Distill or Annotate?
Cost-Efficient Fine-Tuning of Compact Models

Junmo Kang, Wei Xu, Alan Ritter
Georgia Institute of Technology
`junmo.kang@gatech.edu`
`{wei.xu, alan.ritter}@cc.gatech.edu`

Abstract

Fine-tuning large models is highly effective, however, inference can be expensive and produces carbon emissions. Knowledge distillation has been shown to be a practical solution to reduce inference costs, but the distillation process itself requires significant computational resources. Rather than buying or renting GPUs to fine-tune, then distill a large model, an NLP practitioner might instead choose to allocate the available budget to hire annotators and manually label additional fine-tuning data. In this paper, we investigate how to most efficiently use a fixed budget to build a compact model. Through extensive experiments on six diverse tasks, we show that distilling from `T5-XXL` (11B) to `T5-Small` (60M) is almost always a cost-efficient strategy compared to annotating more data to directly train a compact model (`T5-Small`). We further investigate how the optimal budget allocated towards computation varies across scenarios. We will make our code, datasets, annotation cost estimates, and baseline models available as a benchmark to support further work on cost-efficient training of compact models.

1 Introduction

Increasing the size of pre-trained models can consistently improve performance on downstream tasks after fine-tuning, as seen in studies based on BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019), BART (Lewis et al., 2019), T5 (Raffel et al., 2020), and the work on empirical scaling laws (Brown et al., 2020; Lester et al., 2021; Hernandez et al., 2021). However, using large models for inference is expensive and contributes to carbon emissions (Patterson et al., 2021). To address this, researchers have explored methods to compress large models through techniques such as knowledge distillation (Hinton et al., 2015; Sanh et al., 2019; Gou et al., 2021), which is effective in reducing inference costs (Magister et al., 2022) and improving the generalization of smaller student models (Stanton et al., 2021). Nonetheless, the distillation process itself still requires significant computational, memory, and storage resources (Xia et al., 2022).

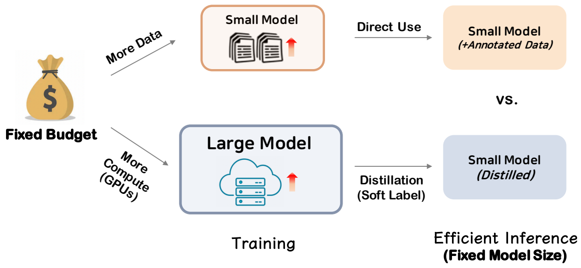

Figure 1: An illustration of two practical strategies to build a compact fixed-size model. Given a fixed budget and a small amount of initially annotated data, (i) one can annotate more data to directly fine-tune a small model. (ii) Alternatively, one may leverage a larger model with more computational resources to distill its knowledge into a small model for efficient inference.

In addition to compressing models, an alternative approach to improve performance without increasing inference costs is to simply label additional data for fine-tuning. Recent work has shown that a few hundred extra labels can sometimes lead to better performance than billions of additional model parameters (Kirstain et al., 2022). This raises the question of how to most efficiently use a fixed budget to train a compact model which supports efficient inference while maximizing performance. One option is to use an available budget to hire annotators to label additional data and directly fine-tune a small model. Alternatively, the budget could be used to purchase or rent GPUs to fine-tune and distill a large teacher model (see Figure 1).

In this paper, we use the theory of consumer choice (Becker, 1965; Lancaster, 1966; Bai et al., 2021) to investigate the question of when distillation is a cost-efficient strategy for model compression. Based on extensive empirical analysis,

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 11100–11119
July 9-14, 2023 ©2023 Association for Computational Linguistics

Dataset	Task	#Train	$/Label	Total $
WLP (Tabassum et al., 2020)	Named Entity Recognition	11,966	$0.260	$3,111
STANCEOSAURUS (Zheng et al., 2022)	Stance Classification	12,130	$0.364	$4,415
FEVER (Thorne et al., 2018)	Fact Verification	104,966	$0.129	$13,544
MULTIPIT$_{\texttt{Id}}$ (Dou et al., 2022)	Paraphrase Identification	92,217	$0.200	$18,443
MULTIPIT$_{\texttt{Gen}}$ (Dou et al., 2022)	Paraphrase Generation	49,673	$0.371	$18,443
NATURAL QUESTIONS (Kwiatkowski et al., 2019)	Question Answering	87,372	$0.129	$11,271

Table 1: Data annotation costs for various NLP datasets/tasks.

we provide recommendations on how to allocate a fixed budget for human annotation and computing resources to train a compact model. Our experiments across six NLP tasks reveal that distillation with unlabeled data is almost always a cost-efficient strategy for improving the performance of compact models when compared to annotation (see Table 2). Furthermore, our analysis shows that the optimal allocation of budget towards distillation increases as more labeled data becomes available (see §4.1 and Figure 2). For smaller budgets, it is Pareto optimal (Abdolrashidi et al., 2021; Treviso et al., 2022) to use smaller amounts of unlabeled data for distillation, while increasing the amount of labeled data, as this leads to a more knowledgeable teacher. As the budget increases, it becomes economical to distill using larger unlabeled datasets, because the teacher model outperforms the student by a significant margin. Finally, we investigate the cost efficiency of data annotation with `GPT-3.5` (Ouyang et al., 2022) (Figure 6). We find that, although `GPT-3.5` is cheaper than human annotators, fine-tuning `T5-XXL` and then distilling a small model is more cost-efficient than directly fine-tuning the small model with pseudo-labels from `GPT-3.5`.

We will make our code, datasets, annotation cost estimates, and baseline models available as a benchmark to support further work on cost-efficient training of compact models.

2 Study Design

In this section, we first describe how we formulate the problem for the cost-efficiency study (§2.1). We then compare two strategies (§2.2 & 2.3) for building compact models that incur different proportions of computational and human annotation costs. Finally, we explain how to estimate the annotation cost (§2.4) and computational cost (§2.5) involved in the two strategies.

2.1 Problem Formulation and Assumptions

The main focus of this study is to fairly evaluate the two approaches (§2.2 & §2.3) under a fixed budget. When financial constraints are in place, practitioners may be faced with weighing options of allocating money towards *data* or *compute*; we empirically investigate their trade-offs to maximize the resulting utility. To enable extensive studies, we simulate the process of labeling data using a variety of existing crowdsourced datasets, and the cloud GPU rentals that charge per hour of use.

We assume the NLP engineer's salary is a fixed cost, so their time spent building models and/or managing a group of annotators to label data are not a factor in determining the total cost. The only costs considered are the direct costs for human data labeling and GPU computation. No task-specific labeled data is initially assumed to be available for free, but we do assume that pre-trained models such as `T5` (Raffel et al., 2020), which are publicly available, have zero cost.

2.2 Strategy 1: Building a Compact Model Directly with Annotations (`Ann.`)

This strategy directly fine-tunes a compact model (e.g., `T5-Small` (60M)), allocating the entire budget towards human annotation. This is considered the most straightforward approach practitioners would choose to train a compact model.

In particular, given a budget constraint, we prepare data that can be maximally annotated using the budget, and we train `T5` (Raffel et al., 2020) on the data under a unified text-to-text framework for all tasks (Table 1), maximizing the likelihood of a target text Y given an input text X. The format for an input X and the corresponding target Y for each task is detailed in Appendix B.

Note that the most dominant cost associated with this strategy is the annotation cost. While the total cost of building this direct model can include the fine-tuning cost (i.e., computational cost), we found it negligible in most cases and thus omitted it, unless otherwise noted, for the sake of simplicity.[1]

2.3 Strategy 2: Distilling from a Larger Model (Dist.)

As an alternative to annotating more data, one could allocate part of the budget towards computation to train a larger (e.g., T5-XXL (11B)) model on a smaller amount of data. The large model can then be distilled to produce a final compact model that also supports efficient inference.

Following recent work (Xia et al., 2022; Zhou et al., 2022b), our study mostly focuses on task-specific model compression rather than general distillation (Sanh et al., 2019),[2] however we provide analysis of general vs. task-specific distillation in Appendix F. General distillation requires significant computational resources; also task-specific and general distillation can be used together in a complementary fashion (Jiao et al., 2020).

Notably, even for Strategy 2, annotated data is needed to train the large teacher model. Therefore, we assume to have a certain number (N) of data initially annotated by spending some part of the budget, and fine-tune the larger model using this data in the same way as in §2.2. After that, a small model (i.e., student) is trained by distilling the larger model's (i.e., teacher) knowledge (Hinton et al., 2015), in which the teacher's probability distributions over a target sequence given a source input are used as soft labels. We adopt KL divergence loss, which compares two distributions, to make the student's distribution P_S follow the teacher's output distribution P_T with respect to task-specific unlabeled data[3]:

$$D_{KL}(P_T||P_S) = \sum_{v \in V} P_T(v) \log \frac{P_T(v)}{P_S(v)} \quad (1)$$

where V is vocabulary space. Input and target tokens that are conditioned to produce probabilities are omitted above for brevity.

