Exploring Self-training and Pre-training for Unsupervised Domain Adaptation

Simran Arora
Stanford University
simran@cs.stanford.edu

Aldo Gael Carranza
Stanford University
aldogael@stanford.edu

Shikhar Murty
Stanford University
smurty@cs.stanford.edu

Abstract

Self-training has gained interest for unsupervised domain adaptation due to its effective simplicity in extracting information from unlabeled target data, often disregarded in standard representation-based domain adaptation methods. Pre-training has also gathered interest for unsupervised domain adaptation due to the significant performance gains of pre-trained language models on many transfer learning NLP tasks. In this study, we explore both self-training and pre-training for unsupervised domain adaptation in text classification tasks on the WILDS text data sets which contain real-world distribution shifts.

1 Introduction

Unsupervised domain adaptation involves leveraging a model trained on a source domain to a new target domain, often because the target domain is data-poor or labeling data from the target domain is costly. The standard approach to unsupervised domain adaptation is feature representation adaptation [Ganin et al., 2016, Long et al., 2017]. However, recent work has presented major limitations of these methods in settings of low overlap and label shift [Zhao et al., 2019, Johansson et al., 2019]. Moreover, this approach makes no use of any available unlabeled data. For many tasks, unlabeled target domain data is easily accessible. Consequently, over the past few years there has been a significant interest in alternate approaches to unsupervised domain adaptation that do not rely on learning invariant representations. In this work, we investigate two such popular approaches, namely self-training and pre-training / self-supervised learning for unsupervised domain adaptation.

Self-training involves training a classifier on the labeled training data, pseudolabeling the unlabeled data, adding *confidently* labeled points to the training set, and training a new classifier in iteration on the updated training data. Often, these pseudo-labels can be noisy and self-training can put overconfident label beliefs on wrong classes and lead to accumulated errors. Consequently, the success of self-training relies on correct pseuodolabel selection for the unlabeled data. A strict confidence threshold where few pseudolabeled points are added to the training data may not significantly affect the classifiers in future iterations, and a relaxed threshold may lead classifiers learned in future iterations to propagate errors forwards.

In the context of domain adaptation, the presence of distributional shift makes the problem of erroneous pseudo-labeling more prevalent and suitable pseudolabel selection becomes more crucial. Choosing over-confident pseudolabeled data could fit the model better to the source domain but at the same time impair generalization to the target domain. In this work we investigate the effect of changing the threshold on the selection of pseudolabels to test how choosing the most confident

Preprint. Under review.

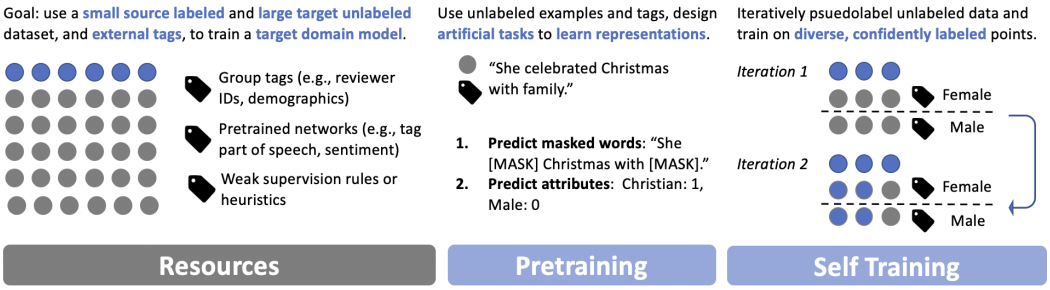

Figure 1: We compare two alternate methods to leverage unlabeled data for domain adaptation, self-training and pre-training, and demonstrate how each method can benefit from the use of the same sets of external knowledge signals. These external signals come from sources such as example metadata, predictions from pretrained networks, and weak supervision rules.

pseudolabels affects domain adaptation. We also explore alternate strategies of utilizing *metadata* to encourage the selection fo diverse set of examples, as a way to regularize over-confident labeling.

