Parameter-Efficient Legal Domain Adaptation

Jonathan Li[1], **Rohan Bhambhoria**[1,2], **Xiaodan Zhu**[1,2]

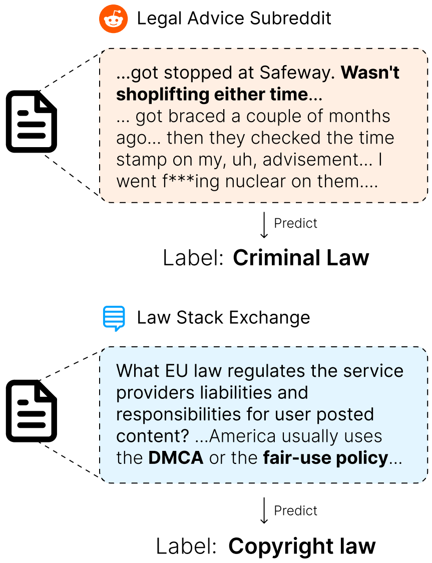
[1] Ingenuity Labs, Queen's University
[2] Department of Electrical and Computer Engineering, Queen's University
{jxl, r.bhambhoria, xiaodan.zhu}queensu.ca

Abstract

Seeking legal advice is often expensive. Recent advancements in machine learning for solving complex problems can be leveraged to help make legal services more accessible to the public. However, real-life applications encounter significant challenges. State-of-the-art language models are growing increasingly large, making parameter-efficient learning increasingly important. Unfortunately, parameter-efficient methods perform poorly with small amounts of data (Gu et al., 2022), which are common in the legal domain (where data labelling costs are high). To address these challenges, we propose parameter-efficient legal domain adaptation, which uses vast unsupervised legal data from public legal forums to perform legal pre-training. This method exceeds or matches the fewshot performance of existing models such as LEGAL-BERT (Chalkidis et al., 2020) on various legal tasks while tuning only approximately 0.1% of model parameters. Additionally, we show that our method can achieve calibration comparable to existing methods across several tasks. To the best of our knowledge, this work is among the first to explore parameter-efficient methods of tuning language models in the legal domain.

1 Introduction

Seeking legal advice from lawyers can be expensive. However, a machine learning system that can help answer legal questions could greatly aid laypersons in making informed legal decisions. Existing legal forums, such as Legal Advice Reddit and Law Stack Exchange, are valuable data sources for various legal tasks. On one hand, they provide good sources of labelled data, such as mapping legal questions to their areas of law (for classification), as shown in Figure 1. On the other hand, they contain hundreds of thousands of legal questions that can be leveraged for domain adaptation. Furthermore, questions on these forums can serve as a starting point for tasks that do not have labels found directly in the dataset, such as classifying the severity of a legal question. In this paper, we show that this vast unlabeled corpus can improve performance on question classification, opening up the possibility of studying other tasks on these public legal forums.



...got stopped at Safeway. **Wasn't shoplifting either time...**
... got braced a couple of months ago... then they checked the time stamp on my, uh, advisement... I went f***ing nuclear on them....

What EU law regulates the service providers liabilities and responsibilities for user posted content? ...America usually uses the **DMCA** or the **fair-use policy**...

Figure 1: Example classification task using legal questions from Legal Advice Subreddit (top) and Law Stack Exchange (bottom). Reddit data is generally more informal than Stack Exchange.

In the past few years, large language models have shown effectiveness in legal tasks (Chalkidis et al., 2022). A widespread method used to train these models is finetuning. Although finetuning is very effective, it is prohibitively expensive; training all the parameters requires large amounts of memory and requires a full copy of the language model to be saved for each task. Recently, prefix tuning (Li and Liang, 2021; Liu et al., 2022) has shown great promise by tuning under 1% of the parameters and still achieving comparable performance to finetuning. Unfortunately, prefix tuning performs poorly in low-data (i.e., fewshot) settings (Gu et al., 2022), which are common in the legal

Proceedings of the Natural Legal Language Processing Workshop 2022, pages 119 - 129
December 8, 2022 ©2022 Association for Computational Linguistics

domain. Conveniently, domain adaptation using large public datasets is an ideal setting for the legal domain with abundant unlabelled data (from public forums) and limited labelled data. To this end, we introduce prefix domain adaptation, which performs domain adaptation for prompt tuning to improve fewshot performance on various legal tasks.

