Rethinking Semi-supervised Learning with Language Models

Zhengxiang Shi[1]* **Francesco Tonolini**[2] **Nikolaos Aletras**[2,3] **Emine Yilmaz**[1,2]
Gabriella Kazai[2] **Yunlong Jiao**[2]
[1] University College London, London, United Kingdom
[2] Amazon, London, United Kingdom
[3] University of Sheffield, Sheffield, United Kingdom
zhengxiang.shi.19@ucl.ac.uk
{tonolini,eminey,aletras,gkazai,jyunlong}@amazon.com

Abstract

Semi-supervised learning (SSL) is a popular setting aiming to effectively utilize unlabelled data to improve model performance in downstream natural language processing (NLP) tasks. Currently, there are two popular approaches to make use of unlabelled data: *Self-training* (ST) and *Task-adaptive pre-training* (TAPT). ST uses a teacher model to assign pseudo-labels to the unlabelled data, while TAPT continues pre-training on the unlabelled data before fine-tuning. To the best of our knowledge, the effectiveness of TAPT in SSL tasks has not been systematically studied, and no previous work has directly compared TAPT and ST in terms of their ability to utilize the pool of unlabelled data. In this paper, we provide an extensive empirical study comparing five state-of-the-art ST approaches and TAPT across various NLP tasks and data sizes, including in- and out-of-domain settings. Surprisingly, we find that TAPT is a strong and more robust SSL learner, even when using just a few hundred unlabelled samples or in the presence of domain shifts, compared to more sophisticated ST approaches, and tends to bring greater improvements in SSL than in fully-supervised settings. Our further analysis demonstrates the risks of using ST approaches when the size of labelled or unlabelled data is small or when domain shifts exist. We offer a fresh perspective for future SSL research, suggesting the use of unsupervised pre-training objectives over dependency on pseudo labels.[1]

1 Introduction

Pre-training (PT) language models (LMs) (Devlin et al., 2019; Liu et al., 2019; Radford et al., 2019) over large amounts of text data (e.g. with masked language modelling) and then fine-tuning on task-specific labelled data offer large performance gains across NLP tasks. *Semi-supervised learning* (SSL) (Grandvalet and Bengio, 2004; Chapelle et al., 2009; Kipf and Welling, 2017) is a powerful and effective approach to utilize unlabelled data. A typical SSL setting assumes access to a (relatively small) labelled training set and an (often large) unlabelled set. The goal of SSL is to make effective use of the unlabelled data to improve model (i.e. LMs) performance.

In NLP, *Self-training* (ST) approaches have been proposed to produce pseudo labels for unlabelled examples to train the model (e.g. in Yarowsky, 1995; McClosky et al., 2006). With the advent of neural networks, ST approaches typically focus on using student-teacher models to assign pseudo-labels to the unlabelled data (e.g. in Artetxe et al., 2018; Cai and Lapata, 2019; Dong and de Melo, 2019; Xie et al., 2020a; Gera et al., 2022). Apart from the sophisticated ST approaches, Gururangan et al. (2020) proposed *task adaptive pre-training* (TAPT), which is a straightforward yet effective method for utilising unlabelled examples. This method involves continuing pre-training the LM on the task-specific data without using labels, before proceeding with fully-supervised fine-tuning. TAPT and ST are both motivated by the need for effectively leveraging unlabelled examples, raising the questions of how TAPT performs in SSL tasks, as well as how these two approaches perform against each other.

In this work, we investigate the performance of TAPT against five state-of-the-art ST approaches across five NLP tasks (§4). We empirically show that TAPT outperforms all state-of-the-art ST approaches on several tasks, suggesting that it should serve as a strong baseline for SSL methods. Previous research (Gururangan et al., 2020) has shown that TAPT can improve performance in fully-supervised settings. Our study goes further by showing that TAPT can be even more effective in SSL settings (§4).

We next study the impact of using different

[*] This work was done during an internship at Amazon, Alexa Shopping.

[1] Code is available at https://github.com/amzn/pretraining-or-self-training.

Findings of the Association for Computational Linguistics: ACL 2023, pages 5614–5634
July 9-14, 2023 ©2023 Association for Computational Linguistics

amounts of labelled and unlabelled data for SSL (§5). Our experiments show that ST approaches are prone to suffering from insufficient labelled or unlabelled data, while TAPT is more robust across different combinations of labelled and unlabelled data sizes. Contrary to the common assumption that TAPT requires a large amount of data to perform well (e.g. Li et al., 2021b; Hou et al., 2022), our results show that TAPT improves performance with just a hundred unlabelled samples. We conduct further analysis on the impact of domain shifts in labelled or unlabelled data. While ST approaches generally suffer from domain shifts, TAPT is more robust and even benefits from domain shifts (§6).

In summary, the main contributions of this paper are as follows:

- An extensive empirical study to directly compare five state-of-the-art ST approaches and TAPT across various NLP tasks in SSL, with varying amounts of labelled and unlabelled data as well as the effect of domain shifts;

- Practical insights learned about the limitations of ST approaches, alongside an exploration of the often-unrecognized yet impressive capacity of TAPT as a simple, stable and powerful SSL learner;

- A fresh perspective for future SSL research by demonstrating that leveraging unsupervised signals from unlabelled texts presents a promising and effective approach alternative to dependence on pseudo labels.

2 Preliminaries

2.1 Task Adaptive Pre-training (TAPT)

LMs are adapted to downstream NLP tasks by fine-tuning (FT) on task-specific data. TAPT introduces a simple additional step before fine-tuning by continuing pre-training with a masked language modelling (MLM) objective (Devlin et al., 2019; Liu et al., 2019) on the task-specific data without requiring labels. The main advantage of TAPT is that it provides a simple way for the LM to explore the task space while it can easily make use of all available labelled and unlabelled data.

2.2 Self-training (ST)

The core idea behind ST approaches is to utilise a teacher model trained on labelled examples to make predictions for unlabelled examples, and train a new student model with these predictions. Formally, let $L \triangleq \{(x_1, y_1), \ldots, (x_n, y_n)\}$ denote n labelled examples and $U \triangleq \{\tilde{x_1}, \ldots, \tilde{x_m}\}$ denote m unlabelled examples, where usually $m \gg n$. The ST framework is trained with three main steps as follows.

Step 1. A teacher model F, parameterized by a neural network Θ, is trained via minimizing the cross entropy loss ℓ on labelled examples L:

$$\mathcal{L}_{teacher}(L) = \sum_{x_i, y_i \in L} \ell(y_i, F(x_i, \Theta)), \quad (1)$$

Step 2. The teacher model F is used to make predictions (referred to as "pseudo-labels") on unlabelled examples U:

$$\tilde{y_i} = F(\tilde{x_i}, \Theta), \quad (2)$$

where $\tilde{y_i}$ can be either the continuous logit or the discrete label induced by an ARGMAX operation.

Step 3. A student model G, parameterized by a fresh neural network Φ, is trained to fit labelled and pseudo-labelled examples:

$$\begin{aligned}\mathcal{L}_{student}(L, U) = &\sum_{x_i, y_i \in L} \ell(y_i, F(x_i, \Phi)) \\ &+ \sum_{\tilde{x_i}, \tilde{y_i} \in U} \ell(\tilde{y_i}, F(\tilde{x_i}, \Phi)) \quad (3)\end{aligned}$$

This process is repeated for a given number of times by treating the student as a new teacher to re-predict pseudo-labels as in eq. (2) and then training a new student with eq. (3). In practice, ST with techniques such as consistency regularization (Miyato et al., 2018; Clark et al., 2018; Berthelot et al., 2019b), strong data augmentation (Sohn et al., 2020; Xie et al., 2020b,a), confidence threshold (Sohn et al., 2020; Zhang et al., 2021; Berthelot et al., 2022) usually leads to substantial improvements in model performance.

3 Experimental Setup

Datasets. We experiment with five datasets used in previous related work for SSL (Gururangan et al., 2019; Chen et al., 2020b; Xie et al., 2020a; Li et al., 2021a; Gera et al., 2022), including IMDB (Maas et al., 2011), SST-2 (Wang et al., 2018), AG NEWS (Zhang et al., 2015), AMAZON REVIEW (McAuley and Leskovec, 2013), and YAHOO! ANSWER (Chang et al., 2008). Table 1 shows data

Dataset	Task Type	Train Size	Dev. Size	Test Size	$\lvert\mathcal{Y}\rvert$	L
IMDB (Maas et al., 2011)	Movie Review Sentiment	23,000	2,000	25,000	2	149
SST-2 (Wang et al., 2018)	Movie Review Sentiment	60,000	7,349	872	2	37
AG NEWS (Zhang et al., 2015)	News Topic Classification	100,000	10,000	7,600	4	134
AMAZON REVIEW (McAuley and Leskovec, 2013)	Product Review Sentiment	250,000	25,000	650,000	5	79
YAHOO! ANSWER (Chang et al., 2008)	Topic Classification	500,000	50,000	60,000	10	32

Table 1: Statistics of datasets. $|\mathcal{Y}|$: # of classes for classification tasks. L: average # of words in input sentence(s). Note that we only sample examples from the original training set in our experiments.

statistics. We also provide descriptions and examples of datasets in Appendix §A.1. We show the process for quantifying the similarity between datasets in Appendix §A.2. Adhering to previous work (e.g. Chen et al., 2020b; Wang et al., 2022), we sample the same amount of labelled data per class from the train set, given the labelled size, to form the labelled set. We re-sample the labelled data using the same five seeds for all different approaches and report the average performance with an error bar.

