes | 34.3 | 54.0 | 36.5 |
| reserved box | **35.1** | **55.0** | **37.6** |

behave as effective supervision signals for learning on unlabeled data. DTG consists of two key components, i.e., Inverse NMS Clustering (INC) and Rank Matching (RM), then we delve into these two parts with results reported in Tab. 2(b). Through imitating the candidate grouping of the teacher in NMS, INC provides appropriate training labels for the student. With the derived classification label, the model enjoys 4.4 mAP gains. When regression labels are also applied, another 2.2 mAP improvement is found, reaching 35.1 mAP. Based on INC, RM is introduced to provide teacher's score rank over clustered candidates for the student, which enables the student to reserve the same candidate in each NMS cluster as the teacher. As results show, RM can bring an obvious improvement, +1.2 mAP, demonstrating comprehensive alignment on NMS behaviour is beneficial. Besides, the gains also indicate the candidate rank carries helpful relation information learned by the teacher.

**Comparison of sparse-to-dense and dense-to-dense paradigms.** In this work, we replace the traditional sparse-to-dense (spa-den for short) paradigm with new dense-to-dense (den-den) paradigm. Detailed comparisons are conducted in Tab. 2(c-d). We first build a sparse-to-dense baseline, then incrementally substitute our den-den paradigm into each network stage (i.e., RPN, RCNN) in Tab. 2(c). Thanks to the appropriate mixing ratio $r = 1/4$ between labeled and unlabeled images and Focal Loss, our sparse-to-dense baseline achieves comparable performance (34.5 mAP) to the previous methods. When we substitute the den-den paradigm into R-CNN, it yields 35.5 mAP, which is 1.0 points higher than the spa-den R-CNN. After that, we further apply den-den to RPN, which attains +0.8 mAP gains. These experiments validate the proposed dense-to-dense paradigm is superior to the traditional sparse-to-dense paradigm in both RPN and RCNN stages.

On the other hand, we also compare two paradigms under various labelling ratios in Tab. 2(d). Under 5% and 10% ratios, our method is obviously better than the sparse-to-dense baseline, and surpasses it by 0.8% and 1.8% mAP, respectively. Nevertheless, when the labelling ratio is 1%, the performance gap shrinks. This is easy to understand: When labeled data is extremely scarce, detection performance is poor. Thus the predictions of the teacher are relatively inaccurate, resulting in noisy teacher guidance. Especially for Rank Matching, the teacher model fails in modeling accurate candidate relations due to weak performance. To extract accurate guidance from the very weak teacher model is left for future work.

**Training efficiency.** Apart from the standard training schedule, we also attempt to halve the training iterations in Tab. 2(e). Under the 10% labelling ratio, our DTG-SSOD yields 33.7 mAP with only 90k iterations, which is very close to the previous best method Soft Teacher with 180k iterations, 34.0 mAP. Besides, under the Fully Labeled Data setting (100% labelling ratio), our method halves the

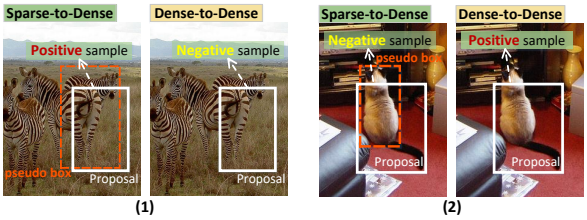


Figure 3: For the *identical* proposals (white), two paradigms will assign *different* training labels. The **poor** and **precise** proposals are displayed in sub-fig (1) and (2), respectively.

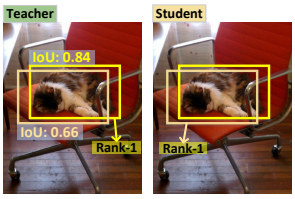


Figure 4: One example to illustrate how candidates are ranked by the teacher and student.

training iterations of Soft Teacher and achieves even better performance, 44.7 vs. 44.5 mAP. These results validate the high efficiency of DTG-SSOD.

**Ablation study on hyper-parameters.** In Tab. 3(a-b), we conduct studies on the hyper-parameters $T$ and $\beta$. When we ablate $T$, $\beta$ is set to 1.0. To conclude, we adopt $T = 1.0$, $\beta = 3$ in all experiments.