Overall, our main contributions are as follows:

- We introduce prefix adaptation, a method of domain adaptation using a prompt-based learning approach.
- We show empirically that performance and calibration of prefix adaptation matches or exceeds LEGAL-BERT in fewshot settings while only tuning approximately 0.1% of the model parameters.
- We contribute two new datasets to facilitate different legal NLP tasks on the questions asked by laypersons, towards the ultimate objective of helping make legal services more accessible to the public.

2 Related Works

Forums-based Datasets Public forums have been used extensively as sources of data for machine learning. Sites like Stack Overflow and Quora have been used for duplicate question detection (Wang et al., 2020; Sharma et al., 2019). Additionally, many prior works have used posts from specific sub-communities (called a "subreddit") on Reddit for NLP tasks, likely due to the diversity of communities and large amount of data provided. Barnes et al. (2021) used a large number of internet memes from multiple meme-related subreddits to predict how likely a meme is to be popular. Other works, such as Basaldella et al. (2020), label posts from biomedical subreddits for biomedical entity linking. Similar to the legal judgement prediction task, Lourie et al. (2021) suggest using "crowdsourced data" from Reddit to perform ethical judgement prediction; that is, they use votes from the "r/AmITheAsshole" subreddit to classify who is "in the wrong" for a given real-life anecdote. We explore using data from Stack Exchange and Reddit, which has been vastly underexplored in previous works for the legal domain.

Full Domain Adaptation Previous works such as BioBERT (Lee et al., 2019) and SciBERT (Beltagy et al., 2019) have shown positive results while domain adapting models. In the industry, companies often use full domain adaptation for legal applications [1]. Chalkidis et al. (2020) introduce LEGAL-BERT, a BERT-like model domain adapted for legal tasks. They show improvements across various legal tasks by training on a domain-specific corpus. Zheng et al. (2021) also perform legal domain adapation, using the Harvard Law case corpus, showing better performance in the CaseHOLD multiple-choice question answering task. Unlike existing works, we perform domain adaptation parameter-efficiently, showing similar performance in a few-shot setting. We compare our approach against LEGAL-BERT as a strong baseline.

Parameter-efficient Learning Language models have scaled to over billions of parameters (He et al., 2021; Brown et al., 2020), making research memory and storage intensive. Recently, parameter-efficient training methods—techniques that focus on tuning a small percentage of the parameters in a neural network—have been a prominent research topic in natural language processing. More recently, prefix tuning (Li and Liang, 2021) has attracted much attention due to its simplicity, ease of implementation, and effectiveness. In this paper, we use P-Tuning v2 (Liu et al., 2022), which includes an implementation of prefix tuning.

Previously, Gu et al. (2022) explored improving prefix tuning's fewshot performance with pre-training by rewriting downstream tasks for a multiple choice answering task (in their "unified PPT"), and synthesizing multiple choice pre-training data (from OpenWebText). Unlike them, we focus on domain adaptation and not general pre-training. We show a much simpler method of prompt pre-training using the masked language modelling (MLM) task while preserving the format of downstream tasks. Ge et al. (2022) domain adapt continuous prompts (not prefix tuning) to improve performance with vision-transformer models for different image types (e.g., "clipart", "photo", or "product").