TAPT. Our approach to *task adaptive pre-training* (TAPT) using ROBERTA-BASE (Liu et al., 2019) is to further pre-train on the training text corpus including labelled and unlabelled data (see Table 12 in Appendix for hyperparameter details). The model is then fine-tuned on the labelled data where the `[CLS]` token representation is passed to an extra feed-forward layer for classification (see Table 13 in Appendix for hyperparameter details). The process of TAPT + FINE-TUNING is simply denoted by TAPT henceforth.

ST. We implement five state-of-the-art ST approaches, including VAT (Miyato et al., 2018), FixMatch (Sohn et al., 2020), Dash (Xu et al., 2021b), FlexMatch (Zhang et al., 2021), and AdaMatch (Berthelot et al., 2022) (see descriptions of these approaches in Appendix §B). We use ROBERTA-BASE as the backbone, and the `[CLS]` token representation with an extra feed-forward layer is used for classification (see Table 14 in Appendix for hyperparameter details). Adhering to previous work (Xie et al., 2020a; Wang et al., 2022), back-translation (Ott et al., 2019) is used for data augmentation.

Baselines. For reference, we also evaluate two baseline models that are only fine-tuned (from an off-the-shelf ROBERTA-BASE checkpoint) on: (1) the same labelled set as TAPT and ST (SUPERVISED); and (2) the whole training set (FULLY-SUPERVISED).

4 ST vs TAPT

Overview. Table 2 shows the performance of TAPT against five state-of-the-art ST approaches and the baselines (SUPERVISED and FULLY-SUPERVISED) across five datasets, each with two different sizes of labelled data for training following Wang et al. (2022). Overall, we observe that: (1) TAPT achieves highly competitive results compared with state-of-the-art ST approaches; and (2) TAPT gains more improvement compared to the SUPERVISED baselines when using fewer labelled samples.

For our first finding, the experimental results show that TAPT outperforms all five state-of-the-art ST approaches with lower variances on AMAZON REVIEW, and YAHOO! ANSWER, as shown in Table 2. For example, TAPT obtains a F_1 score of 68.8% compared to the best ST approach's F_1 score of 68.0% (using 500 labelled samples) and 71.5% compared to ST's 69.6% (using 2000 labelled samples) on YAHOO! ANSWER. For an example of the second finding, TAPT gains 3.6% F_1 improvement over SUPERVISED (using 20 labelled samples) compared to 2.2% (using 100 labelled samples) on IMDB. Below we delve deeper into these two findings and discuss them in more detail.

#1. TAPT is a strong semi-supervised learner and can outperform state-of-the-art ST approaches. Figure 1 shows how the performance of ST, TAPT, and SUPERVISED vary with respect to five different labelled sizes on each dataset, where two latest ST approaches (ADAMATCH and FLEXMATCH) are selected as representatives for ST. Experimental results further verify that TAPT has a consistent advantage over ADAMATCH and FLEXMATCH across different labelled sizes on AMAZON REVIEW and YAHOO! ANSWER. It is also worth noting that, while TAPT brings a stable improvement over SUPERVISED across all datasets with varying labelled sizes, ST can sometimes bring more substantial improvement, for example when

Method	IMDB		SST-2		AG NEWS		AMAZON REVIEW		YAHOO! ANSWER	
	20	100	40	100	40	200	250	1000	500	2000
ST Approaches										
VAT	$90.2_{0.9}$	$92.0_{0.4}$.	$75.0_{12.0}$	$86.2_{3.4}$	$87.5_{1.0}$	$89.5_{0.7}$	$52.2_{1.3}$	$57.5_{0.2}$	$66.9_{0.5}$	$68.6_{0.2}$
FIXMATCH	$93.4_{0.1}$	$93.4_{0.1}$	$37.3_{8.5}$	$66.4_{21.3}$	$75.6_{8.7}$	$88.8_{0.6}$	$55.9_{1.1}$	$59.0_{0.5}$	$67.5_{1.0}$	$69.6_{0.4}$
DASH	$93.2_{0.3}$	$93.4_{0.2}$	$38.2_{10.1}$	$73.3_{18.6}$	$74.3_{6.6}$	$88.5_{0.6}$	$56.6_{1.8}$	$59.3_{0.2}$	$67.6_{1.0}$	$69.5_{0.3}$
FLEXMATCH	$93.3_{0.1}$	$93.4_{0.1}$	$40.6_{7.7}$	$83.0_{8.3}$	$80.6_{4.4}$	$88.2_{0.5}$	$54.9_{3.9}$	$58.8_{0.4}$	$66.6_{0.7}$	$68.7_{0.4}$
ADAMATCH	$94.4_{0.4}$.	$94.7_{0.2}$	$42.6_{13.3}$	$83.1_{4.4}$	$82.7_{5.9}$	$88.6_{0.4}$	$55.5_{2.8}$	$59.0_{0.7}$	$68.0_{0.7}$	$69.5_{0.3}$
SUPERVISED	$83.3_{7.4}$	$88.7_{0.2}$	$74.7_{6.1}$	$84.0_{2.7}$	$84.6_{1.6}$	$88.0_{0.8}$	$53.1_{0.7}$	$57.2_{0.1}$	$65.4_{0.3}$	$68.5_{0.3}$
+ TAPT	$86.9_{2.8}$	$90.9_{0.6}$	$82.6_{4.0}$	$85.4_{2.4}$	$84.0_{1.3}$	$88.7_{0.7}$	$58.4_{0.7}$	$60.6_{0.1}$	$68.8_{0.7}$	$71.5_{0.3}$
FULLY-SUPERVISED	$93.9_{0.1}$		$93.0_{0.6}$		$94.8_{0.1}$		$65.0_{0.2}$		$75.3_{0.2}$	
+ TAPT	$94.0_{0.2}$		$93.5_{0.3}$		$95.0_{0.1}$		$65.6_{0.1}$		$75.4_{0.1}$	

Table 2: Performance of TAPT, ST approaches and the baselines across five datasets using two different sizes of the training labelled data. We report average Macro-F_1 on the test set across five seeds, with standard deviations in subscripts. Blue and orange represent the best and second-best performance in a column respectively.

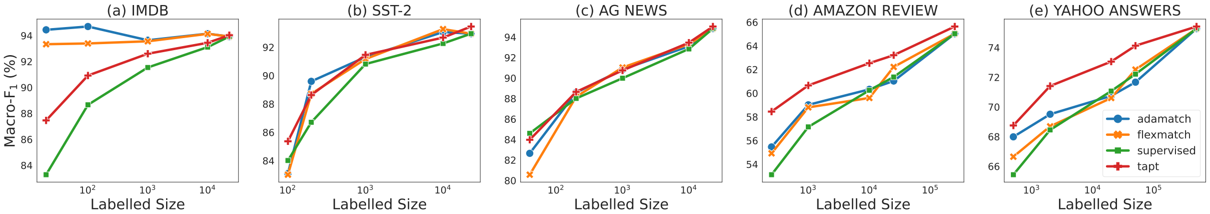

Figure 1: The effect of labelled size on TAPT and ST. Average test Macro-F_1 score over 5 seeds is reported. From the left to the right, TAPT and ST utilizes 23k, 60k, 100k, 250k, and 500k unlabelled samples respectively.

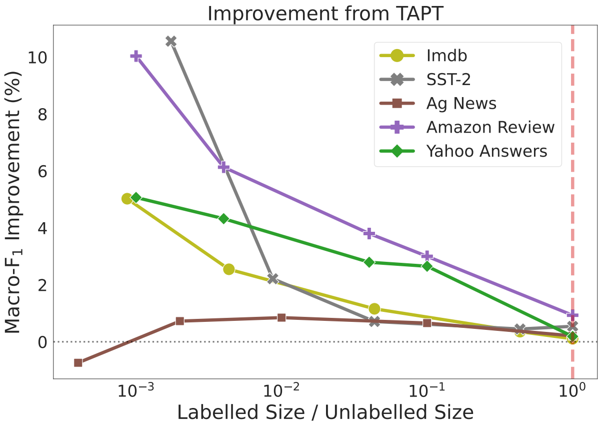

Figure 2: The impact of labelled size on the F_1 improvement from TAPT over SUPERVISED, where unlabelled size is fixed for each dataset. The red vertical line highlights the FULLY-SUPERVISED setting on which prior work (Gururangan et al., 2020) focuses.

only a few hundreds of labelled samples are available from IMDB. However, we do not observe similar phenomena for ST on other datasets. Our experimental results demonstrate that TAPT is a simple, effective and strong learner for SSL tasks, and it should serve as a baseline for SSL tasks in NLP.

#2. TAPT tends to bring more improvements in SSL than in FULLY-SUPERVISED setting. We further study the behaviour of TAPT *itself* under SSL, where we select SUPERVISED as the baseline rather than ST approaches. Figure 1 shows that the differences in performance (in absolute values) between TAPT (red lines) and SUPERVISED (green lines) generally increase as the labelled size decreases. To gain a better understanding of the impact of labelled data sizes, we plot the improvement from TAPT over SUPERVISED (in percentages) against the ratio between labelled size and unlabelled size (unlabelled size is fixed for each dataset) in Figure 2. We see that TAPT improves over SUPERVISED further as the ratio of labelled and unlabelled sizes decreases, highlighting the trends of gaining greater improvement in low-resource SSL setting. This finding is complementary to prior works (e.g. in Howard and Ruder, 2018; Gururangan et al., 2020) that focus on TAPT's improvement from the FULLY-SUPERVISED perspective, represented by the rightmost red vertical line in Figure 2. The rising trend of the improvement is not monotonic as the labelled size is reduced. Rather it could provide insight into how TAPT improves over SUPERVISED in SSL and inspire the design of new

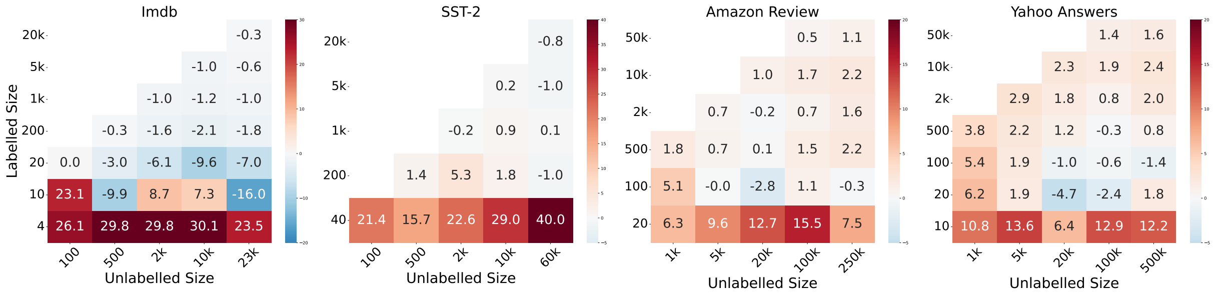

Figure 3: Performance difference between TAPT and ST with varying labelled and unlabelled sizes on IMDB, SST-2, AMAZON REVIEW and YAHOO! ANSWER. Positive values indicate that TAPT performs better, while negative values indicate that ST performs better. Average Macro-F_1 score on test sets over five seeds is reported.

approaches.

