ReCLIP: Refine Contrastive Language Image Pre-Training with Source Free Domain Adaptation (Supplementary Material)

Figure 1. CLIP performs classification on target classes by comparing visual embeddings with the text embeddings generated from class names.

Appendix I: Background on CLIP

CLIP performs contrastive learning over 400 millions web-retrieved pairs of images and captions by pulling the visual and text representation near if they are from the same pair and away if they are not. At inference stage, CLIP makes classification prediction by matching the visual embeddings of query images with the text embeddings of categories names (wrapped in template text such as ”a photo of {}”, or a list of templates and uses the averaged embedding, as discussed in the main paper), and selects the category with the highest cosine similarity as prediction, as shown in Figure 1. CLIP is capable of performing classification over novel tasks without any training example, as long as the category names are provided. CLIP has demonstrated outstanding zero-shot classification accuracy, e.g. 76.3% top-1 accuracy on ImageNet without seeing any examples from the dataset. [25].

Appendix II: Algorithms

As described in Section 3.3 of the main paper, ReCLIP is composed of two parallel components that are designed for visual and text encoder fine-tuning, namely ReCLIP-V and ReCLIP-T. On top of ReCLIP-T and ReCLIP-V, we integrate the pseudo labels by filtering the commonly-agreed ones to produce high-confidence training signals for both sides. In this Section, we present the detailed description of ReCLIP-T and ReCLIP-V in Algorithm 1, and the pseudo label sharing in Algorithm 2.

Appendix III: Evaluation Benchmarks

For the main result from the paper, we have evaluated our model as well as the baseline methods on the validation or test splits from 22 image classification benchmarks, according to the setup as stated from Radford, et al [25]. The 22 benchmarks is composed of the one ablation datasets AID [32] that we used for hyper-parameter selection, and the 21 benchmarks (Caltech101 [21], CIFAR10 [20], CIFAR100 [20], ImageNet [9], SUN397 [33], Birdsnap [1], Country211 [25], DTD [7], EuroSAT [15], FER2013 [35], FGVC [22], Flowers [23], Food101 [2], GTSRB [27], MNIST [10], Oxford Pet [24], PCam [30], SST2 [25], RESISC45 [6], Cars [19], STL10 [8]) from the 27 benchmarks CLIP reported in Radford, et al [25], except: i) KITTI [13], UCF101 [26], VOC2007 [12], Kinetics700 [3] that are object detection or video classification benchmarks that are out of the scope of our discussion; ii) HatefulMemes [18] and CLEVR [17], where CLIP uses custom splits that are not released at the time of this submission. The detailed statistics on the number of images and the number of classes are reported in Table 1.

For comparison with POUF published score, we reported our scores on the Office-Home datasets. Office-Home contains 65 categories and 15588 images from four different domains: 2427 Art images, 4365 Clipart images, 4439 Product images and 4357 Real-World Images.

Appendix IV: Implementation Details

As mentioned in the main paper, we use AID to choose the best hyper-parameters for each baselines and evaluate them with the same hyper-parameters across the 22 datasets for SFDA evaluation.

For ReCLIP, we use learning rate of $10^{-3}$, weight decay of $10^{-4}$, momentum of 0.9, batch size of 64, maximum length of $\min\{5000 \text{ iterations}, 50 \text{ epochs}\}$ and SGD optimization on both visual and text encoders. For Birdsnap, Country211, SUN397 and ImageNet which have more than 200 classes, we use a batch size of 32 due to large memory occupation from text inputs to fit the training on a single V100 GPU. For Label Propagation, we use propagation strength $\alpha = 0.99$ and neighbor size $k = 20$. For datasets with more than 500 classes (Birdsnap, ImageNet), we notice the accuracy of pseudo labels generated by label propagation becomes unstable, and it requires additional hyper-parameter tuning to achieve good performance. To maintain
Algorithm 1 Visual and Text Encoder Self-Training: ReCLIP-V and ReCLIP-T

Require: Vision Language Pre-trained Model $M = \{M_v, M_t\}$
Require: Unlabeled Images $X = \{x_1, ..., x_n\}$
Require: Class Names $C = \{c_1, ..., c_m\}$
Require: Mode = ReCLIP-V or ReCLIP-T