Zhang et al. (2021) domain adapt an adapter (Houlsby et al., 2019), which is another type of parameter-efficient training method where small neural networks put between layers of the large language model are trained. Vu et al. (2022) explored the transferability of prompts between tasks. They trained a general prompt for the "prefix LM"

[1] https://vectorinstitute.ai/2020/04/02/how-thomson-reuters-uses-nlp-to-enable-knowledge-workers-to-make-faster-and-more-accurate-business-decisions/

(Raffel et al., 2020) objective on the Colossal Clean Crawled Corpus (Raffel et al., 2020). They do not study the efficacy of their general-purpose prompt in fewshot scenarios. Though we use a similar unsupervised language modelling task (Devlin et al., 2019), we aim to train a domain adapted prompt and not a general-purpose prompt.

3 Background

Legal Forums Seeking legal advice from a lawyer can be incredibly expensive. However, public legal forums are incredibly accessible to laypersons to ask legal questions. One popular community is the Legal Advice Reddit community (2M+ members), where users can freely ask personal legal questions. Typically, the questions asked on the Legal Advice Subreddit are written informally and receive informal answers. Another forum is the Law Stack Exchange, a community for questions about the law. Questions are more formal than on Reddit. Additionally, users are not allowed to ask about a specific case and must ask about law more hypothetically, as specified in the rules.

In particular, data from the Legal Advice Subreddit is especially helpful in training machine learning models to help laypersons in law, as questions are in the format and language that regular people would write in (see Figure 1). We run experiments on Law Stack Exchange (LSE) for comprehensiveness, though we believe that the non-personal nature of LSE data makes it less valuable than Reddit data in helping laypersons.

Prefix Tuning As language models grow very large, storage and memory constraints make training impractical or very expensive. Deep prefix tuning addresses these issues by prepending continuous prompts to the transformer. These continuous prefix prompts, which are prepended to each attention layer in the model, and a task-specific linear head (such as a classification head) are trained.

More formally, for each attention layer L_i (as per Vaswani et al., 2017) in BERT's encoder, we append some trainable prefixes P_k (trained key prefix) and P_v (trained value prefix) with length n to the key and value matrices for some initial prompts:

$$\begin{aligned} L_i = \mathbf{Attn}(&xW_q^{(i)}, \\ &Cat(P_k^{(i)}, xW_k^{(i)}), \\ &Cat(P_v^{(i)}, xW_v^{(i)})) \end{aligned} \tag{1}$$

With $W_{\{q,k,v\}}^{(i)}$ representing the respective query, key, or value matrices for the attention at layer i, and x denoting the input to layer i. Here, we assume single-headed attention for simplicity. Here, the Cat function concatenates the two matrices along the dimension corresponding to the sequence length.

Note that in Equation 1 we do not need to left-pad any query values, as the shape of the query matrix does not need to match that of the key and value matrices.

Expected Calibration Error First suggested in Pakdaman Naeini et al. (2015) and later used for neural networks in Guo et al. (2017), expected calibration error (ECE) can determine how well a model is calibrated. In other words, ECE evaluates how closely a model's logit weights reflect the actual accuracy for that prediction. Calibration is important for two main reasons. First, having a properly calibrated model reduces misuse of the model; if output logits accurately reflect their real-world likelihood, then software systems using such models can better handle cases where the model is uncertain. Second, better calibration improves the interpretability of a model as we can better understand how confident a model is under different scenarios (Guo et al., 2017). Bhambhoria et al. (2022) used ECE in the legal domain, where it is especially important due the high-stakes nature of legal decision making.

4 Methods

Here we outline our approach and other baselines for comparison.

RoBERTA To establish a baseline, we train RoBERTa (Liu et al., 2019) for downstream tasks using full model tuning (referred to as "full finetuning"). In addition to the state of the art performance that RoBERTa achieves in many general NLP tasks, it has also shown very strong performance in legal tasks (Shaheen et al., 2020; Bhambhoria et al., 2022). Unlike some transformer models, RoBERTa has an encoder-only architecture, and is normally pre-trained on the masked language modelling task (Devlin et al., 2019). We evaluate the model on both of its size variants, RoBERTa-base (approximately 125M parameters) and RoBERTa-large (approximately 335M parameters).