5 Exploring the limits of ST and TAPT

In §4, our experimental results showed inconsistent results across datasets. For example, ST performs better on IMDB while TAPT achieves better results on AMAZON REVIEW and YAHOO! ANSWER. We hypothesize that this might be attributed to the exposure to different sizes of labelled or unlabelled data. To verify this hypothesis and shed light on the differences in performance between datasets, we compare TAPT and ST (using ADAMATCH and FLEXMATCH as representatives) by sampling different labelled and unlabelled sizes in IMDB, SST-2, AMAZON REVIEW and YAHOO! ANSWER.

Figure 3 visualizes the differences in performance between TAPT and ST, where each cell represents the macro-F_1 performance difference of TAPT over ST (averaged across five seeds). In each case, the highest performance among FLEXMATCH and ADAMATCH is selected to represent the performance of ST. Overall, we observe that: (1) TAPT improves the fine-tuning performance even with a few hundred unlabelled examples; and (2) TAPT performs more stable across the different labelled and unlabelled data sizes than ST approaches. Below we provide a comprehensive analysis of the impact of labelled and unlabelled sizes.

#1. TAPT works even with a few hundred unlabelled samples. It is generally assumed that TAPT requires a large amount of unlabelled data to perform well (e.g. Li et al., 2021b; Hou et al., 2022). However, we surprisingly observe that TAPT can bring substantial improvement over SUPERVISED baseline even with a relatively small number of unlabelled samples, as shown in Figure 5. To explore the effectiveness of TAPT over SUPERVISED in the low-resource setting of unlabelled data, we select the performance of TAPT and SUPERVISED from the first column (the lowest unlabelled size) for each dataset in Figure 3 and plot their average performance over different labelled sizes. Figure 4 shows that TAPT improves over the SUPERVISED baseline with just one hundred or one thousand samples. For instance, TAPT achieves a 5.5% increase in F_1 score compared to the SUPERVISED baseline when using only 1k unlabelled samples on YAHOO! ANSWER. Additionally, this performance is achieved without the need for large amounts of tokens in each sample, as training samples from SST-2, on average, contain only 9 tokens and training samples from YAHOO! ANSWER contain about 32 tokens (see examples in Table 6 of Appendix).

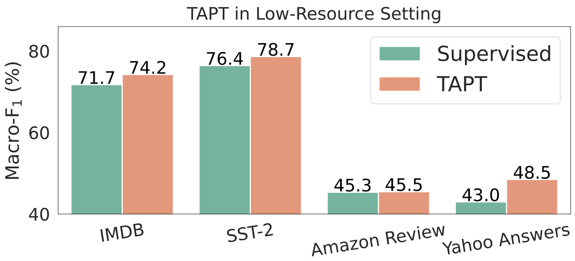

Figure 4: The performance of TAPT against the SUPERVISED baseline in the low-resource setting of unlabelled data. From the left to the right, TAPT utilizes 100, 100, 1 000, and 1 000 unlabelled samples respectively.

#2. Scarce labelled data and adequate unlabelled data. TAPT appears to be a more favourable choice than ST approaches in this setting. The bottom of each sub-figure in Figure 3 shows a clear labelled size boundary, below which FLEXMATCH and ADAMATCH are outperformed by TAPT with a large margin, regardless of datasets and unlabelled size used. This suggests that ST might not be able to efficiently handle large amounts of unlabelled data if labelled data

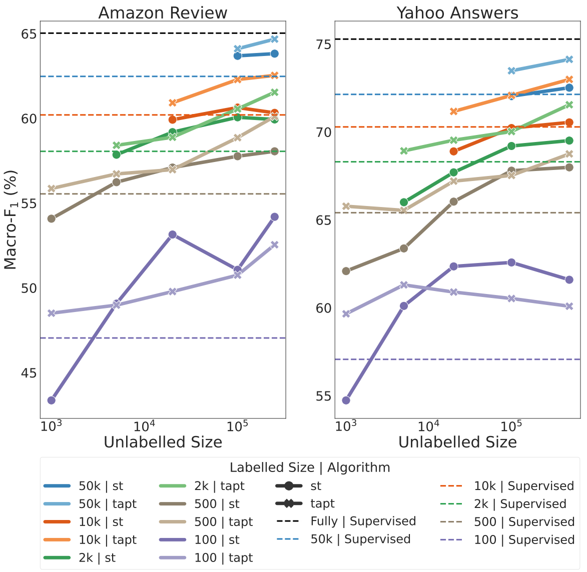

Figure 5: The performance of ST and TAPT using different unlabelled sizes. Average test results across five seeds are reported where the best result from FLEXMATCH and ADAMATCH is selected to represent ST.

do not provide adequate information. This might be attributed to *confirmation bias* (Tarvainen and Valpola, 2017; Arazo et al., 2020), which results from the accumulation of errors in the iterative ST process caused by incorrect pseudo-labels.

The specific value of adequate labelled size boundary for ST approaches depends on the nature of the dataset. For example, even though both IMDB and SST-2 are binary classification tasks for movie review sentiment analysis, the labelled size boundary for SST-2 is higher ($40 > 4$), indicating that this boundary tends to increase as the task becomes more challenging. While it may be easy to obtain dozens of labelled data in this case, when the task becomes more intricate or contains noisy weak labels, it is important to be aware of this potential issue with ST approaches. TAPT could serve as an alternative in situations where collecting adequate labelled data for training is costly. We provide specific values of the performance of ST and TAPT, and further verify that this finding applies to other ST approaches in Appendix §D.

#3. Adequate labelled data and scarce unlabelled data. In this setting, TAPT is more robust, while ST has a greater chance of performing worse than the SUPERVISED baseline. In Figure 5, we plot the performance of ST approaches and TAPT against five different sizes of unlabelled data, grouped by size (using similar colours). We note that ST approaches perform worse than their corresponding SUPERVISED baselines (represented by horizontal lines) until a certain amount of unlabelled data has been reached. For example, when the labelled size is 500, ST requires about 20k unlabelled samples to achieve the corresponding SUPERVISED baseline performance on YAHOO! ANSWER. On the other hand, TAPT generally outperforms SUPERVISED baselines demonstrating its robustness across various unlabelled sizes.

#Unl.	**10**	**50**	**100**	**500**
FLEXMATCH	$57.3_{17.9}$	$35.2_{3.4}$	$45.1_{22.5}$	$33.4_{0.1}$
ADAMATCH	$53.3_{22.1}$	$36.8_{6.1}$	$33.5_{0.2}$	$33.6_{0.3}$

Table 3: Test results on IMDB with 4 fixed labelled data. An average Macro-F_1 score over five seeds is reported.

To further quantify the model performance in case of scarce unlabelled and adequate labelled data, we choose the three lowest unlabelled sizes (the first three columns) excluding the lowest labelled size (the last row) in Figure 3 for each dataset. Our analysis shows that ST has 67%, 56% and 54% probability of falling below the SUPERVISED baselines on SST-2, AMAZON REVIEW, and YAHOO! ANSWER respectively. Even on IMDB where ST generally performs well, it still has a probability of 33% to fall behind SUPERVISED. In contrast, TAPT never performs worse than SUPERVISED in those cases. We provide computation details and comparative statistics in Appendix §C.

The specific value of adequate unlabelled size boundary for ST approaches depends on the nature of the dataset as well as the labelled size. Figure 5 illustrates that as the size of the labelled data increases, ST approaches require more unlabelled data to surpass the SUPERVISED baselines. For example, on AMAZON REVIEW, ST trained with 100 labelled samples requires about 5k unlabelled samples to perform better than SUPERVISED, while ST trained with 10k labelled samples requires about 100k unlabelled samples. Adjusting the unlabelled size accordingly might be conducive to exploiting the full potential of ST approaches.

#4. Scarce labelled and unlabelled data. When the labelled data is insufficient, increasing unlabelled size is not helpful or even detrimental to ST approaches. This finding is well-illustrated in the last row of results on SST-2 shown in Figure 3. In

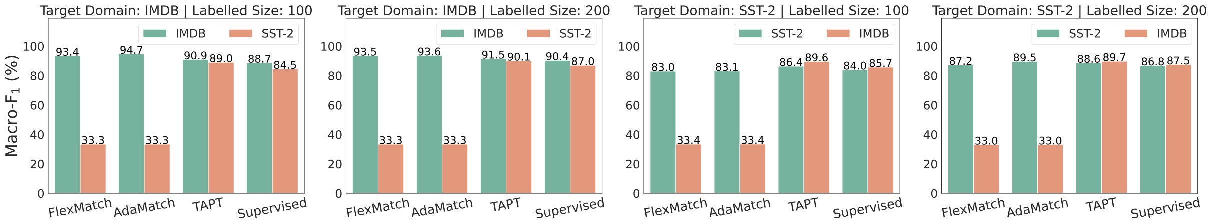

Figure 6: Results of UDA experiments. Legends indicate domains of labelled training data. Orange/green represents the performance with/without domain shift. Average Macro-F_1 score on test sets over five seeds is reported.