Next, given the improved understanding of how to make self-training successful, we evaluate self-training against alternate unsupervised learning methods, namely pre-training. For application areas in natural language processing, pretraining rather than self-training has emerged as the popular approach for leveraging unlabeled data. Pretraining involves designing *pre-text* tasks over the unlabeled data to learn useful representations, which can be used for downstream tasks of interest. The pre-trained representations can be fine-tuned on any task using additional output layers suited for the task. Pre-trained language models have been used for transfer learning across domains using large amounts of in-domain source data [Devlin et al., 2018]. These in-domain pre-trained models effectively find a representation suitable for cross-lingual and cross-domain performance even in low-resource domains. This motivates our exploration of pre-trained language models on unlabeled target data for domain adaptation text classification tasks, as explored in Han et al. [2016], Gururangan et al. [2020]. We consider standard self-supervised tasks for pre-training such as masked language modeling. Moreover, just as in self-training, we leverage auxiliary metadata information to define a more domain-adaptive self-supervised task of predicting metadata group information using textual context.

The two goals of this study are to 1) compare and contrast the performance of self-training and pre-training, two popular unsupervised learning paradigms, in the setting of unsupervised domain adaptation for text classification, and 2) demonstrate how external knowledge (e.g, of group labels over the unlabeled data) can be used to improve the use of unlabeled data in each of these approaches (Figure 1). Notably, with readily available pretrained networks (e.g., sentiment or part of speech taggers), knowledge bases (e.g., NLTK or Wikidata), and weak supervision rules, external knowledge tags are increasingly prevalent. We evaluate our approaches on datasets derived from the WILDS benchmark [Koh et al., 2020].

2 Related Work

Self-training for domain adaptation Self-training is becoming a prominent alternative to unsupervised domain adaptation, especially in image-based tasks such as image classification and semantic segmentation, because it is able to leverage auxiliary unlabeled data from the target domain. Zou et al. [2018] explore the use of self-training in semantic segmentation tasks under distribution shift and they propose a pseudolabel assignment procedure that encourages balance across classes. Zou et al. [2019] explore a form of regularization based on the confidence score of the pseudolabels on the training procedure for domain adaptation in image classification and image segmentation tasks. Mei et al. [2020] introduce a pseudolabel selection procedure that adaptively updates the confidence threshold for each unlabeled point individually which allows for instance-dependent selection, rather than aggregate group-preserving selection. Liu et al. [2021] combines feature representation learning and self-training by running self-training on a domain-adaptive representation and learning an inverse classifier that predicts domains using the pseudolabels, whose predictions are used in a reconstruction loss that can be used to update the representation.

Pre-training for domain adaptation Pre-training is another method of unsupervised training that has led to significant gains in domain adaptation tasks. Gururangan et al. [2020] show that unlabeled data for a specific task improves performance even after domain-adaptive pretraining on the domains of the task. Li et al. [2020] explore the out-of-domain adaptability of a pre-trained cross-lingual language model on in-domain unlabeled data and they propose an information-based unsupervised method that encourages extraction of domain-invariant features suitable for domain adaptation, much like the representation-based domain adaptation methods. In contrast, we explore pre-training using unlabeled target data for text classification. Along this line of research, [Han et al., 2016] propose a method for domain adaptation via pre-training on unlabeled source and target data, then fine-tuning to classify source labeled data.

3 Methods

3.1 Self-training

Self-training [Yarowsky, 1995] for classification tasks is a semi-supervised learning method in which a classifier model p is trained using labeled data L to create *pseudo-labels* for unlabeled data U which are then used for further training. For domain adaptation, we assume L is data sampled from the source domain and U is auxiliary data sampled from the target domain. First, a model is trained to minimize the cross entropy loss on the labeled training data,

$$\theta_s \in \arg\min_{\theta} -\frac{1}{|L|} \sum_{(x_s, y_s) \in L} \log p(y_s | x_s; \theta). \tag{1}$$

Then, for each unlabeled example $x \in U$, the trained model θ_s provides class confidence scores $p(y|x;\theta_s)$ in the form of a probability distribution over classes, and the example is assigned a pseudo-label defined by $\hat{y}(x) = \arg\max_y p(y|x;\theta_s)$. A pseudo-labeled data set L' is formed consisting of samples $(x_u, \hat{y}_u) = (x_u, \hat{y}(x_u))$ for all $x \in U$ whose label confidence exceeds a certain threshold τ. Then, a model is re-trained on the previously labeled data L and the pseudo-labeled data L',

$$\arg\min_{\theta} -\frac{1}{|L \cup L'|} \left(\sum_{(x_s, y_s) \in L} \log p(y_s | x_s; \theta) + \sum_{(x_u, \hat{y}_u) \in L'} \log p(\hat{y}_u | x_u; \theta) \right). \tag{2}$$

This procedure is repeated, updating the labeled data L to include selected pseudo-labeled samples L' and updating the unlabeled data U to exclude the pseudo-labeled features, until no more samples are added to the labeled data set. The final trained model becomes the model we test on the target domain classification task. See Algorithm 1 (in Appendix 6.2) for an implementation of the method. Different implementations differ on the approach for selecting the confident examples, the number of iterations of training before pseudo-labeling again, or the regularization method used to fit the pseudolabeled data.