**Ablation study on regression labels of INC.** In Inverse NMS Clustering, regression labels have two options: (1) The finally reserved teacher box in each cluster, as described in Sec. 3.2. (2) All teacher's predicted boxes (denoted as "all boxes"). As shown in Tab. 3(c), taking the reserved box as the target achieves superior performance than "all bboxes", 35.1 vs. 34.3 mAP.

### 4.5 Analyses

In this section, we provide some analyses on why dense teacher guidance is superior to sparse pseudo labels. In the sparse-to-dense paradigm, some pseudo boxes are poorly localized, which will mislead the standard IoU-based label assignment and cause noisy training labels for the student. Without the dependency on pseudo boxes, our method can effectively alleviate this problem. Two examples are shown in Fig. 3, where the *identical* student proposals will obtain *different* training labels in the two paradigms. Compared with the sparse-to-dense paradigm, our method obviously offers more precise training labels for the student. In the sub-fig (1), the sparse-to-dense paradigm treats the poor proposal (in white) as a positive sample. Because the proposal has a high IoU value (i.e., 0.6) with the coarse pseudo box (in red), therefore, it reaches the requirement of positive samples in the IoU-based label assignment. In contrast, our Inverse NMS Clustering takes teacher predictions into consideration and succeeds in suppressing this misleading positive. Concretely, the teacher model predicts low confidence (i.e., 0.3) for this proposal, thus it will be a clear background sample in NMS. Another example is shown in the sub-fig (2). Due to the poor localization of the pseudo box, a relatively **precise** proposal is wrongly assigned as a negative sample in the sparse-to-dense paradigm. However, our INC can avoid this problem by taking the teacher's NMS behavior into consideration.

In Fig. 4, we also provide an example to illustrate that the teacher is better at modeling candidate rank. Between the two candidates shown in the figure, the teacher ranks the one with more precise localization first, while the student fails. Therefore, distilling the capacity of modeling candidate rank from the teacher to student is beneficial, which is missed in sparse pseudo labels.

## 5 Conclusion

In this work, we analyze the traditional semi-supervised object detection methods, where *sparse* pseudo boxes with the corresponding category labels are adopted for *dense* supervision of the student, formulating the popular "sparse-to-dense" paradigm. We point out that the sparse-to-dense paradigm could accumulate noise in the hand-crafted NMS and label assignment process, and further lack dense and direct teacher guidance. To address the problem, we propose a novel "dense-to-dense" paradigm, which integrates Dense Teacher Guidance into the student training. Specifically, we introduce Inverse NMS Clustering and Rank Matching to instantiate the dense guidance: INC leads the student to perform the same candidate grouping in NMS as the teacher does; RM passes the teacher's knowledge of rank distribution over clustered candidates to the student. Through INC and RM, the student can receive sufficient, informative, and dense guidance from the teacher, which naturally leads to better performance. On COCO benchmark, extensive experiments under various labelling ratios validate that our DTG-SSOD can achieve SOTA performance in both accuracy and efficiency.

# References


[1] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[2] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021.

[3] Xiang Li, Wenhai Wang, Xiaolin Hu, Jun Li, Jinhui Tang, and Jian Yang. Generalized focal loss v2: Learning reliable localization quality estimation for dense object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11632–11641, 2021.

[4] Fangyun Wei, Yue Gao, Zhirong Wu, Han Hu, and Stephen Lin. Aligning pretraining for detection via object-level contrastive learning. *Advances in Neural Information Processing Systems*, 34, 2021.

[5] Enze Xie, Jian Ding, Wenhai Wang, Xiaohang Zhan, Hang Xu, Peize Sun, Zhenguo Li, and Ping Luo. Detco: Unsupervised contrastive learning for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8392–8401, 2021.

[6] Byungseok Roh, Wuhyun Shin, Ildoo Kim, and Sungwoong Kim. Spatially consistent representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1144–1153, 2021.

[7] Xinlong Wang, Rufeng Zhang, Chunhua Shen, Tao Kong, and Lei Li. Dense contrastive learning for self-supervised visual pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3024–3033, 2021.

[8] Mengde Xu, Zheng Zhang, Han Hu, Jianfeng Wang, Lijuan Wang, Fangyun Wei, Xiang Bai, and Zicheng Liu. End-to-end semi-supervised object detection with soft teacher. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3060–3069, 2021.