LEGAL-BERT We evaluate the effectiveness of our approach against LEGAL-BERT, a fully

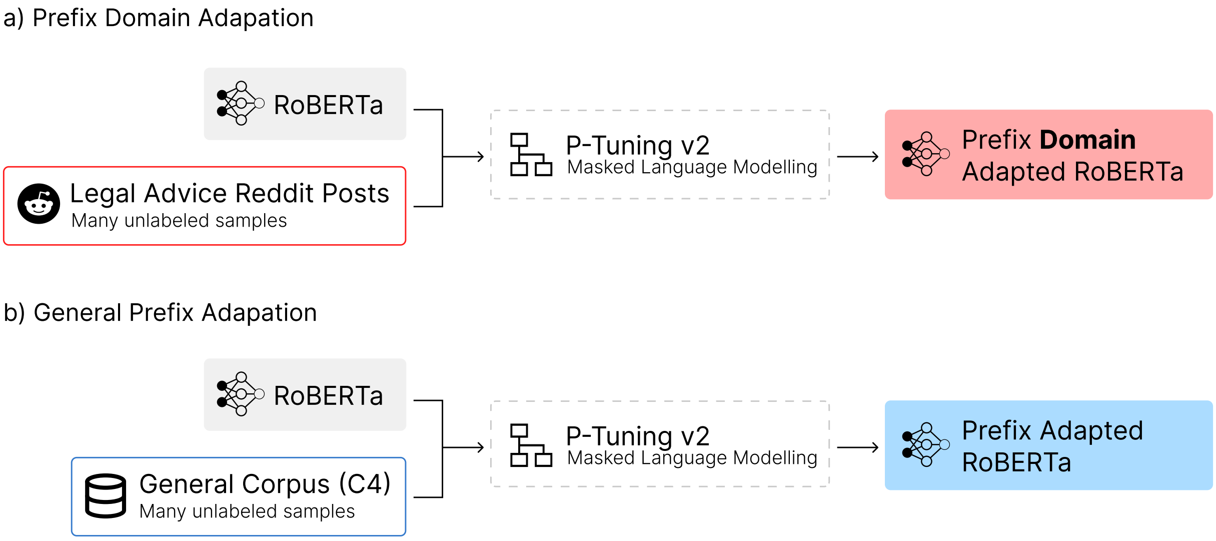

a) Prefix Domain Adaptation
RoBERTa
Legal Advice Reddit Posts
Many unlabeled samples
P-Tuning v2
Masked Language Modelling
Prefix **Domain** Adapted RoBERTa
b) General Prefix Adaptation
RoBERTa
General Corpus (C4)
Many unlabeled samples
P-Tuning v2
Masked Language Modelling
Prefix Adapted RoBERTa

Figure 2: Training process for our methods, with colored boxes representing model weights, colored outlines representing datasets, and dotted outlines representing training method (in this case, P-Tuning v2). Notice that (a) prefix domain adapation and (b) prefix adaptation both use the same starting model and training method, but different datasets.

domain-adapted version of BERT for the legal domain (Chalkidis et al., 2020). In our experiments, we further perform full finetuning for each downstream task. The number of parameters in LEGAL-BERT (109M) is comparable to RoBERTa-base (125M), as used in our other experiments.

Full Domain Adaptation We also perform full domain adaptation by pre-training all model parameters using the masked language modelling (MLM) task with text from each dataset. Then, we train this fully domain adapted model using full-model tuning for each downstream task. This method is a strong baseline for comparison, as we tune all model parameters twice (MLM pre-training and downstream task) for each task, taking up many computational resources.

P-Tuning v2 We compare our approach against P-Tuning v2 (Liu et al., 2022), an "alternative to finetuning" that only optimizes a fractional percentage of parameters (0.1%-3%). It works by freezing the entire model, then appending some frozen prompts in each layer. That is, trainable prompts are added as prefixes to each layer, with only the key and value matrices of the self-attention mechanism trained. We use P-Tuning v2 as a baseline, being the original parameter-efficient training method that we base our study on.