Pseudolabel selection Pseudolabel selection becomes more critical in the context of domain adaptation because the pseudolabels are generated by a model trained on a different distribution. Sufficient regularization of the selection procedure is necessary to ensure smooth robust changes to the decision boundaries on target domain during self-training given that it is susceptible to greater error propagation under the danger of faulty labeled data. Prior work on regularized/balanced pseudolabel selection leverages confidence metrics from the teacher model or uses unsupervised clustering methods to select pseudolabels for the following iteration of self-training. We follow a selection approach based on leveraging external knowledge. In particular, pre-trained models, knowledge bases, and weak supervision rules can be utilized to cheaply *tag* properties of unlabeled examples and labeled examples. We can use these tags to establish groups within the data set. Then, instead of selecting the most confident pseudolabels across the unlabeled data set, we can select the most confident pseudolabels within each group to ensure some level of diversity.

3.2 Pre-training

Another popular approach to leverage unlabeled data is *pre-training*. The overall goal of pre-training is to learn useful representations that are tailored to the domain of interest. For textual data, a popular approach to learning useful representations is based on the idea of denoising a corrupted version of the input. In particular, consider a text input $x = \{x^1, x^2, x^3, \ldots, x^n\} \in \mathcal{S}$ where each x^i is a word from some vocabulary $\mathcal{V}$ and $\mathcal{S}$ is the space of all such text inputs. Also consider a corruption function $\mathcal{C} : \mathcal{S} \to \mathcal{S}$ which outputs a *corrupted* version of the input (e.g. by replacing a randomly chosen word with a different word). In denoising based pre-training, the objective is to learn a model $f_\theta : \mathcal{S} \to \mathcal{S}$ that can recover the original x. Thus, given an unlabeled corpus of sentences $U = \{x_i\}$, the model is trained to minimize a reconstruction loss,

$$\theta_s \in \arg\min_\theta \sum_{x \in U} \hat{L}(x, f_\theta(C(x))) \tag{3}$$

where $\hat{L}$ is an appropriately chosen loss function.

Masked Language Modeling. One instantiation of this framework is the popular BERT model from Devlin et al. [2018]. The corruption function is chosen to be random masking of words (e.g. for $x =$ ‘‘`The man went to the store`", $\mathcal{C}(x)$ might return ‘‘`The man [MASK] to the store`"). Given an input x with n tokens, the BERT model outputs $p(x_i|C(x); \theta)$ for each of the n tokens. Thus, the objective function (termed masked language modeling or MLM) becomes

$$\theta_s \in \arg\min_\theta \sum_{x \in U} \sum_{x_i \in x} \log p(x_i|C(x); \theta). \tag{4}$$

Pre-training by predicting metadata. In some settings, it might be possible to obtain metadata associated with inputs. For example, source from which the example was curated, other group information such as demographic information of the people mentioned in the example etc. Thus, useful representations may also be learnt by training a model to predict this metadata. Concretely, suppose each input $x^{(i)} \in U$ is labeled with metadata $g^{(i)} = \{g_1^{(i)}, g_2^{(i)}, \ldots, g_k^{(i)}\}$, where each $g_j^{(i)}$ is a categorical variable with an output space of size G_j. Consequently, for an input x, the model outputs $f_\theta(x) = \{f_\theta(x)_1, f_\theta(x)_2, \ldots, f_\theta(x)_k\}$ where $f_\theta(x)_j = p(g_j|x, \theta)$. Thus, the objective function (which we term GroupInfoLoss) becomes

$$\theta_s \in \arg\min_\theta \sum_{x^{(i)} \in U} \sum_{g_j \in g^{(i)}} \log p(g_j|x^{(i)}; \theta). \tag{5}$$