The total cost includes both the initial cost for N (the number of initially annotated training examples) and the computational cost for fine-tuning a large model and then distilling it into a compact model.

2.4 Cost Estimation for Data Annotation

This study considers six diverse and practical NLP tasks, shown in Table 1. We estimate the annotation cost for each dataset based on mentions in the corresponding literature if available, correspondence with creators of the dataset, or prices of the Data Labeling Service from Google Cloud, following Wang et al. (2021)[4]. Detailed descriptions of our cost estimates for each dataset are provided in Appendix A.

2.5 Estimation of Computational Cost

This work assumes that computing resources are rented from Google Cloud for model training. We specifically consider NVIDIA A100 GPUs, each equipped with 40GB of VRAM, to fit a large model (e.g., 11B parameters) into them. The price of this, which includes a virtual machine and storage, is set at about $3.75 per 1 GPU hour. For extensive studies, we exploit our own resources, A40 GPUs that have been shown to be approximately 2x slower than A100 through benchmark results[5] as well as our preliminary experiment that compares the training time. As a result, we estimate the computational cost as $1.875 per 1 GPU hour. This is a realistic price that practitioners would need to pay, unlike theoretical measures such as FLOPs, which do not reflect the real runtime (Xu and McAuley, 2022) and costs. An example breakdown of cost estimates for building compact models is provided in Appendix (Table 6).

3 Evaluating Annotation and Distillation under a Fixed Budget

In Table 2, we evaluate the two strategies under varying budgets for six different tasks. We first set N, the number of starting data annotated by spending an initial $. Given a fixed budget, we then either *annotate more data* for the annotation (Ann.) strategy, or use more *GPU hours* along with more *unlabeled data* for the distillation (Dist.) strategy.

We consider T5-Small (60M) as a compact model and T5-XXL (11B) as a teacher model for our main study. All models are fine-tuned based

[1] Fine-tuning T5-Small (60M) on 5K data, for example, takes less than half an hour, which costs approximately $1, based on the computational cost in §2.5.

[2] In general distillation, a pre-trained model is distilled before fine-tuning, such as DistillBERT.

[3] For example, source sentences without target paraphrased sentences for a paraphrase generation task. Refer to Appendix D for details of the unlabeled data.

[4] https://cloud.google.com/ai-platform/data-labeling/pricing#labeling_costs

[5] https://lambdalabs.com/blog/nvidia-rtx-a40-benchmarks

Task	*N* (Initial $)	Strategy	Additional $				
			Ann. Performance (*#Additional Data*) Dist. Performance (*GPU Hours / #Unlabeled Data*)				
WLP			+$0	+$100	+$200	+$300	+$500
	1K ($260)	T5-Small (Ann.)	**40.7** (*+0*)	50.0 (*+384*)	53.7 (*+769*)	57.8 (*+1153*)	62.7 (*+1923*)
		T5-XXL [**72.4**] ⇒ T5-Small (Dist.)	N/A	**71.1** (*54h / 19K*)	**71.3** (*107h / 42K*)	**70.9** (*160h / 65K*)	**70.8** (*267h / 111K*)
	5K ($1300)	T5-Small (Ann.)	**67.4** (*+0*)	**68.2** (*+384*)	68.6 (*+769*)	68.7 (*+1153*)	69.3 (*+1923*)
		T5-XXL [**74.2**] ⇒ T5-Small (Dist.)	N/A	65.3 (*54h / 7K*)	**71.8** (*107h / 30K*)	**72.4** (*160h / 53K*)	**72.5** (*267h / 99K*)
STANCEOSAURUS			+$0	+$100	+$150	+$200	+$300
	1K ($364)	T5-Small (Ann.)	**37.5** (*+0*)	45.4 (*+274*)	45.5 (*+412*)	45.5 (*+549*)	44.7 (*+824*)
		T5-XXL [**62.5**] ⇒ T5-Small (Dist.)	N/A	**54.2** (*54h / 37K*)	**54.6** (*80h / 60K*)	**56.3** (*107h / 82K*)	**56.9** (*160h / 126K*)
	5K ($1820)	T5-Small (Ann.)	**49.4** (*+0*)	50.7 (*+274*)	52.6 (*+412*)	49.1 (*+549*)	50.3 (*+824*)
		T5-XXL [**69.6**] ⇒ T5-Small (Dist.)	N/A	**52.4** (*54h / 17K*)	**55.4** (*80h / 40K*)	**56.2** (*107h / 62K*)	**60.5** (*160h / 106K*)
FEVER			+$0	+$50	+$75	+$100	+$150
	1K ($129)	T5-Small (Ann.)	**49.7** (*+0*)	49.7 (*+387*)	49.7 (*+581*)	49.7 (*+775*)	49.8 (*+1162*)
		T5-XXL [**73.5**] ⇒ T5-Small (Dist.)	N/A	**71.3** (*27h / 54K*)	**71.1** (*40h / 86K*)	**71.6** (*54h / 118K*)	**71.7** (*80h / 182K*)
	5K ($645)	T5-Small (Ann.)	**67.2** (*+0*)	68.2 (*+387*)	68.1 (*+581*)	68.1 (*+775*)	68.9 (*+1162*)
		T5-XXL [**78.0**] ⇒ T5-Small (Dist.)	N/A	**73.4** (*27h / 35K*)	**74.1** (*40h / 67K*)	**74.3** (*54h / 99K*)	**74.8** (*80h / 163K*)
MULTIPIT$_{\text{Id}}$			+$0	+$100	+$150	+$200	+$300
	1K ($200)	T5-Small (Ann.)	**53.0** (*+0*)	53.1 (*+500*)	53.1 (*+750*)	54.6 (*+1000*)	54.2 (*+1500*)
		T5-XXL [**79.9**] ⇒ T5-Small (Dist.)	N/A	**79.1** (*54h / 75K*)	**78.3** (*80h / 115K*)	**78.8** (*107h / 156K*)	**77.9** (*160h / 237K*)
	5K ($1000)	T5-Small (Ann.)	**78.0** (*+0*)	77.4 (*+500*)	77.0 (*+750*)	78.1 (*+1000*)	77.8 (*+1500*)
		T5-XXL [**84.5**] ⇒ T5-Small (Dist.)	N/A	**80.6** (*54h / 54K*)	**80.5** (*80h / 95K*)	**81.1** (*107h / 136K*)	**81.9** (*160h / 217K*)
MULTIPIT$_{\text{Gen}}$			+$0	+$100	+$150	+$200	+$300
	1K ($371)	T5-Small (Ann.)	**56.8** (*+0*)	57.7 (*+269*)	58.9 (*+404*)	59.2 (*+539*)	59.3 (*+808*)
		T5-XXL [**67.4**] ⇒ T5-Small (Dist.)	N/A	**60.3** (*54h / 56K*)	**62.1** (*80h / 87K*)	**62.0** (*107h / 118K*)	**62.6** (*160h / 179K*)
	10K ($3710)	T5-Small (Ann.)	**68.6** (*+0*)	**68.6** (*+269*)	68.6 (*+404*)	68.6 (*+539*)	68.7 (*+808*)
		T5-XXL [**74.8**] ⇒ T5-Small (Dist.)	N/A	68.4 (*54h / 10K*)	**72.1** (*80h / 41K*)	**73.7** (*107h / 72K*)	**74.0** (*160h / 133K*)
NATURAL QUESTIONS			+$0	+$50	+$75	+$100	+$150
	1K ($129)	T5-Small (Ann.)	**3.5** (*+0*)	4.1 (*+387*)	4.2 (*+581*)	4.5 (*+775*)	5.0 (*+1162*)
		T5-XXL [21.9] ⇒ T5-Small (Dist.)	N/A	**11.3** (*27h / 34K*)	**11.8** (*40h / 54K*)	**13.0** (*54h / 75K*)	**13.5** (*80h / 115K*)
	10K ($1290)	T5-Small (Ann.)	**9.8** (*+0*)	**10.2** (*+387*)	9.9 (*+581*)	10.4 (*+775*)	10.3 (*+1162*)
		T5-XXL [**26.1**] ⇒ T5-Small (Dist.)	N/A	N/A	**12.0** (*40h / 17K*)	**16.3** (*54h / 46K*)	**18.0** (*80h / 104K*)

Table 2: Main results of the cost efficiency of a small model with more data annotation (Ann.) and teacher [**performance**] ⇒ student distillation (Dist.) on various NLP tasks. *N* indicates the number of starting data annotated with the corresponding (initial $). (*#Additional Data*) refers to the number of annotated data additional to *N*, and (*GPU Hours*) denotes the total GPU hours for fine-tuning the teacher model on *N* data, plus for the distillation into a small model using varied (*#Unlabeled Data*). N/A is used when it is not feasible to build a model given the cost.

on T5 v1.1 (Roberts et al., 2020), which was pretrained in an unsupervised way only, unlike the original T5 (Raffel et al., 2020).