Prefix Domain Adaptation Inspired by domain adaptation, we introduce *prefix domain adaptation*, which domain adapts a deep prompt (Li and Liang, 2021) to better initialize it for downstream tasks. As the domain adapted deep prompt is very small (approximately 0.1% the size of the base model), it is easy to store and distribute. Once trained, the deep prompt is used as a starting point for downstream tasks.

More specifically, we train a deep prompt, using prefix tuning as in Liu et al. (2022)[2], for the masked language modelling task (Devlin et al., 2019) on a large, domain-specific unsupervised corpus, as shown in Figure 2(a). Next, we use this pre-trained prompt and randomly initialize a task-specific head (such as a classification head for a classification task) for each downstream task. Finally, we train the resulting model for the downstream task, using the same prompt tuning approach from Liu et al. (2022). To the best of our knowledge, no prior works have trained a prefix prompt for a specific domain to better initialize it for downstream tasks using an unsupervised pre-training task (masked language modelling).

Formally, we can treat a prefix-tuned model as having a trained prefix P, and a trained task-specific head H. We group each downstream task into m domains in $\{\mathcal{D}_1, \mathcal{D}_2, ..., \mathcal{D}_m\}$, such that there is some overlap between the tasks in each domain $\mathcal{D}_i$. For each domain $\mathcal{D}_i$, we use a domain-specific corpus, C_i, to train some prefix P_i for the masked language modelling task with prompt

[2] Same implementation as provided in Liu et al. (2022)

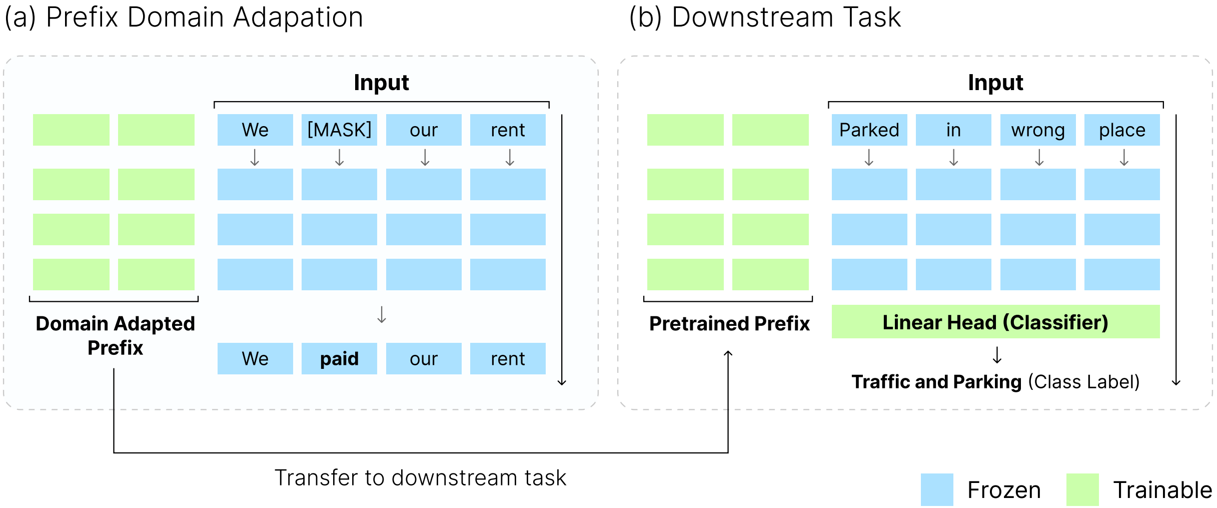


Figure 3: Toy example of how our framework works. (a) We pre-train a prefix on the unsupervised masked language modelling task. (b) We randomly initialize the classification head but preserve the prefix for downstream tasks. Green blocks represent trainable prompt embeddings or layers of the transformer, and blue blocks represent frozen embeddings or computations.

tuning (Figure 3(a)). Then, for each downstream task in $\mathcal{D}_i$, we use the deep prefix P_i to initialize the prompts, while randomly initializing the task-specific head H_i (Figure 3(b)).