Train (Lab.)	Train (Unl.)	#Lab.	FLEXMATCH	ADAMATCH	TAPT	SUPERVISED	
IMDB	IMDB	100	$93.4_{0.1}$	$94.7_{0.2}$	$90.9_{0.6}$	$88.7_{0.2}$	★
	SST-2	100	$89.1_{1.2}$ (▼4.6%)	$87.6_{2.2}$ (▼7.5%)	$89.9_{0.6}$ (▼1.1%)	$88.7_{0.2}$	
	AMAZON REVIEW	100	$92.1_{0.7}$ (▼1.4%)	$92.4_{0.2}$ (▼2.4%)	$91.4_{0.3}$ (▲0.6%)	$88.7_{0.2}$	
IMDB	IMDB	200	$93.5_{0.1}$	$93.6_{0.1}$	$91.8_{0.3}$	$90.3_{0.4}$	★
	SST-2	200	$89.5_{2.4}$ (▼4.3%)	$88.9_{1.0}$ (▼5.0%)	$90.3_{0.4}$ (▼1.6%)	$90.3_{0.4}$	
	AMAZON REVIEW	200	$92.5_{0.4}$ (▼1.1%)	$92.7_{0.5}$ (▼1.0%)	$92.1_{0.2}$ (▲0.3%)	$90.3_{0.4}$	
SST-2	SST-2	100	$83.0_{8.3}$	$83.1_{4.4}$	$85.4_{2.4}$	$84.0_{2.7}$	★
	IMDB	100	$46.7_{2.1}$ (▼43.7%)	$49.2_{7.3}$ (▼40.8%)	$88.5_{0.9}$ (▲3.6%)	$84.0_{2.7}$	
	AMAZON REVIEW	100	$46.4_{4.9}$ (▼44.1%)	$48.2_{11.0}$ (▼42.0%)	$88.9_{0.9}$ (▲4.1%)	$84.0_{2.7}$	
SST-2	SST-2	200	$87.2_{3.9}$	$89.5_{0.9}$	$88.6_{0.9}$	$86.8_{0.3}$	★
	IMDB	200	$62.7_{7.4}$ (▼28.1%)	$61.0_{2.8}$ (▼31.8%)	$89.1_{1.1}$ (▲0.6%)	$86.8_{0.3}$	
	AMAZON REVIEW	200	$61.8_{7.7}$ (▼29.1%)	$56.0_{10.3}$ (▼17.4%)	$89.4_{1.0}$ (▲0.9%)	$86.8_{0.3}$	

Table 4: Results of STL experiments. We report the average Macro-F_1 score on the test set across five seeds, with standard deviations as subscripts. Blue represents the best result for each row. Stars highlight rows without domain shifts. Arrows in colours stand for the changes in performances against the star row result within each cell.

Task	Lab.	Unl.
Semi-supervised Learning	Target	Target
Unsupervised Domain Adaptation	Source	Target
Self-taught Learning	Target	Source

Table 5: A summary of domain adaptation, where the distribution of source and target domains are different.

other words, reducing the size of unlabelled data could be beneficial for ST approaches when the labelled size is inadequate. We further zoom in on this phenomenon in Table 3 by selecting 4 fixed labelled and 500 unlabelled samples, and gradually removing unlabelled samples on IMDB. This is a stark contrast to the case where more unlabelled data is beneficial for ST approaches when adequate labelled data is available. Meanwhile, TAPT generally benefits from training on more in-domain unlabelled data, following the scaling law in LMs (Kaplan et al., 2020; Hoffmann et al., 2022).

#5. Adequate labelled and unlabelled data. Both ST and TAPT have demonstrated the ability to exploit unlabelled data in this setting. Figure 3 shows that ST dominates in IMDB when more than 10 labelled and 100 unlabelled samples are available. On the other hand, TAPT generally performs better than ST on AMAZON REVIEW and YAHOO! ANSWER, indicating that the answer to which approach is better depends on the nature of the dataset and task. As labelled and unlabelled data size increase, the difference between ST and TAPT shrinks (colours fade and lines converge in Figures 3 and 5). As the labelled data in size reaches the unlabelled data, the method of ST reduces to FULLY-SUPERVISED, which is generally outperformed by TAPT (Gururangan et al., 2020).

6 Domain Adaptation

We next investigate how ST and TAPT compare in the presence of domain shifts between labelled and unlabelled data in two additional settings (refer to Table 5). First, we experiment with the *Unsupervised Domain Adaptation* (UDA) setting, where domain shifts exist between the labelled data from a source domain and the unlabelled data from the target domain (Ben-David et al., 2010; Saito et al., 2018; Ramponi and Plank, 2020). Then, we experiment with *Self-taught Learning* (STL) (Raina et al., 2007) in a domain adaptation setting, where the unlabelled data come from the source domain and the labelled data from the target domain. In both settings, we use the (labelled) validation and test sets from the target domain. Validation and test sets are excluded from any pool of labelled or unlabelled train data.

#1. Unsupervised Domain Adaptation (UDA). In this setting, we use two movie sentiment datasets, IMDB and SST-2, as the source and target domain (and vice versa) with two different sizes of labelled data (i.e. 100 and 200).

Figure 6 depicts the performance of ST and TAPT in UDA. In case of domain shifts, we observe that FLEXMATCH and ADAMATCH fail to deliver satisfactory results and their performance drops to the level of random guessing, with a F_1 score of 33% across all labelled sizes and datasets. This highlights the vulnerability of ST approaches in UDA. In contrast, TAPT demonstrates robust performance even with domain shifts, on par with its own SSL performance without domain shifts. Additionally, TAPT even benefits from training on the source domain. For instance, training on IMDB (source domain) further improves the performance of TAPT on SST-2 (target domain) from 86.4% to 89.6% with 100 labelled samples and from 88.6% to 89.7% with 200 labelled samples.

#2. Self-taught Learning (STL). We select IMDB, SST-2, and AMAZON REVIEW for this setting. Although they are all sentiment reviews datasets, IMDB and AMAZON REVIEW are more closely related (see the similarity analysis in Table 7 of Appendix) and arguably contain richer language than SST-2 (see examples in Table 6 of Appendix).

Table 4 presents the performance of ST and TAPT in STL setting. We find that domain shifts in unlabelled data consistently hurt the performance of ST, depending on the similarity between the source and target domains. The performance of ST drops sharply if the source and target domains are vastly different. For example, when SST-2 is used as the labelled data (target domain) and IMDB or AMAZON REVIEW is used as unlabelled data (source domain), the performance of ST falls from over 80% to around 60% or lower. On the other hand, when using SST-2 and IMDB as the source and target domains, the performance of ST drops by a much smaller margin (a few percentage points). This shows the importance of training ST approaches using more informative labelled data, which is also consistent with our findings in §5.

TAPT in the STL setting is in fact a variation of *domain adaptive pre-training* (Beltagy et al., 2019; Gururangan et al., 2020) applied to SSL tasks. Table 4 shows that the performance of TAPT remains stable when there exist domain shifts in the unlabelled data. Using more informative unlabelled data can further improve the performance of TAPT. For example, using IMDB or AMAZON REVIEW as unlabelled data when SST-2 is a target task, we see an improvement of about 4% with 100 labelled samples. However, it is worth noting that ST methods can still be competitive compared to TAPT if the source and target domains are relatively similar. For instance, when using AMAZON REVIEW and IMDB as the source and target domains, ST still achieves better results than TAPT.

7 Related Work

Leveraging unlabelled data by Continuing Pre-training. Previous work has shown that further pre-training LMs on the unlabelled data of a task (e.g. Alsentzer et al., 2019; Mehri et al., 2020; Margatina et al., 2022) or in-domain data (e.g. Logeswaran et al., 2019; Gururangan et al., 2020; Xue et al., 2021) is beneficial to downstream tasks. However, it is unknown whether this is valid in SSL settings. Previous studies in computer vision (Zoph et al., 2020) and speech recognition (Xu et al., 2021a) have compared PT and ST. However, our study has a different focus, specifically, we compare TAPT and ST in NLP tasks. Concurrently to our work, Shi and Lipani (2023) put forward prompt-based continued pre-training, which primarily aims to enhance the performance of prompt-based fine-tuning techniques (Schick and Schütze, 2021; Gao et al., 2021). This approach outperforms these state-of-the-art ST approaches (Sohn et al., 2020; Xu et al., 2021b; Zhang et al., 2021; Berthelot et al., 2022) as well as the conventional CLS-based fine-tuning with TAPT.

Semi-supervised Learning. Recent work in SSL has demonstrated great progress in effectively exploiting unlabelled data. A wide range of approaches has been proposed including Pseudo Labeling (Lee et al., 2013), Temporal Ensemble (Laine and Aila, 2017), Mean Teacher (Tarvainen and Valpola, 2017), Virtual Adversarial Training (Miyato et al., 2018), FixMatch (Sohn et al., 2020). A major issue for ST approaches is *confirmation* *bias*, where the student model would accumulate errors from the teacher model when learning with inaccurate pseudo-labels (e.g. Wang et al., 2021; Goel et al., 2022; Chen et al., 2022).