In the case of FEVER and NATURAL QUESTIONS, following Lee et al. (2020) and Roberts et al. (2020) respectively, we consider a closed-book setting where models should rely solely on its parametric knowledge, and report performances on dev sets as test sets are private. To measure performances, we use accuracy for FEVER and MULTIPIT$_{\text{Id}}$, F1 for WLP, STANCEOSAURUS, and NATURAL QUESTIONS, and BERT-iBLEU (Niu et al., 2021) (i.e., the harmonic mean of self-BLEU and BERTSCORE (Zhang et al., 2020)) for MULTIPIT$_{\text{Gen}}$. More details about experimental settings are described in Appendix C.

3.1 Annotation vs. Distillation

In Table 2, we observe that interestingly, the distillation (Dist.) strategy significantly outperforms the annotation (Ann). strategy across almost all cases for all tasks. While knowledge distillation (Hinton et al., 2015) has been proven effective for compression/generalization in previous works (Sanh et al., 2019; Kang et al., 2020; Le et al., 2022), our result that takes into account the realistic costs involved in building models is quite surprising, which highlights a new aspect: it is economically efficient. In other words, this suggests that exclusive reliance on scaling data by hiring human annotators might not be a good practice in light of cost efficiency.

Note that Dist. needs to be first fine-tuned on *N* labeled data that requires a considerable computational cost, so if the fine-tuning cost exceeds the given budget, we denote such cases as N/A. In such scenarios, Ann. is essentially the right choice. We also notice some scenarios where Ann. is a better option with limited budgets. For example, Ann. defeats its counterpart with $100 for WLP

Model (Teacher ⇒ Student)	WLP	Stanceosaurus	FEVER	MultiPIT$_{\text{Id}}$	MultiPIT$_{\text{Gen}}$	Natural Questions
`T5-Small ⇒ T5-Small (Self-Dist.)`	65.2 [67.4]	50.3 [50.5]	67.6 [67.2]	77.1 [78.0]	66.1 [68.1]	3.8 [9.8]
`T5-XXL ⇒ T5-Small (Dist.)`	70.6 [74.2]	58.9 [69.6]	74.2 [78.0]	80.9 [84.5]	73.8 [74.8]	17.8 [26.1]

Table 3: Results of self-distillation and distillation with the same amount of unlabeled data (*100K*). Numbers in [ ] represent the performances of the teacher models that are trained on *5K* annotated data.

Model	WLP		Stanceosaurus		FEVER		MultiPIT$_{\text{Id}}$		MultiPIT$_{\text{Gen}}$		Natural Questions	
`T5-XXL ⇒ T5-Small (Dist.)`	70.6	($502)	**58.9**	($279)	74.2	($101)	80.9	($161)	**73.8**	($245)	17.8	($148)
`T5-Small (Ann.)`	70.5	($1,300)	N/A		74.0	($1,032)	81.0	($1,980)	N/A		17.8	($3,321)
`T5-Small (Ann.)` - Upper Bound	**71.1**	($1,800)	53.0	($2,595)	**76.9**	($12,899)	**87.5**	($17,443)	69.3	($14,469)	**26.2**	($9,981)

Table 4: Performances along with (the corresponding budget) of `Dist.`, `Ann.` that performs the same/similar to `Dist.`, and `Ann.` upper bound by leveraging all existing annotated data. The best performance for each task is in bold.

(*N=5K*) and MultiPIT$_{\text{Gen}}$ (*N=10K*). In these cases, the *#unlabeled data* used for distillation are highly limited (*7K* and *10K* , respectively) as fine-tuning costs make up a substantial portion of limited budgets.

3.2 Does Distillation Work Better Simply by Making Use of Unlabeled Data?

In Table 2, we observe a substantial performance gap between `Ann.` and `Dist.` One notable point is that there is a big difference in the absolute number of data (*#labeled data* and *#unlabeled data*) used for each strategy given a fixed budget. In Table 2, for instance in WLP, given `$500`, *1923* more data can be annotated for `Ann.`, whereas *111K* unlabeled data can be leveraged for `Dist.` This not only means that annotated data is expensive, but also raises a question: *is the performance gap simply because of the difference in the number of data points?* To investigate this question by building a fair ground in terms of the size of data, we take a `self-distillation (Self-Dist.)` approach (Zhang et al., 2019) in which the architecture of a teacher and a student is the same (i.e., `T5-Small`).

In Table 3, we compare `Dist.` against `Self-Dist.` using the same *100K* unlabeled data. We see that `Self-Dist.` is worse than the `Dist.` across all tasks by remarkable margins even though the same number of data is used. In fact, the performance of `Self-Dist.` is found to be bounded by its teacher (i.e., `T5-Small (Ann.)`), as also observed in (Zhou et al., 2022a). This analysis suggests that the performance gap between `Dist.` and `Ann.` can indeed be attributed to exploiting the large pre-trained language model's capability, not simply making use of more data.

3.3 Comparison under Larger Budgets

Our experiments suggest that `distillation (Dist.)` is a more economical choice than relying completely on the human annotation to train a compact model, at least within scenarios presented in Table 2. However, this raises a question: *could* `Ann.` *reach the performance of* `Dist.` *when investing a much larger budget?* Table 4 shows the results of `Dist.` with budgets for *100K* unlabeled data, and `Ann.` with much larger budgets (or upper bound by using all available *#labeled data*). Interestingly, in some cases (Stanceosaurus & MultiPIT$_{\text{Gen}}$), `Dist.` turns out to be an astoundingly economically efficient way to train a compact model. Even though all existing annotated data (*∼50K*) are used for MultiPIT$_{\text{Gen}}$ training (w/ `$14,469`), it never outperforms `Dist.` (w/ only `$245`). For other tasks except for the aforementioned ones, we notice that `Ann.` can outperform `Dist.` with much larger budgets (e.g., `$12,899` for FEVER). In practice, however, we still find that `Ann.` can be much more costly (e.g. 10x in the case of FEVER) to obtain similar performance.

4 Further Analyses

In this section, we study varied values of each variable: the initial number (N) of annotated data (§4.1), the compact model size (§4.2), and the teacher model size (§4.3), all of which are fixed in the main experiment (§3.1).

4.1 Pareto Curves

In Figure 2, we explore different combinations of *#labeled data* (`L={0.1K, 0.5K, 1K, 5K, 10K}`) and *#unlabeled data* (`U={ 0 , 10K , 100K }`). Note that `U=0` indicates the `annotation (Ann.)` strategy in essence. We plot the performances of each combination and approximate the Pareto frontier

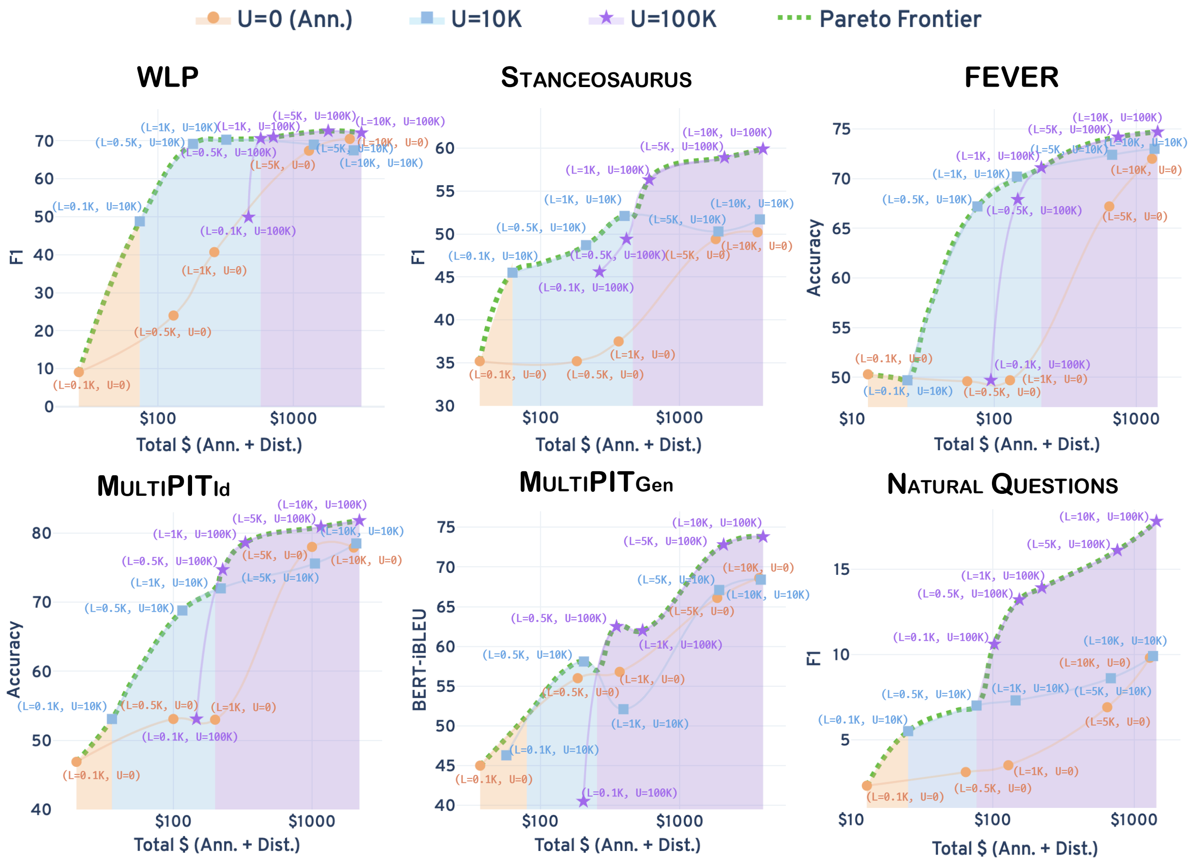

Figure 2: Pareto curves with various combinations of *#labeled data* (L={0.1K, 0.5K, 1K, 5K, 10K}) and *#unlabeled data* (U={ 0 , 10K , 100K }). U=0 denotes the annotation (Ann.) strategy. The Pareto frontier (····) is the set of optimal solutions that practitioners would choose from, and is approximated by interpolating the given data points. The X-axis is on a logarithmic scale.