Prefix Adaptation In addition to prefix domain adaptation, we conduct experiments using our approach in general settings, inspired by work done in Vu et al. (2022) and Gu et al. (2022). We name this more general approach *prefix adaptation*. That is, we test the performance of initializing a prompt with the masked language modelling task on a subset of the Colossal Clean Crawled Corpus (Raffel et al., 2020), instead of domain-specific texts (illustrated in Figure 2(b)). Formally, we use the same prefix domain adaptation approach as previously mentioned, but we group all tasks under one "General" domain $\mathcal{D}$, and thus only train one prefix P.

5 Datasets

We evaluate each of the approaches listed above on three different datasets.

Legal Advice Reddit We introduce a new dataset from the Legal Advice Reddit community (known as "/r/legaldvice"), sourcing the Reddit posts from the Pushshift Reddit dataset (Baumgartner et al., 2020) [3]. The dataset maps the text and title of each legal question posted into one of eleven classes, based on the original Reddit post's "flair" (i.e., tag). Questions are typically informal and use non-legal-specific language. Per the Legal Advice Reddit rules, posts must be about actual personal circumstances or situations. We limit the number of labels to the top eleven classes and remove the other samples from the dataset (more details in Appendix B). To prefix adapt the model for Reddit posts, we use samples from the Legal Advice sub-reddit that are not labelled or do not fall under the top eleven classes. We use the provided "flair" for each question for a legal area classification task (Soh et al., 2019), as illustrated in Figure 1.

European Court of Human Rights We use the European Court of Human Rights (ECHR) dataset (Chalkidis et al., 2019), which consists of a list of facts specific to a legal case, labelled with violated human rights articles (if any). Specifically, we evaluate our approach on the binary violation prediction task, where the task is to predict whether a given case violates any human rights articles given a list of facts. We undersample this relatively large dataset to simulate a fewshot learning environment. To prefix adapt the model for ECHR cases, we use the original corpus of unlabelled cases (similar to what was done in Chalkidis et al., 2020). As the average document length is 700 words (above BERT's maximum length limit), we truncate the text to 500 tokens, concatenating the title and facts of the case together.

Law Stack Exchange We also introduce a second dataset with data from the Law Stack Exchange

[3] https://huggingface.co/datasets/jonathanli/legal-advice-reddit

Dataset Name	N_{class}	Fewshot Sizes
ECHR	2	4, 8, 16, 32
Legal Advice Reddit	11	32, 64, 128, 256
Law Stack Exchange	16	32, 64, 128, 256

Table 1: Classification tasks evaluated in our experiments. N_{class} represents the number of classes, and "Fewshot Sizes" represents the various number of samples used (4 different fewshot sizes evaluated for each dataset).

(LSE)[4]. This dataset is composed of questions from the Law Stack Exchange, which is a community forum-based website containing questions with answers to legal questions. Unlike the Legal Advice Reddit dataset, the Law Stack Exchange dataset is generally more formal (shown in Figure 1), and questions are generally more theoretical or hypothetical. We link the questions with their associated tags (e.g., "copyright" or "criminal-law"), and perform the multi-label classification task. Though posts can have multiple tags, we use the questions with only one tag in the top 16 most frequent tags (excluding tags associated with countries). Similarly to the Legal Advice Reddit dataset, we use other unused questions from the Law Stack Exchange to prefix domain adapt the model.

6 Experimental Setup

We test our approaches under a fewshot setting, where prompt tuning is known to perform poorly (Gu et al., 2022). We use RoBERTa-base and RoBERTa-large (Liu et al., 2019) for our experiments. To simulate a fewshot learning scenario, we randomly undersample the train and validation sets for each dataset, ensuring that the distribution of train and validation data roughly matches. Additionally, we vary the amount of data undersampled to study how fewshot size affects performance. In these tasks, we use a validation size of 256 (much smaller than the original) to represent true fewshot learning better (Perez et al., 2021). Considering that fewshot learning is quite unstable, we ran all of our experiments five times, using the seeds $\{10, 20, 30, 40, 50\}$. We provide more training details in Appendix A.