While many efforts towards ST (e.g. Ruder and Plank, 2018; Gururangan et al., 2019; Li et al., 2019; Chen et al., 2020b; Meng et al., 2020; Chen et al., 2020a; He et al., 2020; Gera et al., 2022) have been made in NLP, the performance of ST approaches across various labelled and unlabelled sizes has yet to be thoroughly explored. Although Mukherjee and Awadallah (2020); Li et al. (2021b) noted that training ST approaches from TAPT checkpoints can improve the performance, the performance of TAPT in SSL tasks has not been either well-researched by previous works or compared with state-of-the-art ST approaches.

8 Conclusion

In this work, we shed light on how TAPT performs against state-of-the-art ST approaches in various SSL settings. Our experiments reveal that TAPT achieves strong and robust performance, even with just a few hundred unlabelled examples. We further demonstrate that the ST approaches are vulnerable to small amounts of either labelled or unlabelled data. We also find that TAPT is more robust than ST approaches in joint domain adaptation and SSL settings. Overall, our empirical study demonstrates that TAPT is a strong SSL learner, competitive to more sophisticated ST approaches. In future work, we plan to further explore the potential of TAPT with unsupervised learning signals.

Limitations

For easier comparison with previous work, we only focus on text classification tasks, while ST can also be applied to a variety of NLP tasks, such as language generation, conversational systems and commonsense reasoning (Kedzie and McKeown, 2019; He et al., 2020; Shi et al., 2022a,b; Hendriksen et al., 2022). We also assume that the datasets are roughly balanced. However, real-world datasets are usually class-imbalanced (Li et al., 2011), which might impact the performance of TAPT and ST. While this is out of the scope of this paper, we believe that this is an interesting avenue for future work. Additionally, different labelled and unlabelled sizes may impact the performance of ST approaches in the domain shift setting. However, this doesn't alter our conclusion that the effectiveness of ST approaches significantly fluctuates across different scenarios.

References

Emily Alsentzer, John Murphy, William Boag, Wei-Hung Weng, Di Jindi, Tristan Naumann, and Matthew McDermott. 2019. Publicly available clinical BERT embeddings. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 72–78, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Eric Arazo, Diego Ortego, Paul Albert, Noel E O'Connor, and Kevin McGuinness. 2020. Pseudo-labeling and confirmation bias in deep semi-supervised learning. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018. A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 789–798, Melbourne, Australia. Association for Computational Linguistics.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620, Hong Kong, China. Association for Computational Linguistics.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. 2010. A theory of learning from different domains. *Machine learning*, 79(1):151–175.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, page 41–48, New York, NY, USA. Association for Computing Machinery.

David Berthelot, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Kihyuk Sohn, Han Zhang, and Colin Raffel. 2019a. Remixmatch: Semi-supervised learning with distribution alignment and augmentation anchoring. *arXiv preprint arXiv:1911.09785*.

David Berthelot, Nicholas Carlini, Ian J. Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. 2019b. Mixmatch: A holistic approach to semi-supervised learning. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 5050–5060.

David Berthelot, Rebecca Roelofs, Kihyuk Sohn, Nicholas Carlini, and Alexey Kurakin. 2022. Adamatch: A unified approach to semi-supervised learning and domain adaptation. In *International Conference on Learning Representations*.

Rui Cai and Mirella Lapata. 2019. Semi-supervised semantic role labeling with cross-view training. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1018–1027, Hong Kong, China. Association for Computational Linguistics.

Ming-Wei Chang, Lev Ratinov, Dan Roth, and Vivek Srikumar. 2008. Importance of semantic representation: dataless classification. In *Proceedings of the 23rd national conference on Artificial intelligence-Volume 2*, pages 830–835.

Olivier Chapelle, Bernhard Scholkopf, and Alexander Zien. 2009. Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]. *IEEE Transactions on Neural Networks*, 20(3):542–542.

Baixu Chen, Junguang Jiang, Ximei Wang, Pengfei Wan, Jianmin Wang, and Mingsheng Long. 2022. Debiased self-training for semi-supervised learning. In *Advances in Neural Information Processing Systems*, NIPS'22.

Jiaao Chen, Zhenghui Wang, Ran Tian, Zichao Yang, and Diyi Yang. 2020a. Local additivity based data augmentation for semi-supervised NER. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1241–1251, Online. Association for Computational Linguistics.

Jiaao Chen, Zichao Yang, and Diyi Yang. 2020b. MixText: Linguistically-informed interpolation of hidden space for semi-supervised text classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2147–2157, Online. Association for Computational Linguistics.

Kevin Clark, Minh-Thang Luong, Christopher D. Manning, and Quoc Le. 2018. Semi-supervised sequence modeling with cross-view training. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1914–1925, Brussels, Belgium. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, Minneapolis, Minnesota. Association for Computational Linguistics.

Xin Dong and Gerard de Melo. 2019. A robust self-learning framework for cross-lingual text classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6306–6310, Hong Kong, China. Association for Computational Linguistics.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2021. Making pre-trained language models better few-shot learners. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3816–3830, Online. Association for Computational Linguistics.

Ariel Gera, Alon Halfon, Eyal Shnarch, Yotam Perlitz, Liat Ein-Dor, and Noam Slonim. 2022. Zero-shot text classification with self-training. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Arushi Goel, Yunlong Jiao, and Jordan Massiah. 2022. Pars: Pseudo-label aware robust sample selection for learning with noisy labels. *arXiv preprint arXiv:2201.10836*.

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

Suchin Gururangan, Tam Dang, Dallas Card, and Noah A. Smith. 2019. Variational pretraining for semi-supervised text classification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5880–5894, Florence, Italy. Association for Computational Linguistics.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. 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. Association for Computational Linguistics.

Junxian He, Jiatao Gu, Jiajun Shen, and Marc'Aurelio Ranzato. 2020. Revisiting self-training for neural sequence generation. In *International Conference on Learning Representations*.

Mariya Hendriksen, Maurits Bleeker, Svitlana Vakulenko, Nanne van Noord, Ernst Kuiper, and Maarten de Rijke. 2022. Extending clip for category-to-image retrieval in e-commerce. In *Advances in Information Retrieval: 44th European Conference on IR Research, ECIR 2022, Stavanger, Norway, April 10–14, 2022, Proceedings, Part I*, page 289–303, Berlin, Heidelberg. Springer-Verlag.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. 2022. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*.

Zejiang Hou, Julian Salazar, and George Polovets. 2022. Meta-Learning the Difference: Preparing Large Language Models for Efficient Adaptation. *Transactions of the Association for Computational Linguistics*, 10:1249–1265.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339, Melbourne, Australia. Association for Computational Linguistics.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Chris Kedzie and Kathleen McKeown. 2019. A good sample is hard to find: Noise injection sampling and self-training for neural language generation models. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 584–593, Tokyo, Japan. Association for Computational Linguistics.

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*.

Samuli Laine and Timo Aila. 2017. Temporal ensembling for semi-supervised learning. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

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

Changchun Li, Ximing Li, and Jihong Ouyang. 2021a. Semi-supervised text classification with balanced deep representation distributions. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5044–5053, Online. Association for Computational Linguistics.

Shiyang Li, Semih Yavuz, Wenhu Chen, and Xifeng Yan. 2021b. Task-adaptive pre-training and self-training are complementary for natural language understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 1006–1015, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Shoushan Li, Zhongqing Wang, Guodong Zhou, and Sophia Yat Mei Lee. 2011. Semi-supervised learning for imbalanced sentiment classification. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Volume Three*, IJCAI'11, page 1826–1831. AAAI Press.

Zhenghua Li, Xue Peng, Min Zhang, Rui Wang, and Luo Si. 2019. Semi-supervised Domain Adaptation for Dependency Parsing. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2386–2395, Florence, Italy. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Lajanugen Logeswaran, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, Jacob Devlin, and Honglak Lee. 2019. Zero-shot entity linking by reading entity descriptions. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3449–3460, Florence, Italy. Association for Computational Linguistics.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, USA. Association for Computational Linguistics.

Katerina Margatina, Loic Barrault, and Nikolaos Aletras. 2022. On the importance of effectively adapting pretrained language models for active learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 825–836, Dublin, Ireland. Association for Computational Linguistics.

Julian McAuley and Jure Leskovec. 2013. Hidden factors and hidden topics: Understanding rating dimensions with review text. In *Proceedings of the 7th ACM Conference on Recommender Systems*, RecSys '13, page 165–172, New York, NY, USA. Association for Computing Machinery.

David McClosky, Eugene Charniak, and Mark Johnson. 2006. Effective self-training for parsing. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 152–159, New York City, USA. Association for Computational Linguistics.

Shikib Mehri, Mihail Eric, and Dilek Z. Hakkani-Tür. 2020. Dialoglue: A natural language understanding benchmark for task-oriented dialogue. *ArXiv*, abs/2009.13570.

Yu Meng, Yunyi Zhang, Jiaxin Huang, Chenyan Xiong, Heng Ji, Chao Zhang, and Jiawei Han. 2020. Text classification using label names only: A language model self-training approach. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9006–9017, Online. Association for Computational Linguistics.

Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. 2018. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 41:1979–1993.

Subhabrata Mukherjee and Ahmed Awadallah. 2020. Uncertainty-aware self-training for few-shot text classification. In *Advances in Neural Information Processing Systems*, volume 33, pages 21199–21212. Curran Associates, Inc.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of NAACL-HLT 2019: Demonstrations*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8).

Rajat Raina, Alexis Battle, Honglak Lee, Benjamin Packer, and Andrew Y. Ng. 2007. Self-taught learning: Transfer learning from unlabeled data. In *Proceedings of the 24th International Conference on Machine Learning*, ICML '07, page 759–766, New York, NY, USA. Association for Computing Machinery.