for epoch $\leftarrow 1$ to Max Epoch do
    $\{t_1, ..., t_m\} \leftarrow M_v(\{c_1, ..., c_m\})$
    $\{v_1, ..., v_n\} \leftarrow M_t(\{x_1, ..., x_n\})$
    $U, S, V \leftarrow \text{svd}([t_1, ..., t_m], \text{where } U = [c_1, ..., c_m]$
    $P_2 \leftarrow [e_2, ..., e_m][e_2, ..., e_m]^T$
    $t_i \leftarrow \frac{t_i P_2}{\|t_i P_2\|}$
    $\tilde{v}_j \leftarrow \frac{\tilde{v}_j}{\|\tilde{v}_j P_2\|}$
    $L \leftarrow \{t_1, ..., t_m, \tilde{v}_1, ..., \tilde{v}_n\}$
    $\hat{Y} \leftarrow \text{LabelPropagation}(L)$
    if Mode=ReCLIP-T then
        $\hat{Y} \leftarrow [\hat{v}_1, ..., \hat{v}_n]^T \{t_1, ..., t_m\}$
        Loss$^T$ $\leftarrow \text{Cross-Entropy}(\hat{Y}, \hat{Y})$
        Back-Propagation over $M_t$
    else if Mode=ReCLIP-V then
        $w_i \leftarrow \left(\sum_{\tilde{y}_j = i} v_j \right) / \left(\sum_{\tilde{y}_j = 1} 1\right)$, for $i \in \{1, 2, ..., m\}$
        $\hat{w}_i \leftarrow \frac{w_i}{\|w_i P_2\|}$ for $i \in \{1, 2, ..., m\}$
        $\hat{Y} \leftarrow [\hat{v}_1, ..., \hat{v}_n]^T [\hat{w}_1, ..., \hat{w}_m]$ $\hat{Y}$
        Loss$^V$ $\leftarrow \text{Cross-Entropy}(\hat{Y}, \hat{Y})$
        Back-Propagation over $M_v$
    end if
end for

Algorithm 2 ReCLIP with Pseudo Label Sharing

Require: Component 1 $M^1 = \{M_v^1, M_t^1\}$ (for ReCLIP-V),
Require: Component 2 $M^2 = \{M_v^2, M_t^2\}$ (for ReCLIP-T)
Require: Unlabeled Images $X = \{x_1, ..., x_n\}$
Require: Class Names $C = \{c_1, ..., c_m\}$

Self-Training Adaptation Stage:

for epoch $\leftarrow 1$ to Max Epoch do
    $Y^1, \hat{Y}^1 \leftarrow \text{ReCLIP-V}(M^1, X, C)$
    $Y^2, \hat{Y}^2 \leftarrow \text{ReCLIP-T}(M^2, X, C)$
    Commonly Agreed Index Map $Q \leftarrow (\hat{Y}_1 = \hat{Y}_2)$
    Loss$^V$ $\leftarrow \text{Cross-Entropy}(\hat{Y}^1(Q), \hat{Y}^1(Q))$
    Loss$^T$ $\leftarrow \text{Cross-Entropy}(\hat{Y}^2(Q), \hat{Y}^2(Q))$
    Back-Propagate $M_t^1$ with Loss$^V$
    Back-Propagate $M_t^2$ with Loss$^T$
end for

Inference Stage:

$\hat{Y}^1 \leftarrow \text{ReCLIP-V}(M^1, X, C)$ $\Rightarrow$ Generate inference predictions $\hat{Y}^1, \hat{Y}^2$ and pseudo labels $\tilde{Y}^1, \tilde{Y}^2$.
$\hat{Y}^2 \leftarrow \text{ReCLIP-T}(M^2, X, C)$ $\Rightarrow$ Boolean Index with $\text{True}$ indicates $Y^1$ agrees with $\tilde{Y}^2$.
Commonly Agreed Index Map $Q \leftarrow (\hat{Y}_1 = \hat{Y}_2)$ $\Rightarrow$ Only calculate loss on entries where $Q$ is True ($\hat{Y}^1$ agrees with $\hat{Y}^2$).
Back-Propagate $M_t^1$ with Loss$^V$
Back-Propagate $M_t^2$ with Loss$^T$

return arg max $i\hat{y}_{ji}$ as prediction for image $x_i$ $\Rightarrow$ Generate inference predictions from ReCLIP-T/V
$\hat{Y} = \{\hat{y}_{ji}\}$, where $\hat{y}_{ji}$ is probability of image $x_j$ on class $i$. $\Rightarrow$ At inference time, ReCLIP-T/V skip the pseudo label generation.
$\hat{y}_{ji}$ $\Rightarrow$ Aggregate prediction logits from both ReCLIP-T/V for prediction.
stable performance, we turn off label propagation and simply use model predictions as pseudo labels on datasets with over 500 categories (Birdsnap, ImageNet). For all other datasets, we follow the exact process as described in Algorithm 1 and 2.

For both AaD and POUF, we have tested different hyper-parameters and report the the best performing setting, with learning rate of $10^{-3}$, weight decay of $10^{-3}$, momentum of 0.9, SGD optimization on AaD, and learning rate of $10^{-2}$, weight decay of $10^{-3}$, momentum of 0.9, SGD optimization on POUF. For both AaD and POUF, we extended their default training length to match the training length of ReCLIP, with batch size of 64 and 5000 iterations on AaD, and batch size of 32 × min{5000 iterations, 100 epochs} steps on POUF.

For ReCLIP on Office-Home, we use the Real-World (Rw) domain to choose the hyper-parameter. We use SGD optimizer with learning rate of $10^{-2}$ on the visual encoder and $10^{-3}$ on the text encoder, batch size of 64 and 5000 iteration as maximum step across all domains. For label propagation, we use $k = 10$ due to the smaller dataset size.