There is often confusion around whether fewshot sizes represent the number of samples per class or the total number of samples (Perez et al., 2021). In our results, the fewshot sizes we show are the exact number of training samples used (i.e., total training samples). The exact number of samples is listed in Table 1. To keep the number of samples per class roughly equivalent, we use fewer total samples for the ECHR task, which only has two classes.

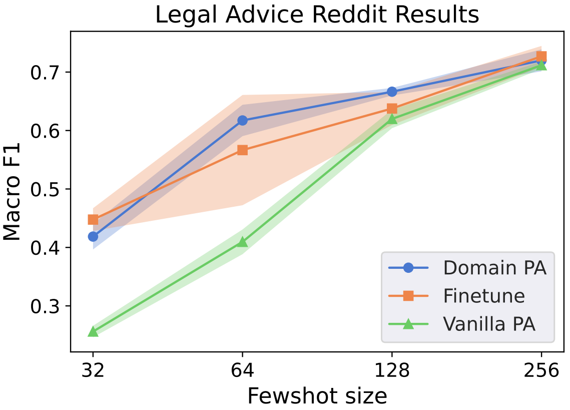


Figure 4: Various fewshot sizes and their performance (measured by macro F1). The shaded region represents the standard deviation across runs, while each point represents the mean performance across runs. Overall, our approach (prefix domain adaptation) matches the performance of full finetuning.

7 Results and Discussion

We make a few observations on our results, shown in Table 2. We observe that our method, prefix domain adaptation, outperforms both regular prefix tuning and full finetuning in most tasks across fewshot sizes, despite training considerably fewer parameters. We find that prefix adaptation is comparable to full domain adaptation; in some settings (such as ECHR and some Reddit fewshot settings), prefix adaptation even outperforms full domain adaptation. We argue that prefix domain adaptation achieves better fewshot performance relative to regular prefix tuning because the pre-trained prompts are closer to an effective prompt after our domain adaptation step. This is similar to full domain adaptation, which improves performance on downstream tasks relative to a base model (Chalkidis et al., 2020) by making parameters closer to optimal parameters. Consistent with Gu et al. (2022), we find that regular prefix tuning falls behind full parameter tuning in fewshot settings.

Additionally, we find that LEGAL-BERT performs worse than other techniques on datasets with more informal language (such as the Reddit