Alan Ramponi and Barbara Plank. 2020. Neural Unsupervised Domain Adaptation in NLP—A Survey. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6838–6855, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Sebastian Ruder and Barbara Plank. 2018. Strong Baselines for Neural Semi-Supervised Learning under Domain Shift. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1044–1054, Melbourne, Australia. Association for Computational Linguistics.

Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada. 2018. Maximum classifier discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3723–3732.

Timo Schick and Hinrich Schütze. 2021. Exploiting cloze-questions for few-shot text classification and natural language inference. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 255–269, Online. Association for Computational Linguistics.

Zhengxiang Shi, Yue Feng, and Aldo Lipani. 2022a. Learning to execute actions or ask clarification questions. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 2060–2070, Seattle, United States. Association for Computational Linguistics.

Zhengxiang Shi and Aldo Lipani. 2023. Don't stop pretraining? make prompt-based fine-tuning powerful learner. *arXiv preprint arXiv:2305.01711*.

Zhengxiang Shi, Qiang Zhang, and Aldo Lipani. 2022b. Stepgame: A new benchmark for robust multi-hop spatial reasoning in texts. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 11321–11329.

Kihyuk Sohn, David Berthelot, Chun-Liang Li, Zizhao Zhang, Nicholas Carlini, Ekin D. Cubuk, Alex Kurakin, Han Zhang, and Colin Raffel. 2020. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA. Curran Associates Inc.

Antti Tarvainen and Harri Valpola. 2017. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 1195–1204, Red Hook, NY, USA. Curran Associates Inc.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Ximei Wang, Jinghan Gao, Mingsheng Long, and Jianmin Wang. 2021. Self-tuning for data-efficient deep learning. In *International Conference on Machine Learning (ICML)*.

Yidong Wang, Hao Chen, Yue Fan, Wang SUN, Ran Tao, Wenxin Hou, Renjie Wang, Linyi Yang, Zhi Zhou, Lan-Zhe Guo, Heli Qi, Zhen Wu, Yu-Feng Li, Satoshi Nakamura, Wei Ye, Marios Savvides, Bhiksha Raj, Takahiro Shinozaki, Bernt Schiele, Jindong Wang, Xing Xie, and Yue Zhang. 2022. USB: A unified semi-supervised learning benchmark for classification. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Qizhe Xie, Zihang Dai, Eduard Hovy, Minh-Thang Luong, and Quoc V. Le. 2020a. Unsupervised data augmentation for consistency training. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA. Curran Associates Inc.

Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. 2020b. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10687–10698.

Qiantong Xu, Alexei Baevski, Tatiana Likhomanenko, Paden Tomasello, Alexis Conneau, Ronan Collobert, Gabriel Synnaeve, and Michael Auli. 2021a. Self-training and pre-training are complementary for speech recognition. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3030–3034. IEEE.

Yi Xu, Lei Shang, Jinxing Ye, Qi Qian, Yu-Feng Li, Baigui Sun, Hao Li, and Rong Jin. 2021b. Dash: Semi-supervised learning with dynamic thresholding. In *International Conference on Machine Learning*, pages 11525–11536. PMLR.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

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

Bowen Zhang, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. 2021. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, volume 34.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, page 649–657, Cambridge, MA, USA. MIT Press.

Barret Zoph, Golnaz Ghiasi, Tsung-Yi Lin, Yin Cui, Hanxiao Liu, Ekin D. Cubuk, and Quoc V. Le. 2020. Rethinking pre-training and self-training. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA. Curran Associates Inc.

Appendix Overview

The appendix is structured as follows:

Appendix §A provides a brief description and example for each dataset (subsection §A.1). Additionally, a similarity analysis among datasets and an illustration of overlaps between IMDB and AMAZON REVIEW are included (subsection §A.2).

Appendix §B presents a brief description of state-of-the-art ST approaches.

Appendix §C includes a supplementary Table that examines the effect of low unlabelled data sizes.

Appendix §D presents additional experiments to verify our findings using other ST approaches.

Appendix §E includes additional experiments to train ST approaches using TAPT checkpoints.

Appendix §F provides implementation details and hyperparameters for TAPT, ST, and FT methods used in our experiments.

A Datasets

In this section, we briefly introduce the datasets used in our work and provide additional analysis of the similarity among them. Specifically, we provide four examples to demonstrate the overlap between IMDB and AMAZON REVIEW, as a supplement to our domain adaptation analysis (§6).

A.1 Description

In this section, we briefly introduce IMDB, SST-2, AG NEWS, AMAZON REVIEW, and YAHOO! ANSWER datasets. Table 6 list examples for each dataset.

IMDB. The IMDB dataset (Maas et al., 2011) contains a collection of 50 000 reviews from the Internet Movie Database, with no more than 30 reviews per movie. This dataset contains an equal number of positive and negative reviews, yielding a 33% Marco-F_1 score for random guessing. There are 25 000 and 25 000 for training and testing, respectively. We follow Wang et al. (2022) to split the dataset by selecting 12 500 samples and 1 000 samples per class from the train set to form a train and validation set, respectively.

SST-2. The SST-2 dataset (Wang et al., 2018) consists of sentences from movie reviews and human annotations of their sentiment. The task is to predict the sentiment of a given sentence. Similar to IMDB, this is also a binary classification task. There are 67 349 and 872 for training and testing. We select 60 000 and 7 349 samples from the train set to form a train and validation set, respectively, where the validation set contains 3 675 and 3 674 samples for two classes, respectively.

AG NEWS. The AG NEWS topic classification dataset is constructed by Zhang et al. (2015), where 4 classes are used. Each class contains 30 000 training samples and 1 900 test samples. We follow Wang et al. (2022) to split the dataset by selecting 25 000 samples and 2 500 samples per class from the train set samples to form a train and validation set, respectively.

AMAZON REVIEW. The AMAZON REVIEW dataset (McAuley and Leskovec, 2013) is a sentiment classification dataset, with five classes. There are 600 000 train samples and 130 000 test samples per class. We follow Wang et al. (2022) to split the dataset by selecting 50 000 samples and 5 000 samples per class from the train set samples to form a train and validation set, respectively.

YAHOO! ANSWER. The YAHOO! ANSWER dataset (Chang et al., 2008) is a topic classification dataset, with ten classes. There are 140 000 train samples and 6 000 test samples per class. We follow Wang et al. (2022) to split the dataset by selecting 50 000 samples and 5 000 samples per class from the train set samples to form a train and validation set, respectively.

A.2 Dataset Similarity

We provide an analysis of the vocabulary overlap of the datasets, as shown in Figure 7. Additionally, in Table 7, we provide some examples to illustrate the overlap between IMDB and AMAZON REVIEW.

As shown in Table 6, although both the SST-2 and IMDB datasets are sentiment analysis tasks for movie reviews, the SST-2 datasets contain shorter and vaguer sentences than the IMDB dataset. This difference could be a potential reason for poor performance of ST approaches in the UDA setting (§6). In contrast, the AMAZON REVIEW dataset, which is a product review sentiment analysis dataset, is more similar to the IMDB dataset than the SST-2 dataset, as shown in Table 7. This suggests a poten-

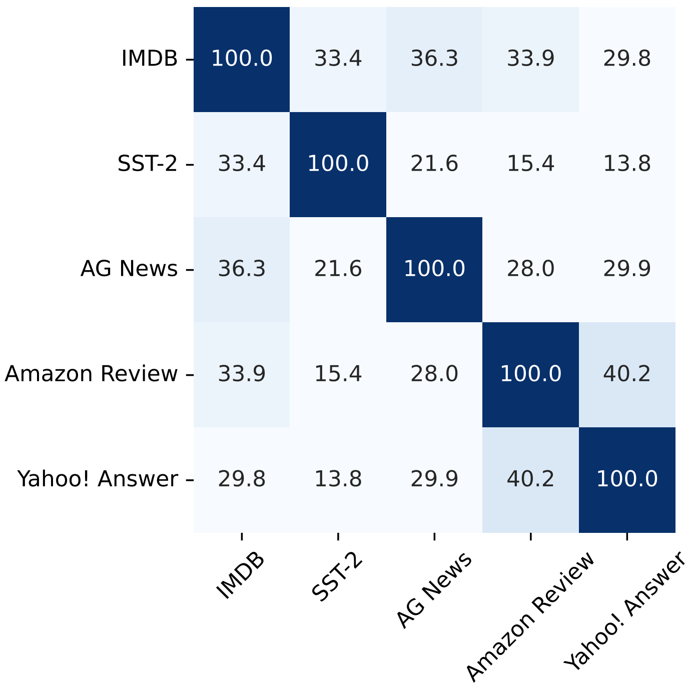

Figure 7: Vocabulary overlap (%) across datasets.

tial reason for the performance of ST and TAPT in the STL setting (§6).

B ST Frameworks

VAT. VAT (Miyato et al., 2018) proposed a regularization technique that forces pairs of data points that are very close in the input space to be close to each other in the output space. VAT adds small perturbation to the input data and forces the model to produce similar predictions.

FIXMATCH. FIXMATCH (Sohn et al., 2020) generates artificial labels using both consistency regularization and pseudo-labelling, where the artificial labels are produced based on weakly-augmented unlabelled data. These artificial labels are then used as targets to train the model on strongly-augmented unlabelled data. FIXMATCH only retains an artificial label if the model assigns a high probability to one of the possible classes.

DASH. DASH (Xu et al., 2021b) extends FIXMATCH by introducing a mechanism with a dynamically adjusted threshold of loss to select a subset of training examples from the unlabelled data for performing SSL.

FLEXMATCH. FLEXMATCH (Zhang et al., 2021) also extends FIXMATCH by introducing the concept of curriculum learning (Bengio et al., 2009) to flexibly adjust thresholds for different classes at each time step and select unlabelled data and their pseudo labels that are more likely to be informative.