4 Experiments

Datasets We evaluate self-training and pre-training approaches on text classification tasks using two text datasets from the WILDS benchmark Koh et al. [2020]. Each data set contains examples of the form (x, y, d) for input x, label y and group/domain d. For both data sets, the source and target data where chosen to consist of samples from disjoint subset of groups. Therefore, in our case, the source and target domains are a mixture of groups/sub-domains and domain shift occurs across disjoint groups. We use the following datasets and Table 4 (in Appendix 6.1) provides examples from each:

- CIVILCOMMENTS-WILDSincludes comments from online articles (x) that mention different demographic identities (d), such as female or male, and involves predicting whether the comment is toxic (y). Note that the demographic identity groups of the dataset are binary vectors with each coordinate representing affiliation to the corresponding demographic identity since the are not mutually exclusive. The source domain includes examples purely tagged with 5 majority demograpics (i.e. White, Male, Christian, Athiest, and Heterosexual) and the target domain includes examples purely tagged with any other (minority) demographics — the examples in each domain are mutually exclusive.
- AMAZON-WILDScontains product reviews (x) from different users (d), and involves predicting the 5-star product rating (y). The star rating labels of the text inputs are discrete from 1 to 5. The source domain contains reviews from one set of reviewers and the target domain contains reviews from a non-overlapping set of reviewers.

Baselines. We use ERM, trained on the splits in Table 5 (in Appendix 6.1), to baseline the semi-supervised methods.

Benchmark	Model	Avg. Val Acc. (ID)	Avg. Test Acc. (OOD)
CIVILCOMMENTS-WILDS	ERM	91.7%	87.9%
	$\rho = 0.8$	91.8%	87.7%
	$\rho = 0.9$	91.9%	87.8%
	$\rho = 0.95$	91.8%	**88.1**%
	$\gamma = 0.25$	92.1%	88.0%
	$\gamma = 0.33$	91.8%	87.3%
	$\gamma_1, \gamma_2 = 0.75, 0.1$	92.0%	87.6%
Benchmark	Model	Avg. Val Acc. (ID)	Avg. Test Acc. (OOD)
AMAZON-WILDS	ERM	73.1%	70.3%
	$\rho = 0.8$	73.5%	**71.3%**
	$\rho = 0.9$	72.6%	70.2%
	$\gamma = 0.25$	73.3%	71.0%
	$\gamma = 0.33$	72.8%	70.7%
	$\gamma_1, \gamma_2 = 0.5, 0.05$	73.2%	**71.3%**
	$\gamma_1, \gamma_2 = 0.33, 0.05$	73.3%	71.1%
	$\gamma_1, \gamma_2 = 0.05, 0.33$	73.3%	71.0%

Table 1: Self-training vs. ERM

4.1 Self Training Experiments

Implementation. For this work, we conduct 3 rounds of self-training under different pseudolabel selection conditions. For each round, we train the model for the default number of epochs and hyperparametres provided by Koh et al. [2020], and use ERM as the training algorithm. The pseudolabel selection procedures we consider are as follows:

- Fixed threshold (ρ): Given the soft-max scores from the last layer, if the max class score is above a fixed threshold, consider the point to be confidently psuedolabeled.
- Fixed group proportion (γ): First split the unlabeled examples by their group tags (e.g., demographic information). Within each group, rank the examples by their max soft-max score across classes. Select the top proportion of examples from each group as confidently pseudolabeled for the iteration.
- Reducing group proportion (γ_1, γ_2): Same as above, but reduce the proportion selected as self-training iterations progress.

Results and Analysis. Our results are provided in Table 7 and statistics about the number of confidently pseudolabeled unlabeled points included in each iteration of self-training are provided in Table 8 (Appendix). Our two key observations concern 1) the importance of considering the confidence differences between groups, and 2) *when* high overall confidence may be more important than diversity across iterations of self-training.

For certain groups in the data distribution, the model may exhibit lower average confidence. When using the fixed threshold pseudolabel selection procedure, we observe that the number of confidently labeled points widely varies across groups (as defined by reviewer ID for AMAZON-WILDSand demographic idenities for CIVILCOMMENTS-WILDS). On the worst performing, $\rho = 0.9$ for AMAZON-WILDS(See Table 7), the number of confidently pseudolabeled points across the OOD reviewers' IDs has mean 11.6, standard deviation 7.8 examples after Round 0. For certain reviewers in the unlabeled data, the model is much more confident. Non-diverse selections can reinforce popular patterns, which is likely unhelpful under domain shift.