(Abdolrashidi et al., 2021; Treviso et al., 2022) by interpolating the given data points. For all tasks, we observe that the distillation (Dist.) strategy is almost always Pareto optimal.[6] In Appendix (Table 11), we also look at the low resource setting in detail.

Furthermore, we observe that using a smaller amount of unlabeled data (U=10K) is Pareto optimal for smaller budgets, while larger unlabeled data (U=100K) maximizes utility as the budget increases. This implies that in low-budget settings, the teacher's capacity is limited, allowing the student to catch up quickly. However, once the teacher outperforms the student by a significant margin, it is more economical to allocate a larger part of the budget towards distillation.

In Figure 3, we provide an additional analysis by varying the number of initially annotated data (N) under fixed budgets to look at the impact of N. Expectedly, we notice that Dist. outperforms Ann. in general except for some cases with low N,

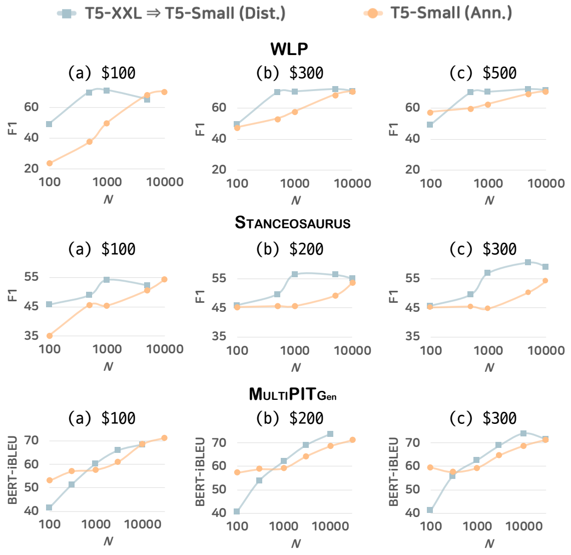

Figure 3: Results according to different number of starting annotated data (N) under fixed additional budgets.

[6] One exception is (L=0.1K, U=0) where a budget is so limited that leveraging a large model is barely feasible.

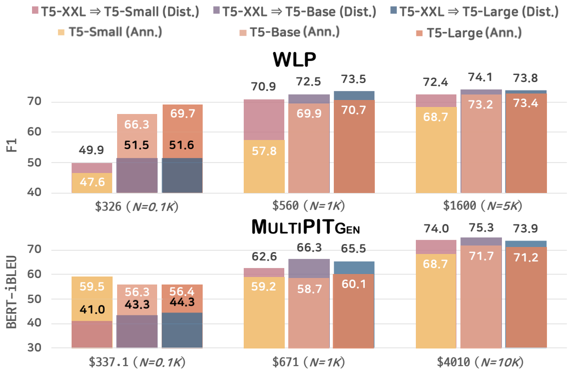

Figure 4: Results with different compact model sizes: `Small` (60M), `Base` (220M), `Large` (770M). For `Dist.`, a teacher is fixed (`XXL-11B`), and the distillation cost is set to `$300`. Best viewed in color.

especially for MULTIPIT$_{\text{Gen}}$ as also evidenced in Appendix (Table 11). It is worth noting that there is a common trend across all tasks that the `Dist.` performances drop with high *N*. This is due to the limited budgets; high *N* requires a substantial fine-tuning cost for a large model, hence the budget to be used for distillation is limited. For instance, in the case of STANCEOSAURUS with `budget=$200`, if *N* is *1K* , *82K* unlabeled data can be used for distillation, whereas only *35K* unlabeled data are used when *N*= *10K* , resulting in the former outperforming the latter. This offers a lesson that unconditionally pursuing larger *N* is not desirable in a fixed budget scenario; it is advisable for practitioners to understand and consider the trade-off between the fine-tuning and distillation costs.

4.2 Varying the Compact Model Size

To consider various inference scenarios, we explore different sizes of a compact model in Figure 4. In general, the performances of all models improve as the budget increases, and `Dist.` outperforms `Ann.` given the same cost except for the low budget (*N=0.1K*) setting. Interestingly, we observe that `T5-XXL` ⇒ `T5-Base (Dist.)` is better than `T5-XXL` ⇒ `T5-Large (Dist.)` in some cases (`$1600` for WLP, `$671` and `$4010` for MULTIPIT$_{\text{Gen}}$) although the former is smaller and more efficient. We conjecture that this is attributed to the model's larger number of parameters that require more GPUs and thereby more cost. This result disproves the prevailing belief that larger models are always superior, at least in fixed-budget scenarios.

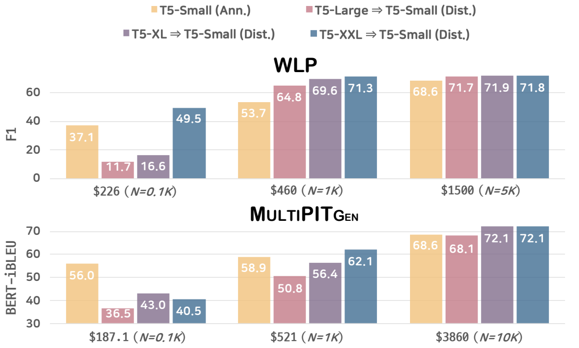

Figure 5: Results with varied scales of the teacher: `Large` (770M), `XL` (3B), `XXL` (11B). The compact model is fixed (`Small-60M`). The distillation cost is fixed as `$200` for WLP and `$150` for MULTIPIT$_{\text{Gen}}$.

4.3 Varying the Teacher Model Size

We now investigate teacher models with different scales (Figure 5). It turns out that relatively smaller teacher models (`T5-Large` & `T5-XL`) cannot be good teachers in the low budgets scenarios. For instance, with `$521` for MULTIPIT$_{\text{Gen}}$, `T5-Large` ⇒ `T5-Small (Dist.)` and `T5-XL` ⇒ `T5-Small (Dist.)` underperform `T5-Small (Ann.)`, whereas `T5-XXL` ⇒ `T5-Small (Dist.)` outperforms `T5-Small (Ann.)`. In higher budget settings, it is noticeable that the largest teacher (`XXL`) is similar to or better than the smaller teacher (`Large`, `XL`). Taken together, this analysis suggests that when adopting distillation, the scale of the teacher model matters, and it may be safe to leverage sufficiently a larger model as a teacher regardless of any budgetary scenarios.

5 GPT-3.5 as an Annotator

Furthermore, we examine the cost efficiency of `GPT-3.5` (Ouyang et al., 2022) annotation through an in-context few-shot learning scheme. Wang et al. (2021) has recently demonstrated that `GPT-3` (Brown et al., 2020) can be used as a cheaper labeler compared to humans. We attempt to scrutinize its applicability to the tasks considered in this work, and also contextualize its result with that of `Dist.` ultimately. We make use of the `text-davinci-003` model to generate pseudo-labels by prompting with 32 training examples. In this experiment, we assign `$200` each for WLP and STANCEOSAURUS for `GPT-3.5` annotation. Note that OpenAI[7] charges money based on the number of tokens used. The cost per label for WLP

[7] https://openai.com/api/pricing

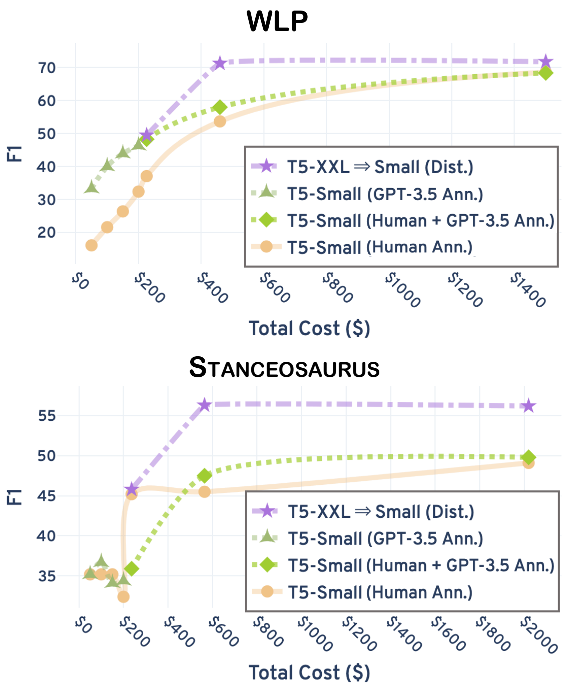

Figure 6: Comparisons with `GPT-3.5` annotation. Given an initial human annotation N={*0.1K*, *1K*, *5K*} with the corresponding costs, `$200` is additionally allocated for distillation or `GPT-3.5` annotation (i.e., `Human + GPT-3.5 Ann.`).

is $0.046 and for STANCEOSAURUS is $0.073, if using `GPT-3.5` (details in Appendix E).