[9] Qiang Zhou, Chaohui Yu, Zhibin Wang, Qi Qian, and Hao Li. Instant-teaching: An end-to-end semi-supervised object detection framework. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4081–4090, 2021.

[10] Yihe Tang, Weifeng Chen, Yijun Luo, and Yuting Zhang. Humble teachers teach better students for semi-supervised object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3132–3141, 2021.

[11] Hengduo Li, Zuxuan Wu, Abhinav Shrivastava, and Larry S Davis. Rethinking pseudo labels for semi-supervised object detection. *arXiv preprint arXiv:2106.00168*, 2021.

[12] Yen-Cheng Liu, Chih-Yao Ma, Zijian He, Chia-Wen Kuo, Kan Chen, Peizhao Zhang, Bichen Wu, Zsolt Kira, and Peter Vajda. Unbiased teacher for semi-supervised object detection. *arXiv preprint arXiv:2102.09480*, 2021.

[13] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Advances in neural information processing systems*, 30, 2017.

[14] JongMok Kim, Jooyoung Jang, Seunghyeon Seo, Jisoo Jeong, Jongkeun Na, and Nojun Kwak. Mum: Mix image tiles and unmix feature tiles for semi-supervised object detection. *arXiv preprint arXiv:2111.10958*, 2021.

[15] 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.

[16] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.

[17] Shifeng Zhang, Cheng Chi, Yongqiang Yao, Zhen Lei, and Stan Z Li. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9759–9768, 2020.

[18] Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, page 896, 2013.

[19] Henry Scudder. Probability of error of some adaptive pattern-recognition machines. *IEEE Transactions on Information Theory*, 11(3):363–371, 1965.

[20] Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. *Advances in neural information processing systems*, 17, 2004.

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

[22] 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*, pages 10687–10698, 2020.

[23] Bowen Zhang, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. *Advances in Neural Information Processing Systems*, 34, 2021.

[24] Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. Unsupervised data augmentation for consistency training. *Advances in Neural Information Processing Systems*, 33:6256–6268, 2020.

[25] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *Advances in Neural Information Processing Systems*, 33:596–608, 2020.

[26] Jisoo Jeong, Seungeui Lee, Jeesoo Kim, and Nojun Kwak. Consistency-based semi-supervised learning for object detection. *Advances in neural information processing systems*, 32, 2019.

[27] 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.

[28] Xiang Li, Wenhai Wang, Lijun Wu, Shuo Chen, Xiaolin Hu, Jun Li, Jinhui Tang, and Jian Yang. Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection. *Advances in Neural Information Processing Systems*, 33:21002–21012, 2020.

[29] Haoyang Zhang, Ying Wang, Feras Dayoub, and Niko Sunderhauf. Varifocalnet: An iou-aware dense object detector. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8514–8523, 2021.

[30] Borui Jiang, Ruixuan Luo, Jiayuan Mao, Tete Xiao, and Yuning Jiang. Acquisition of localization confidence for accurate object detection. In *Proceedings of the European conference on computer vision (ECCV)*, pages 784–799, 2018.

[31] Gang Li, Xiang Li, Yujie Wang, Shanshan Zhang, Yichao Wu, and Ding Liang. Knowledge distillation for object detection via rank mimicking and prediction-guided feature imitation. *arXiv preprint arXiv:2112.04840*, 2021.

[32] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[33] Qize Yang, Xihan Wei, Biao Wang, Xian-Sheng Hua, and Lei Zhang. Interactive self-training with mean teachers for semi-supervised object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5941–5950, 2021.

[34] Qiushan Guo, Yao Mu, Jianyu Chen, Tianqi Wang, Yizhou Yu, and Ping Luo. Scale-equivalent distillation for semi-supervised object detection. *arXiv preprint arXiv:2203.12244*, 2022.

[35] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

[36] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

# Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] See abstract.
   (b) Did you describe the limitations of your work? [Yes] See Section 4.4
   (c) Did you discuss any potential negative societal impacts of your work? [No]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See Section 4.2
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Section 4.2
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] See Section 4.1
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Section 4.1
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] See Section 4.1
   (b) Did you mention the license of the assets? [Yes] See Section 4.1
   (c) Did you include any new assets either in the supplemental material or as a URL? [No]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [No]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [No]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]