We used the Fixed Group Proportion, defined above, to encourage including diverse points. Fixed Group Proportion, taking the top 33% ranked points in each group, causes us to include the same number of pseudolabeled examples as the $\rho = 0.9$ Fixed soft-max Threshold procedure for the first round of self-training (see Table 8). Under this Fixed Group Proportion, the number of unlabeled

examples added to the training data per reviewer in Round 1 has mean 11.8, standard deviation 1.5; the standard deviation across groups is far reduced. Comparing the two runs, the performance of the classifiers after the first round of self-training improves under the Fixed Group Proportion (See 2).

For CIVILCOMMENTS-WILDSwith a Fixed softmax Threshold of $\rho = 0.95$, the confident points added in the first round are distributed variably across demographic groups with a mean of 75%, standard deviation of 6% of each demographic group (for example only 58% (796/1384) of examples from the "Black" demographic are confident, while 81% (1094/1346) of examples from the "Asian" demographic are confident). When we instead select $\gamma = 75\%$ as a fixed proportion from each demographic, this results in mean of 49%, standard deviation of 16% — the challenge here is that each example can fall under multiple demographic groups. The process of selecting the top proportion in each group can still lead to uneven representations in the final count. We observe that the lower standard deviation (6%) model performed better, but the performance differences are not particularly meaningful. In the future, we would modify the selection process to account for the fact that different examples can have multiple identities.

While diversity is initially helpful, we find that it can hurt performance if enforced in later rounds of pretraining. While the Fixed Group Proportion clearly helped on AMAZON-WILDS, we observe a performance drop after the second self-training iteration and hypothesize that forcing diversity leads examples with low max soft-max scores (confidence) to be included in the training data, even if the points are most confident within their respective groups. We we observe that further restrictivity in later iterations of self-training can be helpful – rather than selecting $\gamma = 0.33$ of examples from each group for each of the iterations, we apply a scheme to select 33% then 5% for Rounds 1 and 2 respectively. This provides a performance boost versus the Fixed Group Proportion models (See green vs. red in Figure 2).

Leveraging more of the pseudolabels helps. We broadly observe across the fixed softmax threshold and the fixed group proportion sets of experiments that leveraging more of the unlabeled data helps more, which is to be expected. We can see this in how the $\rho = 0.8$ and $\gamma = 0.5$ gives top performance, $\gamma = 0.33$ and $\gamma = 0.25$ give mid-level performance, and $\rho = 0.9$ gives the worst performance. This follows the order of the amount of unlabeled test data included in the training set for early self-training iterations. However as a particularly weak confidence condition, we use $\rho = 0.7$ and observe that performance drops vs. $\rho = 0.8$. While leveraging more psuedolabels helps, quality is obviously important.

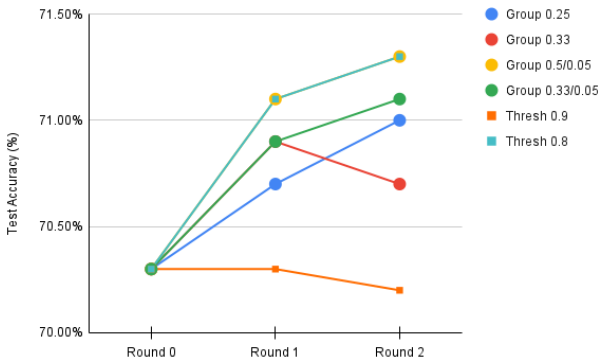

Figure 2: Average test accuracy on the out of domain AMAZON-WILDSreviewers after each round of self-training. In Rounds 1 and 2, pseudolabeled points are used to train the classifier based on the Group or Threshold based confidence conditions.

4.2 Pre-training

Implementation. As above, we use the Distilbert model from Sanh et al. [2019] as our base model. We consider both GroupInfoLoss and MLM loss for our pre-training objective. Moreover, we also experiment with different training schedules. In SaT (Source and Target), we jointly optimize the classification loss on the labeled source domain as well as the pre-training loss on the unlabeled target

domain. In StT (Source then Target), we first optimize the classification loss, and then sequentially finetune on the pre-training loss.