In Figure 6, we compare `GPT-3.5` annotation (`GPT-3.5 Ann.`) against the human annotation and distillation strategy. In addition to `GPT-3.5 Ann.`, we combine it with human annotation (`Human + GPT-3.5 Ann.`) to enhance quality and make a comparison with `Dist`. The results clearly show that while `GPT-3.5` could be better than human annotators as hinted in prior work (Wang et al., 2021), it significantly underperforms the distillation (`Dist.`) strategy given the same budget despite `GPT-3.5`'s larger parameters (175B) than the teacher (11B). This once again highlights the different view of knowledge distillation: cost efficiency.

6 Related Work

The costs associated with building models have been explored or concerned by many prior works.

Data Annotation. On one hand, researchers have attempted to tackle the problem of noisy or expensive human annotation. For example, Zhang et al. (2021) studies how to distribute annotation budgets between more examples with a single label and fewer examples with many labels. Chen et al. (2022) investigates a redundant annotation with a majority vote vs. cleaning or relabeling the incorrect annotations. Wang et al. (2021) compares human annotations against `GPT-3` (Brown et al., 2020) annotations. However, these works only focus on the annotation cost.

Knowledge Distillation. On the other hand, other lines of work address computational budgets associated with knowledge distillation. Ye et al. (2022) proposes using a larger and sparser student model than a teacher model to further reduce inference cost. Jooste et al. (2022) compares different distillation schemes for cheap, fast, and environmentally friendly translation models. Ma et al. (2022) explores an efficient interactive distillation with meta-learning. The aforementioned works, however, ignore the data budgets and/or barely consider the realistic computational costs involved in the distillation process. While knowledge distillation has been shown effective for compression or generalization in previous NLP works (Sanh et al., 2019; Kang et al., 2020; Le et al., 2022), it remains unclear whether or not it is efficient even when considering the actual cost of distillation, which is often overlooked. As concurrent works, Sun et al. (2023) presents a novel principle-driven self-alignment approach, and Hsieh et al. (2023) introduces a method that involves step-by-step distillation using chain-of-thought (Wei et al., 2022) rationales. Although the main focus is completely different from ours (i.e., cost), we believe that these works not only enhance this particular area but also have the potential to support our own findings regarding the cost-efficiency of distillation as the new methods would make the gap with annotation even bigger.

Data and Compute. Unlike most existing works that consider exclusively either annotation or computational cost, our study contextualizes the two superficially dissociated types of costs, known to be expensive (Ning et al., 2019; Hong et al., 2020; Hendrycks et al., 2021; Izsak et al., 2021; Obando-Ceron and Castro, 2021; Minixhofer et al., 2022) while being obscure in how they can be comparable to each other. Kirstain et al. (2022) compares scaling parameters against adding more labeled examples, but a compact model and a realistic cost ($) are not of interest to it. Our work resembles Bai et al. (2021) in terms of study framework, which explores how to optimally assign pre-training and annotation costs specifically for domain adaptation settings. Our focus is more on fine-tuning/distilling a compact model rather than pre-training from scratch and on exploring more general scenarios with diverse tasks.

7 Conclusion

In this work, we address a dilemma that practitioners often face when building a model: *given a limited budget, how to invest it to train a compact model in an economically efficient manner?* We provide empirical evidence that (i) only scaling data using human annotators or `GPT-3.5` for annotation may not be the most economical solution, and (ii) when adopting the distillation strategy, using a smaller amount of unlabeled data leads to Pareto efficient models with a smaller budget, while it becomes more beneficial to use larger amounts of unlabeled data as the budget increases. Furthermore, (iii) we demonstrate that in budget-constrained settings, a smaller final model could produce both better performance and more efficient inference. Given these findings, future work can explore different approaches to leveraging a large model's capability such as pruning for cost-efficient compact models.

Limitations

This paper fundamentally considers a scenario in which practitioners rent cloud GPUs. In the case of hosting GPUs by themselves, the two strategies explored in this study would not be simply comparable. However, in practice, when training a large model (w/ 8 A100 GPUs), we conjecture that renting GPUs could be preferred in many cases as scaling compute powers is not trivial and prohibitively expensive (Izsak et al., 2021; Obando-Ceron and Castro, 2021; Minixhofer et al., 2022). It is also noteworthy that in the future, computational costs may become cheaper as new hardware advances, the pricing policy by cloud platform services changes, and more optimization techniques are applied. On the other hand, human annotation cost is likely to be the same at least or even more expensive. With cost changes in such a direction, the same conclusion made by our study will hold even though the gap between the two strategies will get larger.

For a compression method, our work focuses on knowledge distillation (Hinton et al., 2015). However, it is worth noting that distillation amplifies a societal bias in a compressed model (Hooker et al., 2020; Silva et al., 2021) due to its limited capacity (Ahn et al., 2022). Accordingly, practitioners are encouraged to additionally leverage bias mitigation techniques (Ahn et al., 2022) when adopting distillation for real-world applications. On top of our finding that the distillation scheme is more cost-efficient than the data annotation approach, other efficient methods such as pruning (Xia et al., 2022) may be investigated in future work to decide which one is the best efficient solution among methods that leverages a large model. We believe, however, it should be noted that retaining performances after pruning a large portion (e.g., ~99.995%: 11B ⇒ 60M) for a compact model would not be trivial, evidenced in a prior work (Michel et al., 2019).

Acknowledgments

We thank Fan Bai and Jonathan Zheng for their assistance in estimating data annotation costs and collecting unlabeled data for WLP and Stanceosaurus, respectively. This material is based upon work supported by the NSF (IIS-2052498) and by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via the HIATUS Program contract #2022-22072200004. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of NSF, ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein.

References

AmirAli Abdolrashidi, Lisa Wang, Shivani Agrawal, Jonathan Malmaud, Oleg Rybakov, Chas Leichner, and Lukasz Lew. 2021. Pareto-optimal quantized resnet is mostly 4-bit. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3091–3099.

Jaimeen Ahn, Hwaran Lee, Jinhwa Kim, and Alice Oh. 2022. Why knowledge distillation amplifies gender bias and how to mitigate from the perspective of DistilBERT. In *Proceedings of the 4th Workshop on Gender Bias in Natural Language Processing (GeBNLP)*, pages 266–272, Seattle, Washington. Association for Computational Linguistics.

Fan Bai, Alan Ritter, and Wei Xu. 2021. Pre-train or annotate? domain adaptation with a constrained budget.

In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5002–5015, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Gary S. Becker. 1965. A theory of the allocation of time. *The Economic Journal*, 75(299):493–517.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Derek Chen, Zhou Yu, and Samuel Bowman. 2022. Clean or annotate: How to spend a limited data collection budget. In *Proceedings of the Third Workshop on Deep Learning for Low-Resource Natural Language Processing*, pages 152–168.

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)*, pages 4171–4186.

Yao Dou, Chao Jiang, and Wei Xu. 2022. Improving large-scale paraphrase acquisition and generation. In *In Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. 2021. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129(6):1789–1819.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. 2020. Retrieval augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 3929–3938. PMLR.

Dan Hendrycks, Collin Burns, Anya Chen, and Spencer Ball. 2021. Cuad: An expert-annotated nlp dataset for legal contract review. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1.

Danny Hernandez, Jared Kaplan, Tom Henighan, and Sam McCandlish. 2021. Scaling laws for transfer. *arXiv preprint arXiv:2102.01293*.

Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. 2015. Distilling the knowledge in a neural network. In *NIPS Deep Learning and Representation Learning Workshop*.

Giwon Hong, Junmo Kang, Doyeon Lim, and Sung-Hyon Myaeng. 2020. Handling anomalies of synthetic questions in unsupervised question answering. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3441–3448, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Sara Hooker, Nyalleng Moorosi, Gregory Clark, Samy Bengio, and Emily Denton. 2020. Characterising bias in compressed models.