ADAMATCH. ADAMATCH (Berthelot et al., 2022) aims to solve domain adaptation problems in SSL and build a high-accuracy model that trains on and tests on different data distributions. ADAMATCH builds on FIXMATCH and introduces a relative confidence threshold and a modified distribution alignment from (Berthelot et al., 2019a).

C Probability of performing worsen than SUPERVISED.

In §5, we discuss that we select the model performance with the three lowest unlabelled sizes (the first three columns in Figure 3) for each dataset and exclude the model performance with the lowest labelled size (the last row in Figure 3). This results in 9 cells in IMDB, 3 cells in SST-2, 9 cells in AMAZON REVIEW, and 12 cells in YAHOO! ANSWER, where TAPT has one run per cell and ST (FLEXMATCH and ADAMATCH) has two runs per cell. We consider a run to be a failure if its performance is worse than its corresponding SUPERVISED baseline.

Table 8 lists the probability of ST and TAPT of falling below the SUPERVISED baseline with selected combinations of labelled and unlabelled sizes.

D Further validations with other ST approaches

In this section, we conduct additional experiments on ST approaches, including VAT, DASH, and FIXMATCH to demonstrate that our findings are applicable to other ST approaches as well.

In Table 9, we select several combinations of labelled and unlabelled sizes on IMDB, SST-2, AMAZON REVIEW, and YAHOO! ANSWER datasets. Our experimental results show that other ST approaches do not perform well when the labelled size is low, and that other ST approaches have a high probability to perform worsen than SUPERVISED baselines when the unlabelled size is low. This suggests that poor performance when the labelled or unlabelled size is inadequate may be a common problem of state-of-the-art ST approaches.

E Train ST approaches with TAPT checkpoints

Previous works (Mukherjee and Awadallah, 2020; Li et al., 2021b) have suggested that training ST approaches from a TAPT checkpoint may be beneficial. Here we also provide some additional experiments to train ST approaches with TAPT checkpoints to further corroborate our findings.

Table 10 shows that TAPT outperforms ADAMATCH +TAPT or FLEXMATCH +TAPT with two different labelled sizes on the YAHOO! ANSWER dataset.

Table 11 shows that training ST approaches from TAPT checkpoints could improve the performance of ST but cannot solve the issue of ST approaches when labelled or unlabelled data is not adequate. Specifically, the performance of ST +TAPT is still poor when labelled data is insufficient, as discussed in §5. Meanwhile, in Table 11, the performance of ST +TAPT could be outperformed by the SUPERVISED baselines when unlabelled data is inadequate, while TAPT consistently outperforms the SUPERVISED baselines. When the labelled size is 10, the performance of ST trained with fewer unlabelled samples tends to be better, indicating that reducing the number of unlabelled data can be helpful, as discussed in §5.

F Implementation Details

We consistently use five random seeds, ranging from 1 to 5, for all algorithms. The sampled labelled data is the same for all algorithms for a given seed. The development and test sets remain unchanged for all different labelled and unlabelled data sizes.

Our model implementation uses open-source libraries including HuggingFace Transformers[2], Fairseq[3], and USB[4]. Our experiments of TAPT are performed on 8x32GB V100 GPUs, with a batch size of 16 per device and 2 gradient accumulation steps.

Table 12 lists the hyperparameters used for the TAPT phrase. Table 13 lists the hyperparameters used for the fine-tuning phrase. Table 14 lists the hyperparameters used for ST approaches.

[2] https://huggingface.co

[3] https://github.com/facebookresearch/fairseq

[4] https://github.com/microsoft/Semi-supervised-learning

Table 6: Examples for datasets.

Dataset	**Example**
IMDB	I watched this movie after seeing other comments on IMDb, even convincing my wife that it was a "unique horror movie." I wanted to like this movie, but was unable to.The "love story" was good, but the horror aspect was quite bad. If the story was just about a young man who fell in love with a girl suffering from parasomnia, then it would have been a better movie.The care centre stretched credulity well past the limits, in fact it was quite ridiculous. The doctor happily ignors privacy laws and professionalism. A nurse goes into a room for a routine feeding of a dangerous patient (without security escort), and drops the tray and runs out of the room screaming for no apparent reason. The forensic patient (and the film's villain) is tied up in a standing position fully clothed - apparently for years? None of it makes much sense.The movie even had some actors that I've liked in other things, such as the detectives, but still I can't recommend this movie.
SST-2	a rewarding work of art for only the most patient and challenge-hungry moviegoers.
AG News	Teen flies in plane #39;s landing gearA homeless teenager who hid in the landing gear of a passenger plane survived a 700-kilometre flight across south-western China but his companion fell and probably died, state media reported on Friday.
Amazon Review	THIS is MUSIC at its BESTRob Dougan has done it. He's crafted musical perfection, or close to it anyway. I have finally found the music I've been waiting for my whole life in this album - Rob D you are a genius. I think a lot of us wanted to know more about this guy as soon as we heard the track playing to the "Woman in the Red Dress" scene. Now I know why the Wachowski brothers have enlisted his musical talents to flesh out their movies.I know I should be trying to write a more helpful, objective review but I can do nothing but wax poetic for Rob Dougan and his debut album. He has mixed classical melodies with awesome electric beats and it all comes together in an audio orgy. Just buy the album already and let's get Rob some more mainstream recognition.
Yahoo! Answer	Does anybody know a great deal about angels? I'm looking for names, if they're good or bad, what they look like, etc. The more detail the better. All religions accepted

Table 7: Similarity analysis between IMDB and Amazon Review with four examples that highlight the overlap.

IMDB	**Amazon Review**
I loved this **movie** since I was 7 and I saw it on the opening day. It was so **touching** and beautiful. I strongly recommend seeing for all. It's a **movie** to watch with your family by far. My MPAA rating: PG-13 for thematic elements, prolonged scenes of disastor, nudity/sexuality and some language.	This is a very **touching**, spiritual **movie**! When I first saw this film, [...]. I was deeply moved by this motion picture, and the DVD brings the story to your own home. The bonus materials could be better, but the main part of the DVD is the actual **movie**. Great, great, great film... [...]
Pacino is over-the-top but to good effect as he's clearly having loads of **fun**. Beatty is **great** [...] The lighting, velvet overtones and smog/smoke combine to create a **great** effect.There are some really **funny** cameos [...] **Highly recommended**. 4.5/5 stars. [...]	Makes a **great** gift! We bought this book for my dad for Father's Day this year, and thought he would have **fun** reading it since he has four granddaughters. He loved it and has even selected stories to read to the girls during over-nights with Grandpa and Grandma. I **highly recommend** it as a **great** gift.
The late [...] scripted this tale of **terror** and it was absolutely one of the **scariest movies** I ever saw as a kid. (I had to walk MILES just to see a **movie**, and it was usually dark when I emerged from the theater; seeing a horror **movie** was always unnerving [...]	Movia ... please .... This **movie** is a masterpiece of **terror** & suspence & Beautifully filmed & acted.Comparisons to reality are not allowed when reviewing films of this caliber. Your reaction (though it MAY be **sarcastic**) is EXACT proof of it's genius! Watch it again...and this time....bask in all it's glory!
Fabulous actors, beautiful scenery, stark reality [...] I tried to buy the video for several years, finally bought it used from a video store that went out of business. But Yippee! The DVD is now for sale, I purchased it on amazon.com. Not cheap, but **well worth** it to me. [...]	**Well worth** the import price. My first impression of this album was a good one, but as time went on it came to grow on me more and more. This is certainly one of the better Costes albums. The mixing is nothing revolutionary, but it is well done and all tracks flow into each other very well. [...].

Table 8: Results on the effect of low unlabelled sizes on St and Tapt. Failure means performing worsen than Supervised.

Task	**#Unl.**	**#Lab.**	**Prob. of St Failure**	**Prob. of Tapt Failure**
IMDB	100, 500, 2k	10, 20, 200, 1k	6/18 (33%)	0/9 (0%)
SST-2	100, 500, 2k	40, 200, 1k, 5k	4/6 (67%)	0/3 (0%)
Amazon Review	1k, 5k, 20k	100, 500, 2k, 10k	10/18 (56%)	0/9 (0%)
Yahoo! Answer	1k, 5k, 20k	20, 100, 500, 2k, 10k	13/24 (54%)	0/12 (0%)

Table 9: We further verify our conclusion on VAT, DASH, FIXMATCH that . We report the average Macro-F_1 score on the test set across five seeds, with standard deviations as subscripts. Blue represents the best results for each row.