Benchmark	Model	Avg. Val Acc. (ID)	Avg. Test Acc. (OOD)
CIVILCOMMENTS-WILDS	ERM	91.4%	87.2%
	SSL-GroupInfo	92.4%	**92.7%**
	SSL-MLM (SaT)	91.2%	87.1%
	SSL-MLM (StT)	90.8%	87.3%
Benchmark	Model	Avg. Val Acc. (ID)	Avg. Test Acc. (OOD)
AMAZON-WILDS	ERM	70.1%	**66.9%**
	SSL-GroupInfo	69.5%	66.1%
	SSL-MLM (SaT)	70.0%	66.6%
	SSL-MLM (StT)	69.7%	66%

Table 2: Self-supervised learning vs. ERM

Results and Analysis. Results are in Table 2. We observe that using the MLM loss to incorporate unlabeled data is largely unsucessful and obtains very similar results compared to ERM. Most domain adaptation problems where MLM has been sucessful typically involve large genre gaps [Gururangan et al., 2020]. We posit that this may be due to the MLM loss picking up on low frequency signal such as genre information, and since there is no genre difference in our constructed source / target domains, we do not see any meaningful improvements.

Next, we consider GroupInfoLoss, which is a pre-training objective based on predicting group information. For CIVILCOMMENTS-WILDS, this group information included demographic attributes such as gender, race, religion etc and for AMAZON-WILDS, we consider both product ID and product category. We observe a large improvement over all models for CIVILCOMMENTS-WILDS, but do not observe a noticeable gain on AMAZON-WILDS. We believe that this may be because demographic information provides useful features for predicting comment toxicity while product ID and category information may not be inherently useful for predicting the review of a product.

4.3 Low Data Regime

Next we study how self-training and pre-training compare in a low labeled-data regime. This is also a highly practical setting for applying these methods — for example, we can consider an organization on a smaller scale than AMAZON-WILDS, with access to a small amount of labeled customer data. We provide results using the high performing baseline pseudolabel selection procedures and our proposed pretraining objective in Table 3. We observe that the improvement from self-training over ERM tends to increase as the size of the labeled training dataset decreases. We observe less benefit from pre-training in the low-data regime.

5 Conclusion

We conducted experiments to evaluate self-training and pre-training methods for domain adaptation in text classification using AMAZON-WILDS and CIVILCOMMENTS-WILDS. Each approach makes use of unlabeled target data to extract information from the target domain. First, we experimented with different strategies of selecting the pseudolabels in self-training. We considered varying the confidence threshold for selection on all data points and on each metadata group separately. Next, we pre-trained a language model on the standard masked word reconstruction task and on a metadata group prediction task. We evaluated our approaches based on average test accuracy on a target domain disjoint from the source domain. Self-training generally led to slight performance increase or similar performance compared to an ERM baseline across the selection procedures. Pre-training using the group prediction task led to significant increases for the CIVILCOMMENTS-WILDS data set but not so much so on the AMAZON-WILDS. This significant difference suggests a fundamental distinction in the type of domain shift present in both data sets and that self-training and pre-training may be more applicable in different forms of domain shift.

Benchmark	Model, Train Set Size	Avg. Val Acc. (ID)	Avg. Test Acc. (OOD)
AMAZON-WILDS	ERM 5k	68.5%	65.9%
	$\rho = 0.8$, 5k	70.0%	68.2%
	Pretrain-GroupInfoLoss	67.9%	64.4%
	ERM 10k	69.5%	67.1%
	$\rho = 0.8$, 10k	70.9%	69.2%
	Pretrain-GroupInfoLoss	68.4%	66.0%
	ERM 25k	71.2%	68.6%
	$\rho = 0.8$, 25k	71.9%	70.2%
	Pretrain-GroupInfoLoss	70.1%	66.9%
CIVILCOMMENTS-WILDS	ERM 5k	90.6%	86.4%
	$\rho = 0.95$, 5k	88.6%	87.0%
	Pretrain-GroupInfoLoss	90.6%	88.8%
	ERM 10k	91.7%	87.1%
	$\rho = 0.95$, 10k	91.4%	87.1%
	Pretrain-GroupInfoLoss	91.2%	88.9%
	ERM 25k	91.8%	87.8%
	$\rho = 0.95$, 25k	91.7%	87.4%
	Pretrain-GroupInfoLoss	91.6%	89.2%

Table 3: We compare self-training and pre-training in a low labeled data regime, where we have access to a large amount of unlabeled data. We have a total of 50k and 15k unlabeled target domain points for AMAZON-WILDS and CIVILCOMMENTS-WILDS respectively, and use three rounds of self-training.