Cheng-Yu Hsieh, Chun-Liang Li, Chih-Kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alexander Ratner, Ranjay Krishna, Chen-Yu Lee, and Tomas Pfister. 2023. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes.

Peter Izsak, Moshe Berchansky, and Omer Levy. 2021. How to train BERT with an academic budget. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10644–10652, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. 2020. Tinybert: Distilling bert for natural language understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4163–4174.

Wandri Jooste, Andy Way, Rejwanul Haque, and Riccardo Superbo. 2022. Knowledge distillation for sustainable neural machine translation. In *Proceedings of the 15th Biennial Conference of the Association for Machine Translation in the Americas (Volume 2: Users and Providers Track and Government Track)*, pages 221–230.

Junmo Kang, Giwon Hong, Haritz Puerto San Roman, and Sung-Hyon Myaeng. 2020. Regularization of distinct strategies for unsupervised question generation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3266–3277, Online. Association for Computational Linguistics.

Yuval Kirstain, Patrick Lewis, Sebastian Riedel, and Omer Levy. 2022. A few more examples may be worth billions of parameters. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

Kelvin J. Lancaster. 1966. A new approach to consumer theory. *Journal of Political Economy*, 74(2):132–157.

Nghia Le, Fan Bai, and Alan Ritter. 2022. Few-shot anaphora resolution in scientific protocols via mixtures of in-context experts. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Nayeon Lee, Belinda Z. Li, Sinong Wang, Wen-tau Yih, Hao Ma, and Madian Khabsa. 2020. Language models as fact checkers? In *Proceedings of the Third Workshop on Fact Extraction and VERification (FEVER)*, pages 36–41, Online. Association for Computational Linguistics.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

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*.

Xinge Ma, Jin Wang, Liang-Chih Yu, and Xuejie Zhang. 2022. Knowledge distillation with reptile meta-learning for pretrained language model compression. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 4907–4917, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Lucie Charlotte Magister, Jonathan Mallinson, Jakub Adamek, Eric Malmi, and Aliaksei Severyn. 2022. Teaching small language models to reason. *arXiv preprint arXiv:2212.08410*.

Paul Michel, Omer Levy, and Graham Neubig. 2019. Are sixteen heads really better than one? In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Sewon Min, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. Noisy channel language model prompting for few-shot text classification. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5316–5330, Dublin, Ireland. Association for Computational Linguistics.

Benjamin Minixhofer, Fabian Paischer, and Navid Rekabsaz. 2022. WECHSEL: Effective initialization of subword embeddings for cross-lingual transfer of monolingual language models. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3992–4006, Seattle, United States. Association for Computational Linguistics.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A human generated machine reading comprehension dataset. In *Proceedings of the Workshop on Cognitive Computation: Integrating neural and symbolic approaches 2016 co-located with the 30th Annual Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain, December 9, 2016*, volume 1773 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Qiang Ning, Hangfeng He, Chuchu Fan, and Dan Roth. 2019. Partial or complete, that's the question. 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)*, pages 2190–2200, Minneapolis, Minnesota. Association for Computational Linguistics.

Tong Niu, Semih Yavuz, Yingbo Zhou, Nitish Shirish Keskar, Huan Wang, and Caiming Xiong. 2021. Unsupervised paraphrasing with pretrained language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5136–5150, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Johan S Obando-Ceron and Pablo Samuel Castro. 2021. Revisiting rainbow: Promoting more insightful and inclusive deep reinforcement learning research. In *Proceedings of the 38th International Conference on Machine Learning*, Proceedings of Machine Learning Research. PMLR.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Gray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*.

David Patterson, Joseph Gonzalez, Quoc Le, Chen Liang, Lluis-Miquel Munguia, Daniel Rothchild, David So, Maud Texier, and Jeff Dean. 2021. Carbon emissions and large neural network training. *arXiv preprint arXiv:2104.10350*.

Fabio Petroni, Aleksandra Piktus, Angela Fan, Patrick Lewis, Majid Yazdani, Nicola De Cao, James Thorne, Yacine Jernite, Vladimir Karpukhin, Jean Maillard, Vassilis Plachouras, Tim Rocktäschel, and Sebastian Riedel. 2021. KILT: a benchmark for knowledge intensive language tasks. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2523–2544, Online. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Adam Roberts, Colin Raffel, and Noam Shazeer. 2020. How much knowledge can you pack into the parameters of a language model? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5418–5426, Online. Association for Computational Linguistics.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. In *NeurIPS EMC$\hat{2}$ Workshop*.

Tal Schuster, Adam Fisch, and Regina Barzilay. 2021. Get your vitamin C! robust fact verification with contrastive evidence. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 624–643, Online. Association for Computational Linguistics.

Andrew Silva, Pradyumna Tambwekar, and Matthew Gombolay. 2021. Towards a comprehensive understanding and accurate evaluation of societal biases in pre-trained transformers. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2383–2389, Online. Association for Computational Linguistics.

Samuel Stanton, Pavel Izmailov, Polina Kirichenko, Alexander A Alemi, and Andrew G Wilson. 2021. Does knowledge distillation really work? *Advances in Neural Information Processing Systems*, 34.

Zhiqing Sun, Yikang Shen, Qinhong Zhou, Hongxin Zhang, Zhenfang Chen, David Cox, Yiming Yang, and Chuang Gan. 2023. Principle-driven self-alignment of language models from scratch with minimal human supervision.

Jeniya Tabassum, Wei Xu, and Alan Ritter. 2020. WNUT-2020 task 1 overview: Extracting entities and relations from wet lab protocols. In *Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020)*, pages 260–267, Online. Association for Computational Linguistics.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for fact extraction and VERification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 809–819, New Orleans, Louisiana. Association for Computational Linguistics.

Marcos Treviso, António Góis, Patrick Fernandes, Erick Fonseca, and André FT Martins. 2022. Predicting attention sparsity in transformers. In *Proceedings of the Sixth Workshop on Structured Prediction for NLP*, pages 67–81.

Shuohang Wang, Yang Liu, Yichong Xu, Chenguang Zhu, and Michael Zeng. 2021. Want to reduce labeling cost? GPT-3 can help. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4195–4205, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2019. Huggingface's transformers: State-of-the-art natural language processing.

Mengzhou Xia, Zexuan Zhong, and Danqi Chen. 2022. Structured pruning learns compact and accurate models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1513–1528, Dublin, Ireland. Association for Computational Linguistics.

Canwen Xu and Julian McAuley. 2022. A survey on model compression and acceleration for pretrained language models.

Qinyuan Ye, Madian Khabsa, Mike Lewis, Sinong Wang, Xiang Ren, and Aaron Jaech. 2022. Sparse distillation: Speeding up text classification by using bigger student models. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2361–2375.

Linfeng Zhang, Jiebo Song, Anni Gao, Jingwei Chen, Chenglong Bao, and Kaisheng Ma. 2019. Be your own teacher: Improve the performance of convolutional neural networks via self distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3713–3722.

Shujian Zhang, Chengyue Gong, and Eunsol Choi. 2021. Learning with different amounts of annotation: From zero to many labels. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7620–7632.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 12697–12706. PMLR.

Jonathan Zheng, Ashutosh Baheti, Tarek Naous, Wei Xu, and Alan Ritter. 2022. Stanceosaurus: Classifying stance towards multilingual misinformation. In *In Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Chunting Zhou, Junxian He, Xuezhe Ma, Taylor Berg-Kirkpatrick, and Graham Neubig. 2022a. Prompt consistency for zero-shot task generalization.

Wangchunshu Zhou, Canwen Xu, and Julian McAuley. 2022b. Bert learns to teach: Knowledge distillation with meta learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7037–7049.

A Details of Annotation Cost Estimation

WLP (Tabassum et al., 2020) This is an annotated corpus containing wet lab protocols, and the included tasks are named entity recognition (NER) and relation extraction (RE). We refer to Bai et al. (2021) for the price per sentence (instance), which is \$0.44. Since this price is measured for both tasks, and we are only interested in NER, we take the ratio of the number of labels for each (59.76%:40.24%) for the estimate of NER in isolation, yielding approximately \$0.26.

STANCEOSAURUS (Zheng et al., 2022) This dataset includes sourced claims and relevant tweets along with annotated stances for stance classification. Since the labeling cost was not explicitly mentioned in the paper, we asked the authors for the details of the average number of annotations per hour (82 tweets) and the hiring cost (\$15 per hour) to calculate the final price per label: \$15 ÷ 82 × 2 (double-annotated) = \$0.364.

MULTIPIT (Dou et al., 2022) This provides Twitter-based paraphrase containing multiple topics. We specifically consider, out of variants, MULTIPIT$_{\text{CROWD}}$ corpus, consisting of sentence pairs labeled whether each pair is paraphrased or not for paraphrase identification (MULTIPIT$_{\text{Id}}$). The cost per pair is considered \$0.2 as mentioned in the paper. For paraphrase generation (MULTIPIT$_{\text{Gen}}$), we sample pairs annotated as paraphrased, and take the proportion of sampled ones out of the total (53.9%) to get the cost per paraphrased source-target instance: 100 ÷ 53.9 × \$0.2 = \$0.371.