Dataset	#Unl.	#Lab.	VAT	FIXMATCH	DASH	FLEXMATCH	ADAMATCH	TAPT	SUPERVISED
IMDB	100	4	$33.5_{0.2}$	$33.4_{0.1}$	$33.4_{0.1}$	$35.7_{4.2}$	$34.1_{0.7}$	$61.8_{6.7}$	$59.4_{4.8}$
	100	10	$61.6_{20.1}$	$45.4_{21.6}$	$34.7_{2.2}$	$49.0_{19.9}$	$52.4_{21.0}$	$75.5_{6.9}$	$71.8_{8.5}$
	100	20	$87.1_{2.2}$	$64.6_{16.5}$	$67.8_{16.6}$	$85.5_{2.9}$	$79.1_{7.6}$	$85.5_{1.0}$	$84.1_{1.9}$
	500	4	$33.4_{0.0}$	$33.4_{0.1}$	$33.4_{0.1}$	$33.4_{0.1}$	$33.6_{0.3}$	$63.4_{7.2}$	$58.2_{7.1}$
	2k	4	$33.3_{0.0}$	$33.3_{0.0}$	$33.3_{0.0}$	$33.3_{0.0}$	$33.3_{0.0}$	$63.1_{6.2}$	$60.9_{5.6}$
	10k	4	$33.3_{0.0}$	$33.5_{0.3}$	$33.3_{0.0}$	$34.0_{1.2}$	$33.6_{0.4}$	$64.1_{8.9}$	$62.4_{7.9}$
	23k	4	$33.3_{0.0}$	$33.3_{0.0}$	$57.4_{29.4}$	$45.3_{23.9}$	$33.3_{0.0}$	$68.8_{5.6}$	$65.6_{10.4}$
SST-2	100	40	$63.3_{10.6}$	$46.9_{9.7}$	$47.9_{7.0}$	$57.2_{4.5}$	$51.0_{14.0}$	$78.7_{2.5}$	$76.4_{3.7}$
	500	40	$55.7_{16.8}$	$53.8_{8.9}$	$51.2_{10.0}$	$67.7_{10.7}$	$59.1_{11.4}$	$83.3_{4.8}$	$72.9_{7.9}$
	500	200	$83.0_{1.6}$	$84.5_{2.8}$	$82.6_{3.5}$	$83.8_{3.0}$	$87.4_{1.9}$	$88.8_{0.9}$	$88.3_{0.9}$
	2k	40	$55.9_{24.2}$	$36.4_{3.0}$	$35.3_{2.0}$	$56.6_{6.7}$	$49.3_{13.8}$	$79.3_{5.9}$	$71.7_{8.2}$
	10k	40	$73.5_{20.5}$	$38.9_{11.4}$	$35.6_{2.6}$	$56.9_{12.5}$	$36.2_{2.9}$	$85.9_{1.0}$	$78.5_{7.5}$
	60k	40	$79.6_{13.4}$	$32.6_{1.7}$	$33.4_{0.6}$	$40.6_{7.7}$	$42.6_{13.3}$	$82.6_{4.0}$	$75.3_{7.2}$
AMAZON REVIEW	1k	20	$13.5_{5.2}$	$14.9_{5.6}$	$20.3_{3.0}$	$25.8_{3.2}$	$20.7_{1.1}$	$32.0_{1.8}$	$32.5_{2.2}$
	1k	100	$46.1_{2.2}$	$36.3_{3.1}$	$35.3_{6.2}$	$43.4_{1.7}$	$40.3_{2.2}$	$48.5_{0.9}$	$48.2_{2.2}$
	1k	500	$52.6_{0.2}$	$50.8_{1.5}$	$49.5_{1.0}$	$54.1_{1.0}$	$52.8_{1.1}$	$55.9_{0.3}$	$55.3_{0.5}$
	5k	20	$15.5_{7.8}$	$13.5_{3.3}$	$22.2_{5.2}$	$23.2_{7.3}$	$16.9_{6.9}$	$32.8_{3.4}$	$32.3_{2.5}$
	20k	20	$19.3_{7.5}$	$15.2_{3.9}$	$20.5_{6.4}$	$19.1_{10.0}$	$19.3_{6.3}$	$32.0_{3.2}$	$31.6_{3.6}$
	100k	20	$14.1_{7.3}$	$11.9_{2.9}$	$20.7_{5.2}$	$15.3_{2.6}$	$12.5_{3.7}$	$30.7_{3.6}$	$30.8_{3.9}$
	250k	20	$10.3_{5.0}$	$10.9_{3.6}$	$22.0_{5.7}$	$22.7_{4.9}$	$14.4_{5.6}$	$30.2_{2.4}$	$32.1_{3.1}$
YAHOO! ANSWER	1k	10	$1.9_{0.1}$	$2.0_{0.1}$	$4.6_{2.9}$	$15.7_{2.6}$	$18.8_{7.9}$	$29.6_{5.8}$	$23.5_{4.5}$
	1k	20	$6.7_{2.8}$	$10.1_{4.2}$	$9.6_{3.2}$	$32.7_{9.1}$	$28.8_{5.8}$	$38.9_{4.1}$	$34.1_{3.6}$
	1k	100	$55.2_{1.7}$	$46.9_{4.4}$	$45.3_{3.7}$	$54.2_{1.4}$	$53.9_{1.3}$	$59.7_{0.8}$	$57.4_{1.6}$
	1k	500	$59.2_{0.4}$	$61.6_{0.6}$	$60.7_{1.3}$	$61.9_{1.1}$	$61.5_{0.9}$	$65.8_{0.3}$	$65.5_{0.2}$
	5k	10	$1.8_{0.0}$	$3.2_{2.6}$	$3.7_{2.7}$	$16.4_{10.8}$	$17.8_{11.7}$	$31.4_{5.1}$	$25.7_{3.9}$
	20k	10	$2.4_{0.9}$	$2.0_{0.3}$	$4.9_{3.1}$	$7.3_{4.7}$	$25.2_{12.2}$	$32.4_{5.6}$	$27.2_{4.4}$
	100k	10	$2.3_{0.6}$	$3.8_{2.5}$	$3.4_{2.9}$	$2.9_{1.1}$	$17.7_{11.4}$	$30.8_{3.8}$	$28.0_{5.0}$
	500k	10	$2.0_{0.4}$	$1.8_{0.0}$	$2.6_{1.2}$	$2.5_{0.9}$	$14.3_{6.0}$	$27.3_{4.6}$	$24.7_{4.8}$

Table 10: Results of ADAMATCH +TAPT and FLEXMATCH +TAPT on YAHOO! ANSWER with two different labelled sizes.

	YAHOO! ANSWER	
	500	2000
ADAMATCH	$68.0_{0.7}$	$69.5_{0.3}$
+ TAPT	$68.2_{1.0}$	$69.8_{0.3}$
FLEXMATCH	$66.6_{0.7}$	$68.7_{0.4}$
+ TAPT	$66.7_{1.2}$	$69.0_{0.5}$
SUPERVISED	$65.4_{0.3}$	$68.5_{0.3}$
+ TAPT	$68.8_{0.7}$	$71.5_{0.3}$
FULLY-SUPERVISED.	$75.3_{0.2}$	
+ TAPT	$75.4_{0.1}$	

Table 11: We further verify our conclusion on FLEXMATCH +TAPT. We report the average Macro-F_1 score on the test set across five seeds, with standard deviations as subscripts. Blue represents the best results for each row.

Dataset	#Unl.	#Lab.	FLEXMATCH + TAPT	FLEXMATCH	TAPT	SUPERVISED
YAHOO! ANSWER	1k	10	$17.0_{4.9}$	$15.7_{2.6}$	$29.6_{5.8}$	$23.5_{4.5}$
	1k	20	$39.4_{2.0}$	$32.7_{9.1}$	$38.9_{4.1}$	$34.1_{3.6}$
	1k	100	$55.2_{1.8}$	$54.2_{1.4}$	$59.7_{0.8}$	$57.4_{1.6}$
	1k	500	$62.0_{0.7}$	$61.9_{1.1}$	$65.8_{0.3}$	$65.5_{0.2}$
	20k	10	$4.0_{1.4}$	$7.3_{4.7}$	$32.4_{5.6}$	$27.2_{4.4}$
	100k	10	$5.1_{6.1}$	$2.9_{1.1}$	$30.8_{3.8}$	$28.0_{5.0}$
	500k	10	$2.5_{1.1}$	$2.5_{0.9}$	$27.3_{4.6}$	$24.7_{4.8}$

Hyperparameter	**Assignment**
number of steps	100 epochs
batch size	256
maximum learning rate	1e-06, 1e-4
learning rate optimizer	AdamW
Adam epsilon	1e-6
Adam beta weights	0.9, 0.98
learning rate scheduler	Warmup linear
Weight decay	0.01
Warmup proportion	0.06
learning rate decay	linear

Table 12: Hyperparameters for task-adaptive pretraining. The learning rate and unlabelled size are tightly connected and need to be adjusted together. We generally recommend increasing the learning rate as you increase the unlabelled size. Different from its predecessor, BERT (Devlin et al., 2019), where the next sentence prediction objective is used, RoBERTa (Liu et al., 2019) is only trained with the MLM objective (i.e., cross-entropy loss on predicting randomly masked tokens), dynamically changing the masking pattern applied to the training examples and typically using the masking probability of 0.15.

Hyperparameter	**Assignment**
number of steps	10 or 50 epochs
batch size	16 or 32
maximum learning rate	2e-05
learning rate optimizer	AdamW
maximum sequence length	256
learning rate scheduler	Warmup linear
Warmup proportion	0.06
learning rate decay	linear

Table 13: Hyperparameters for fine-tuning. More epochs are used when the labelled size is low.

Hyperparameter	**Assignment**
number of steps	25 600 or 51 200 steps
batch size	16
maximum learning rate	2e-05
learning rate optimizer	AdamW
maximum sequence length	256
learning rate scheduler	Warmup linear
Warmup proportion	0.05
learning rate decay	linear

Table 14: Hyperparameters for self training. Algorithm-specific hyperparameters will be released in configuration files with the code.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Right before the reference section.

□ A2. Did you discuss any potential risks of your work?
Not applicable. Left blank.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Abstract and Section 1

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☒ Did you use or create scientific artifacts?

Left blank.

□ B1. Did you cite the creators of artifacts you used?
Not applicable. Left blank.

□ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Not applicable. Left blank.

□ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Not applicable. Left blank.

□ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Not applicable. Left blank.

□ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Not applicable. Left blank.

□ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Not applicable. Left blank.

C ☑ Did you run computational experiments?

Section 4, 5, 6

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Appendix F

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Appendix F

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Section 4, 5, 6 and Appendix C, D, E

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Appendix F

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

□ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
Not applicable. Left blank.

□ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
Not applicable. Left blank.

□ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
Not applicable. Left blank.

□ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Not applicable. Left blank.

□ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Not applicable. Left blank.