A potential follow-up to this study would be to explore the combination of pre-training and self-training. Recent work has explored complementary ways to combine self-training and pre-training [Du et al., 2020, Sun et al., 2020]. However, we are not aware of any work that specifically explores the performance of self-training on top of a pre-trained LM in the setting of unsupervised domain adaptation with unlabeled target data. Pretraining and self-training for text classification can be combined as follows. A language model is pretrained using the unlabeled target domain data. Next, the LM is used to initialize the teacher model which will be trained on the labeled source domain data. This trained teacher model is then used to create pseudolabels on the unlabeled data and the model is trained iteratively. Through this procedure, the unlabeled target domain data is used twice to train a model in an unsupervised way, first via LM pretraining and second via pseudolabel self-training to extract more information from the target domain.

References

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Jingfei Du, Edouard Grave, Beliz Gunel, Vishrav Chaudhary, Onur Celebi, Michael Auli, Ves Stoyanov, and Alexis Conneau†. Self-training improves pre-training for natural language understanding. *ArXiv*, https://arxiv.org/pdf/2010.02194.pdf, 2020.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1):2096–2030, 2016.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.740.

Wenjing Han, Eduardo Coutinho, Huabin Ruan, Haifeng Li, Björn Schuller, Xiaojie Yu, and Xuan Zhu. Semi-supervised active learning for sound classification in hybrid learning environments. *PLOS One*, doi:10.1371/journal.pone.0162075, 2016.

Fredrik D. Johansson, D. Sontag, and R. Ranganath. Support and invertibility in domain-invariant representations. *ArXiv*, abs/1903.03448, 2019.

Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Sara Beery, et al. Wilds: A benchmark of in-the-wild distribution shifts. *arXiv preprint arXiv:2012.07421*, 2020.

Juntao Li, Ruidan He, Hai Ye, Hwee Tou Ng, Lidong Bing, and Rui Yan. Unsupervised domain adaptation of a pretrained cross-lingual language model. *arXiv preprint arXiv:2011.11499*, 2020.

Hong Liu, Jianmin Wang, and Mingsheng Long. Cycle self-training for domain adaptation. *arXiv preprint arXiv:2103.03571*, 2021.

Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. *arXiv preprint arXiv:1705.10667*, 2017.

Ke Mei, Chuang Zhu, Jiaqi Zou, and Shanghang Zhang. Instance adaptive self-training for unsupervised domain adaptation. *arXiv preprint arXiv:2008.12197*, 2020.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

Zijun Sun, Chun Fan, Xiaofei Sun, Yuxian Meng, Fei Wu, and Jiwei Li. Neural semi-supervised learning for text classification under large-scale pretraining. *arXiv preprint arXiv:2011.08626*, 2020.

David Yarowsky. Unsupervised word sense disambiguation rivaling supervised methods. In *33rd Annual Meeting of the Association for Computational Linguistics*, pages 189–196, Cambridge, Massachusetts, USA, June 1995. Association for Computational Linguistics. doi: 10.3115/981658.981684.

H. Zhao, Rémi Tachet des Combes, K. Zhang, and Geoffrey J. Gordon. On learning invariant representations for domain adaptation. In *ICML*, 2019.

Yang Zou, Zhiding Yu, BVK Kumar, and Jinsong Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *Proceedings of the European conference on computer vision (ECCV)*, pages 289–305, 2018.

Yang Zou, Zhiding Yu, Xiaofeng Liu, BVK Kumar, and Jinsong Wang. Confidence regularized self-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5982–5991, 2019.

6 Appendix

6.1 Datasets

Task	Example	Domains
Civil Comments	"As a Christian, I will not be patronizing any of those businesses." *Group Info*: Male, White, Christian commenter *Label*: Non-Toxic Comment	**Source**: majority groups (Male, Christian, Athiest, Heterosexual, White) **Target**: minority groups (Female, Black, LGBTQ, etc.)
Amazon Reviews	"I am disappointed in the quality of these. They have significantly deteriorated in just a few uses. I am going to stick with using foil." *Group Info*: Reviewer ID 10,000 *Label*: 1 Star	**Source**: one set of reviewers **Target**: unique set of reviewers

Table 4: Examples for each downstream task.