FEVER (Thorne et al., 2018) **& NATURAL QUESTIONS** (Kwiatkowski et al., 2019) These are fact verification and question answering datasets respectively for which we estimate the costs by leveraging the price from Google Cloud Platform. This charges \$129 per 50 words for 1,000 units, and hence we get an estimate of \$0.129 per label for both tasks.

B Input-Output Formats for Each Task

Our study uses T5 (Raffel et al., 2020) as our base model under the standard text-to-text framework. The input-output examples for each task are demonstrated in Table 7, and what follows is detailed explanations for each.

WLP This task can be regarded as a token-level classification problem, where the #class is 20 in total: {Amount, Reagent, Device, Time, Speed, Action, Mention, Location, Numerical, Method, Temperature, Modifier, Concentration, Size, Generic-Measure, Seal, Measure-Type, Misc, Ph, Unit}. Given a source input (i.e., procedural sentence), the model is required to generate a target as a form of "Entity [Label] Entity [Label] ...".

STANCEOSAURUS For this task, the source is the concatenation of a claim, a relevant tweet, and context information (e.g., reply), and the target is supposed to one of {Supporting | Refuting | Irrelevant | Discussing | Querying}.

FEVER This is a fact verification task where the source is a claim (closed-book setting as discussed in §3), and the target is Supports or Refutes in a 2-way classification setting following Petroni et al. (2021).

MULTIPIT$_{\text{Id}}$ is also a binary classification task where given two sentences, targets should be Yes or No.

MULTIPIT$_{\text{Gen}}$ The source for this task is a sentence and the target is a paraphrased sentence.

NATURAL QUESTIONS As in **FEVER**, we also consider the closed-book setup that requires a model to rely on its implicit knowledge for this task where the question is a source and the target is directly the answer to the question.

Dataset	**#Train**	**#Dev**	**#Test**	**#Unlabeled Data**
WLP	11,966	2,861	3,562	111,000
STANCEOSAURUS	12,130	3,827	4,750	126,000
FEVER	104,966	10,444	N/A	182,000
MULTIPIT$_{\text{Id}}$	92,217	11,527	11,530	237,000
MULTIPIT$_{\text{Gen}}$	49,673	6,143	6,120	179,000
NATURAL QUESTIONS	87,372	2,837	N/A	115,000

Table 5: Statistics for various NLP datasets. For FEVER and NATURAL QUESTIONS, dev sets are used for evaluation as test sets are private. The maximum number of unlabeled data used for experiments is presented.

C Detailed Settings and Hyperparameters

As described in §3, we utilize `T5 v1.1` (Roberts et al., 2020) as a base model, because the original version of `T5` (Raffel et al., 2020) was pre-trained using a combination of several supervised tasks as well as an unsupervised task. Since this work assumes that no supervised datasets are available, our fine-tuning strategies build upon `T5 v1.1` that was pre-trained in an unsupervised way only. For a

Dataset	Initial $	Annotation (Ann.)	Distillation (Dist.)
	(*N=5K*)	T5-Small (*+1K*)	T5-XXL (*5K*) ⇒ T5-Small (*100K*)
WLP	$1,300	$260	$67.5 ⇒ $435
Stanceosaurus	$1,820	$364	$60 ⇒ $225
FEVER	$645	$129	$22.5 ⇒ $78
MultiPIT$_{Id}$	$1,000	$200	$37.5 ⇒ $123
MultiPIT$_{Gen}$	$1,855	$371	$45 ⇒ $163
Natural Questions	$645	$129	$30 ⇒ $86

Table 6: Example breakdown of cost estimates for training compact models using the two approaches illustrated in Figure 1. Starting with (*5K*) labeled examples, we compare the costs of annotating an additional *+1K* , or fine-tuning, then distilling T5-XXL (11B parameters). For **Distillation (Dist.)**, the computational cost involves fine-tuning T5-XXL (the teacher) on *5K* annotated data, plus distilling it into T5-Small using *100K* unlabeled examples.

Dataset	Task	Example
WLP	Named Entity Recognition	**Source -** Assemble the following reagents in a thin-walled PCR tube
		Target - Assemble [Action] following reagents [Reagent] thin-walled PCR tube [Location]
Stanceosaurus	Stance Classification	**Source -** claim: The suicide rate increased during COVID-19 lockdown. tweet: @USER @USER People who are suicidal can hide the signs very well. [SEP] @USER @USER So we aren't looking at the family units for this then? If people are at home all day, everyday with their kids then why aren't they seeing the signs? Oh wait, it's easier to blame everyone else
		Target - {Supporting | Refuting | Irrelevant | Discussing | Querying }
FEVER	Fact Verification	**Source -** History of art includes architecture, dance, sculpture, music, painting, poetry literature, theatre, narrative, film, photography and graphic arts.
		Target - {Supports | Refutes}
MultiPIT$_{Id}$	Paraphrase Identification	**Source -** sentence1: well 160 people died in Bangladesh due to building collapse sentence2: #bangladesh Death toll climbs in Bangladesh building collapse
		Target - {Yes | No}
MultiPIT$_{Gen}$	Paraphrase Generation	**Source -** President Obama will hold a press conference at 10:15 a.m. at the White House
		Target - President Obama will be taking questions from reporters at 10:15 am ET in the briefing room
Natural Questions	Question Answering	**Source -** Who is the first person who went to moon?
		Target - Neil Alden Armstrong

Table 7: Input-output examples for each task.

Hyperparameters	**WLP**	**Stanceosaurus**	**FEVER**	**MultiPIT$_{Id}$**	**MultiPIT$_{Gen}$**	**Natural Questions**
Max Source Length	128	128	128	64	32	32
Max Target Length	128	8	8	8	32	32
Batch Size	32	32	32	32	32	32
Epochs	50 (20)	50 (20)	50 (20)	50 (20)	50 (20)	50 (20)
Learning Rate	3e-5	3e-5	3e-5	3e-5	3e-5	1e-3 (3e-5)

Table 8: Hyperparameters used for training models. The numbers in () are used exceptionally for T5-XXL (i.e., teacher) fine-tuning.

question answering task, we exceptionally use the checkpoint additionally pre-trained using salient span masking (SSM), an unsupervised pre-training objective known to be helpful for open-domain question answering (Guu et al., 2020), following Roberts et al. (2020).

Table 5 presents the dataset statistics and Table 8 presents the hyperparameters used for training models for each task. We did not try to specifically tune the hyperparameters for each model for each task, taking into account the scenario considered by this study in which annotated data is highly limited. Moreover, in order to minimize factors other than the ones we consider for each setup, we fixed each parameter as much as possible unless significant problems were observed during training. Specifically, we chose the learning rate of 3e-5 (default in the Huggingface (Wolf et al., 2019) code base for question answering and seq2seq distillation), which we believe is not out of the ordinary, for all except for NATURAL QUESTIONS where we adopt 1e-3 when training T5-Small model as we observed the phenomenon that it was not being trained at all by looking at its training loss with 3e-5. We trained all models with 50 epochs except for a T5-XXL model where fewer epochs are assumed to be enough. We used the final batch size of 32 by leveraging the gradient accumulation (e.g., batch size of {16, 8} and gradient accumulation of {2, 4}) when necessary to meet VRAM constraints. We adopt (layer-wise) model parallelism that allows us to load a large model on multiple GPUs. Our reported results are based on a single run due to the high computational cost required by our empirical study. Despite this, a significant difference in performance was observed between the two strategies being compared.

D Unlabeled Data for Each Task

For the distillation strategy, unlabeled data is essentially required to transfer a large model's knowledge into a small model. In this work, unlabeled data is literally referred to the data without the corresponding labels (i.e., only source inputs in Table 7). We exploit only input sources (without annotations) in the existing datasets excluding ones that models are evaluated on. Plus, we collect additional unlabeled corpora for each dataset for an extensive study as follows:

WLP This dataset requires procedural text as an input source. We utilize large-scale PROCEDURE corpus (Bai et al., 2021) that contains diverse domains. We specifically use CHEMSYN, chemical synthesis procedures in patents, for this study.

STANCEOSAURUS The input source for this dataset consists of a claim from diverse fact-checking sites, a tweet relevant to the claim, and contextual information such as a reply or parent tweet if any. Following the methodology described in this work (Zheng et al., 2022), we collected claims and corresponding tweets by anonymizing user information.

FEVER Statements or claims are sufficient to be sources for this dataset. We leverage the synthetically generated claims in Schuster et al. (2021).

MULTIPIT The sources for this dataset are sentences written by Twitter users, which can be collected by following the method in Dou et al. (2022). For this work, we instead exploit sources of MULTIPIT$_{\text{AUTO}}$ (Dou et al., 2022) as unlabeled data, automatically collected recent datasets.