**Appendix V: Additional Ablation Results**

### Choice on Learnable Modules

In Table 2, we evaluate different learnable modules by comparing their fully-supervised fine-tuned performance. As suggested in [31], fine-tuning the normalization weights is shown to be efficient and stable, compared to fine-tuning the entire weights in self-training of ReCLIP.

Recent research [16] as well as POUF [28] also suggests that learnable prompts can also be effective in providing stable and fast performance improvement during the fine-tuning of Transformer [11,29] based models. In Table 2, we perform Visual Prompt tuning following [16], and our own designed Text Prompt. Please refer to Appendix VII for more details.

As shown in Table 2, fine-tuning Layer-Norm weights from Visual Encoder has the best fully supervised accuracy.
inference time). ReCLIP achieves similar improvements in the inductive setting as in the transductive setting.

**Pseudo Label Quality**

In Table 4 we report the pseudo label accuracy of ReCLIP. We report the pseudo label accuracy from ReCLIP on the first epoch, before the self-training algorithm updates the model weights. From Table 4 we observe that the label propagation over projected visual and text embeddings has obtained ReCLIP pseudo labels with consistent improved accuracy over CLIP, only except Birdsnap and ImageNet which have more than 500 categories, as we discussed in Appendix IV. The results from Table 4 demonstrate the effectiveness of our version of the label propagation method in generating reliable pseudo labels for vision-language models. More discussion on pseudo label generation is also covered in Section 4.3.2 of the main paper.

**Appendix VI: Time Analysis**

We present the runtime required by SFDA methods, namely AaD, POUF and ReCLIP, in Table 1. We matched all methods to be at the same training steps for fair comparison. As shown by the result, AaD takes an average of 1.19 hours to adapt. ReCLIP takes 2.35 hours and POUF takes 6.18 hours. ReCLIP is not much slower than AaD although ReCLIP trains two sets of encoders at the same time, except on datasets with more categories due to the time required for the Label Propagation process. However, POUF is much slower than both AaD and ReCLIP, due to its less efficient implementation. However, all three algorithms are very efficient as the adaptation only has to be applied once for each new target domain.

**Appendix VII: Details on the design of learnable Language Prompt**

**What is Language Prompts**

During the large-scale contrastive pre-training, CLIP [25] was trained to match visual-text embedding between training images with their caption sentences such as ‘‘A Golden Retriever dog sitting on grass’’. However, during inference time, category descriptions are usually provided in the form of phrases such as ‘‘Golden Retriever’’ or just ‘‘Dog’’ instead of captions in complete sentences. To mitigate this gap, CLIP has proposed to use templates to wrap the category description phrase into complete sentences to generate better text embeddings.

For optimal performance, CLIP [25] further claims that specific templates which provide contexts to the category names might help generate better text embeddings for classification. For example, CLIP finds the template prompt ‘‘A photo of {category name}, a type of pet’’ works the best for OxfordIII-Pet [24]. CLIP has designed different lists of template prompts for all datasets it was evaluated on. The details can be found on their official GitHub repository https://github.com/openai/CLIP/blob/main/data/prompts.md.

**Learnable Language Prompts**

As demonstrated by CLIP [25], the wisely chosen template prompts might play a vital role in generating accurate text embeddings. However, this process largely depends on the heuristic design. Our goal for the learnable language prompt design is to make the prompt learnable and to avoid having different template prompts for different datasets. Additionally, this can also be an efficient and stable way to fine-tune the performance of CLIP.

We start from the default template prompt ‘‘A photo of {category name}’’, and insert an additional learnable token embedding $t^*$ at the beginning of the sentence, right after the Begin-Of-Sentence (BOS) token, as shown in Figure 2. $t^*$ is initialized with the same embedding value of word ‘‘is’’ for reasonable performance before it is fine-tuned. During the fine-tuning process, token $t^*$ is made to be learnable while token embeddings for all other words are fixed.

**Appendix VIII: Limitation and Future Work**

As mentioned in the Implementation Details section, we have observed that on datasets with more than 500 classes (Birdsnap, ImageNet), the accuracy of pseudo labels generated by label propagation becomes unstable, and it requires additional hyperparameter tuning to achieve good perfor-
mance. To maintain stable performance, we have turned off label propagation and simply used model predictions as our pseudo labels on datasets with over 500 categories. Studies on how the hyper-parameters influence the label propagation performance on datasets with more than 500 categories will be important future work to further improve ReCLIP.

Another future direction will be the utilization of augmentation consistency. Augmentation Consistency has been shown to be a very powerful unsupervised training signal and has been widely applied in unsupervised methods [4, 5, 14]. Due to the scope and complexity of this project, we have not explored the usage of augmentation consistency in source-free domain adaptation. It will be important future work to explore the combination of the current ReCLIP with augmentation consistency to further improve the adaptation performance.

### References


[11] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Syl-