Benchmark	Train	Valid	Unlabeled Test	Labeled Test
Civil Comments Samples	42k	7k	15k	11k
Amazon Reviews Samples	50k	50k	50k	50k

Table 5: Benchmark task statistics for the initial splits. These datasets are reproducible using the provided code.

6.2 Self-Training

Algorithm 1: Self-training

Input: Labeled data L, unlabeled data U
Output: Trained model θ

repeat
  $\theta \leftarrow train\_model(L)$;
  $L' \leftarrow \varnothing$;
  for $x \in U$ **do**
    if $\max_y p(y|x;\theta) > \tau$ **then**
      $L' \leftarrow L' \cup \{(x, \hat{y}(x))\}$;
    end
  end
  $L \leftarrow L \cup L'$;
  $U \leftarrow U \backslash \{x : \forall (x, \hat{y}) \in L'\}$;
until *no more predictions are confident*;

Self-Training Rounds Statistics For each of the self-training runs reported in Table 7, we provide the raw number of points used for each iteration as pseudolabeled points are selected in Table 8.

Self-Training Results without Synthetic Domain Shift For Civil Comments, we used splits where the source domain contained majority demographic examples and the target contained minority demographic examples. Instead, here we use the original Civil Comments data, without our applied

Benchmark	Model	Round 0	Round1	Round2	Round3*
Civil Comments	ERM	41802	-	-	-
	$\rho = 0.8$	41802	55545	56455	56618
	$\rho = 0.9$	41802	54434	55288	55775
	$\rho = 0.95$	41802	52987	54918	55126
	$\gamma = 0.25$	41802	45345	47379	48756
	$\gamma = 0.33.$	41802	46380	48811	50142
	$\gamma_1, \gamma_2 = 0.75, 0.1$	41802	51845	51948	51994
Amazon	ERM	50000	-	-	-
	$\rho = 0.8$	50000	76037	78631	79774
	$\rho = 0.9$	50000	65243	66230	66491
	$\gamma = 0.25$	50000	62007	70972	77730
	$\gamma = 0.33$	50000	65781	76300	83370
	$\gamma_1, \gamma_2 = 0.5, 0.05$	50000	74673	75210	75210
	$\gamma_1, \gamma_2 = 0.33, 0.05$	50000	65781	67089	68379
	$\gamma_1, \gamma_2 = 0.05, 0.33$	50000	51758	66944	68234

Table 6: Number of training examples used per round. Each round confident pseudolabeled points are added to the training data. *Note we only self-trained on the splits for rounds 0, 1, and 2 – we report the number of confidently labeled points for round 3, should self-training continue.

domain shift and report self-training results. For these splits, the trends from the Fixed Threshold and Fixed Group Proportion pseudolabel selection schemes are much more consistent with the Amazon Reviews dataset. For both these splits and the Amazon Reviews dataset, we believe the ID vs. OOD datasets are much more similar than under our synthetic Civil Comments domain shift based on majority and minority demographic identities, notably the unabeled datasets are also much larger for these splits.

Benchmark	Model	Worst-Group Test Acc.	Avg. Test Acc.
Civil Comments	ERM	58.7%	90.9%
	Ours 0.7 Thresh	56.0%	91.5%
	Ours 0.8 Thresh	61.3%	90.7%
	Ours 0.9 Thresh	58.0%	91.1%
	Ours 0.95 Thresh	58.2%	91.1%
	Ours 0.25 Group Proportion	55.5%	91.3%
	Ours 0.33 Group Proportion	55.1%	90.9%

Table 7: Self-training vs. ERM

Benchmark	Model	Round 0	Round1	Round2	Round3*
Civil Comments	ERM	50000	-	-	-
	Ours 0.7 Thresh	50000	97770	99374	99902
	Ours 0.8 Thresh	50000	96334	99367	99894
	Ours 0.9 Thresh	50000	93756	95442	99176
	Ours 0.95 Thresh	50000	90732	98611	98762
	Ours 0.25 Group Prop.	50000	56555	60936	63031
	Ours 0.33 Group Prop.	50000	58575	63596	66183

Table 8: Number of training examples used per round. Each round confident pseudolabeled points are added to the training data. *Note we only self-trained on the splits for rounds 0, 1, and 2 – we report the number of confidently labeled points for round 3, should self-training continue.