NATURAL QUESTIONS The source simply consists of a question. Therefore, we make use of queries in MS MARCO (Nguyen et al., 2016), where the queries are sampled from Bing's search logs.

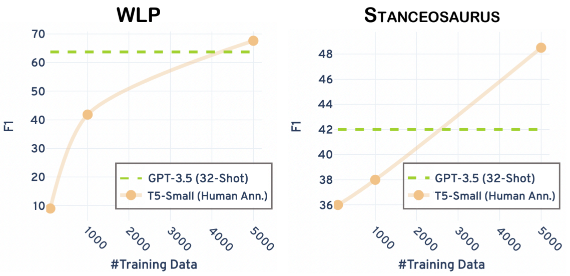

Figure 7: Preliminary results on 200 sampled test sets, comparing GPT-3.5 32-shot in-context learning against T5-Small with varying the size of training data.

E Details of GPT-3.5 Annotation

To annotate pseudo-labels using GPT-3.5, we make use of the strongest version, text-davinci-003 with 32 training examples. Our input prompt consists of a task-specific instruction[8][9] and 32 in-context examples, and unlabeled input to annotate at the end. In order to reduce the high variance (Zhao et al., 2021; Min et al., 2022), we randomly sample and shuffle 32 in-context examples out of a 100 fixed training set for each annotation iteration. In Figure 7, we present the performance of GPT-3.5's 32-shot learning to see its quality and feasibility, and we find that it can be qualified as a cheap labeler to improve performances, especially for low-budget settings, as found in Wang et al. (2021).

Note that OpenAI[10] API charges based on the number of tokens for input prompt plus model output: $0.02 per 1K tokens. Therefore, the $ per label is calculated as $0.046 for WLP (2.3K tokens on average) and $0.073 for STANCEOSAURUS (3.65K tokens on average). Based on this, we annotate 4347 data for WLP and 2739 data for STANCEOSAURUS in total, using $200 assigned for each task.

[8] For WLP, "Classify named entities into one of the following categories: {Class 1, Class2, ...}"

[9] For STANCEOSAURUS, "Classify the stance of a given tweet toward a given claim into one of the following categories: {Class 1, Class2, ...}"

[10] https://openai.com/api/pricing

Dataset	Exisiting Models	T5-XXL (Full)
WLP	75.9 (Bai et al., 2021)	74.4
Stanceosaurus	61.0 (Zheng et al., 2022)	63.3 [69.8]
FEVER	78.9 (Petroni et al., 2021)	82.1
MultiPIT$_{\text{Id}}$	91.4 (Dou et al., 2022)	90.8
MultiPIT$_{\text{Gen}}$	77.8 (Dou et al., 2022)	75.9
Natural Questions	35.2 (Roberts et al., 2020)	31.3 [38.5]

Table 9: Resource-unconstrained performances of existing models and fully fine-tuned in-house T5-XXL for reference or upper bounds. Due to the use of different metrics, we also report macro F1 for Stanceosaurus, and the EM score for Natural Questions, along with the [micro F1] used in this work.

Model	Stanceosaurus		FEVER	
	$664	$2120	$279	$795
	(*N=1K*)	(*N=5K*)	(*N=1K*)	(*N=5K*)
`T5-Small` (Ann.)	44.7	50.3	49.8	68.9
`DistilBERT` (General Dist. + Ann.)	56.3	57.5	69.9	73.5
`BERT`$_{\text{Base}}$ (Ann.)	56.0	59.0	70.7	73.1
`T5-XXL` ⇒ `T5-Small` (Dist.)	56.9	60.5	71.7	74.8

Table 10: Results (*N=5K*) of `Ann.`, general distillation (**`DistilBERT`** (Sanh et al., 2019)), and task-specific distillation on Stanceosaurus and FEVER. For **`DistilBERT`**, the computational cost for distillation in the pre-training phase is assumed to be $0. The final model size is similar to each other (∼60M) except for **`BERT`$_{\text{Base}}$** (110M). General (pre-training) distillation and task-specific (fine-tuning) distillation are complementary (Jiao et al., 2020).

F Additional Results

How well do off-the-shelf models perform for each task? In Table 9, we provide the results of the largest `T5` model (11B) fined-tuned on full training data, along with relevant works' results in resource-rich settings. Those reported numbers can serve as upper bounds or references for calibrating the relative results produced in this work (i.e., resource-limited settings). Note that these should not be used for direct comparison due to various combinations of factors including model architectures, size, approaches, pre-training scheme, training data, and budgets.

What about general distillation? While this work focuses on task-specific distillation, we also provide the result of general distillation (`DistilBERT` (Sanh et al., 2019)) in which a model is distilled during the pre-training phase to learn general language understanding capability before fine-tuning. To measure the total cost, the computational cost for distillation in the pre-training phase is assumed to be `$0` (i.e., it is publicly available). In Table 10, we find that given the same budget, adding general distillation leads to more cost-efficient than the annotation strategy without distillation. In addition to this, it is important to note that intuitively, general distillation (pre-training) and task-specific (fine-tuning) distillation can be combined for the better, evidenced in Jiao et al. (2020). This further spotlights the cost-efficient aspect of distillation methods.

Task	*N* (Initial $)	Strategy	**Additional $**			
			Ann. Performance (*#Additional Data*) Dist. Performance (*GPU Hours / #Unlabeled Data*)			
			+$0	+$100	+$200	+$300
WLP	*100* ($26)	T5-Small (Ann.) T5-XXL [**48.8**] ⇒ T5-Small (Dist.)	**9.1** (*+0*) N/A	23.8 (*+384*) **49.5** (*54h / 22K*)	37.1 (*+769*) **49.5** (*107h / 45K*)	47.6 (*+1153*) **49.9** (*160h / 68K*)
			+$0	+$100	+$200	+$300
Stanceosaurus	*100* ($36)	T5-Small (Ann.) T5-XXL [**44.8**] ⇒ T5-Small (Dist.)	**35.2** (*+0*) N/A	35.2 (*+274*) **45.8** (*54h / 42K*)	45.2 (*+549*) **45.8** (*107h / 87K*)	45.4 (*+824*) **45.6** (*160h / 131K*)
			+$0	+$50	+$100	+$150
FEVER	*100* ($13)	T5-Small (Ann.) T5-XXL [**49.7**] ⇒ T5-Small (Dist.)	**50.3** (*+0*) N/A	49.3 (*+387*) **49.7** (*27h / 59K*)	49.7 (*+775*) 49.7 (*54h / 123K*)	49.7 (*+1162*) 49.7 (*80h / 187K*)
			+$0	+$100	+$200	+$300
MultiPIT$_{\text{Id}}$	*100* ($20)	T5-Small (Ann.) T5-XXL [**53.1**] ⇒ T5-Small (Dist.)	**46.9** (*+0*) N/A	53.1 (*+500*) 53.1 (*54h / 78K*)	53.1 (*+1000*) 53.1 (*107h / 159K*)	53.1 (*+1500*) 53.1 (*160h / 240K*)
			+$0	+$100	+$200	+$300
MultiPIT$_{\text{Gen}}$	*100* ($37)	T5-Small (Ann.) T5-XXL [**55.5**] ⇒ T5-Small (Dist.)	**45.0** (*+0*) N/A	**53.1** (*+269*) 41.4 (*54h / 59K*)	**57.3** (*+539*) 40.6 (*107h / 120K*)	**59.5** (*+808*) 41.0 (*160h / 181K*)
			+$0	+$50	+$100	+$150
Natural Questions	*100* ($13)	T5-Small (Ann.) T5-XXL [**18.6**] ⇒ T5-Small (Dist.)	**2.3** (*+0*) N/A	3.3 (*+387*) **9.1** (*27h / 37K*)	3.9 (*+775*) **11.0** (*54h / 78K*)	4.2 (*+1162*) **11.0** (*80h / 118K*)

Table 11: Detailed results in a few-shot learning scenario (*N*=100) to investigate the cost efficiency of a small model with more data annotations (Ann.) and teacher [**performance**] ⇒ student (small) distillation (Dist.) on various NLP tasks. *N* indicates the number of starting data annotated with the corresponding (initial $). (*#Additional Data*) refers to the number of annotated data additional to *N*, and (*GPU Hours*) denotes the total GPU hours for fine-tuning the teacher model on *N* data, plus for the distillation into a small model using varying (*#Unlabeled Data*).

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Limitation section

☑ A2. Did you discuss any potential risks of your work?
Limitation section

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

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

B ☑ Did you use or create scientific artifacts?

Section 2

☑ B1. Did you cite the creators of artifacts you used?
Section 2

□ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Not applicable. We used existing datasets and pre-trained models, following their licenses.

☑ 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)?
Section D

□ 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.
Section 2

C ☑ Did you run computational experiments?

Section 3 5

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Section 2 5

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?
Section 3.1 and Appendix C

□ 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?
Not applicable. Appendix C

□ 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.)?
Not applicable. Left blank.

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.