[4] https://huggingface.co/datasets/jonathanli/law-stack-exchange

	Legal Advice Reddit				Law Stack Exchange				European Court of Human Rights			
Fewshot Size	*32*	*64*	*128*	*256*	*32*	*64*	*128*	*256*	*4*	*8*	*16*	*32*
FT	$\mathbf{44.8}_{\mathbf{1.9}}$	$56.7_{9.4}$	$63.8_{2.8}$	$\underline{72.7_{1.8}}$	$19.5_{17.1}$	$29.0_{14.8}$	$\underline{58.8_{0.8}}$	$\underline{67.4_{0.9}}$	$53.7_{1.6}$	$60.1_{5.5}$	$66.5_{8.7}$	$66.3_{3.5}$
LEGAL-BERT + FT	$36.1_{2.9}$	$35.2_{16.1}$	$49.5_{3.7}$	$70.2_{1.7}$	$24.6_{13.1}$	$51.2_{0.9}$	$47.6_{24.9}$	$\mathbf{67.5}_{\mathbf{0.2}}$	$59.3_{12.4}$	$55.8_{3.8}$	$61.1_{8.7}$	$\underline{67.6_{3.6}}$
Domain Adapt + FT	$31.8_{16.4}$	$\mathbf{66.7}_{\mathbf{3.3}}$	$\underline{66.6_{1.5}}$	$\mathbf{75.8}_{\mathbf{0.9}}$	$\mathbf{38.5}_{\mathbf{0.4}}$	$\mathbf{53.2}_{\mathbf{2.5}}$	$\mathbf{62.4}_{\mathbf{1.1}}$	$66.6_{0.6}$	$47.6_{2.3}$	$51.2_{1.7}$	$47.9_{1.3}$	$56.7_{2.2}$
Prefix Domain Adapt	$\underline{41.9_{2.2}}$	$\underline{61.7_{2.7}}$	$\mathbf{66.6}_{\mathbf{0.6}}$	$72.0_{1.9}$	$\underline{36.1_{0.9}}$	$\underline{52.4_{1.7}}$	$56.1_{0.8}$	$63.1_{1.7}$	$\mathbf{72.7}_{\mathbf{4.6}}$	$\underline{70.9_{2.3}}$	$\mathbf{75.1}_{\mathbf{1.8}}$	$\mathbf{69.4}_{\mathbf{2.0}}$
Prefix Adapt	$35.5_{2.1}$	$58.0_{6.4}$	$52.7_{21.4}$	$72.2_{0.5}$	$31.7_{1.2}$	$46.8_{2.5}$	$57.0_{0.8}$	$66.6_{0.5}$	$68.9_{6.4}$	$\mathbf{71.4}_{\mathbf{1.4}}$	$\underline{75.0_{2.7}}$	$66.3_{7.8}$
P-Tuning v2	$25.6_{1.0}$	$41.0_{2.1}$	$62.0_{1.6}$	$71.2_{0.6}$	$24.6_{2.2}$	$45.3_{2.0}$	$56.3_{0.7}$	$65.3_{0.6}$	$\underline{70.9_{2.6}}$	$70.5_{3.6}$	$70.9_{2.3}$	$67.1_{0.9}$

Table 2: Classification results with RoBERTa-base (or similarly sized models), with fewshot size listed as italic numbers in the second row. Experiments run five times with different seeds, with subscripts representing the standard deviation of the five runs. **Bolded** results represent the best performance for the fewshot size, and underlined results represent second best. All methods are assumed to be initialized from RoBERTa-base, except for LEGAL-BERT from Chalkidis et al. (2020). "FT" represents fully finetuned for downstream tasks and "Domain Adapt" is full domain adaptation, with a line separating full-model (top) and parameter efficient (bottom) tuning methods.

Fewshot Size	*32*	*64*	*128*	*256*
FT	$42.1_{5.7}$	$55.5_{5.4}$	$62.0_{3.5}$	$\mathbf{77.6}_{\mathbf{1.0}}$
Domain Adapt + FT	$34.2_{7.3}$	$61.7_{5.6}$	$\underline{66.6_{8.7}}$	$\underline{77.3_{1.3}}$
Prefix Domain Adapt	$\mathbf{46.7}_{\mathbf{2.1}}$	$\mathbf{63.5}_{\mathbf{1.5}}$	$\mathbf{67.0}_{\mathbf{1.7}}$	$72.2_{1.0}$
Prefix Adapt	$46.5_{3.4}$	$\underline{63.1_{2.5}}$	$64.3_{1.8}$	$70.0_{1.9}$
P-Tuning v2	$\underline{46.7_{1.7}}$	$59.0_{1.5}$	$65.6_{2.7}$	$69.2_{1.7}$

Table 3: Classification results on RoBERTa-large, evaluated on Reddit data. Note that we do not evaluate results with LEGAL-BERT because LEGAL-BERT models with comparable size to RoBERTA-large do not exist.

dataset). LEGAL-BERT shows more instability across seeds (i.e., larger standard deviation) . As LEGAL-BERT-SC (the model we use) was only trained on very formal legal text, it did not see many colloquialisms or slang during training that are prevalent in informal text. For this reason, we do not think LEGAL-BERT would be effective as initialization for tasks involving legal questions asked by laypersons, which typically do not use incredibly formal legal language.

In contrast to other datasets, the ECHR dataset's train and test split have different distributions. In fewshot scenarios with very little data (i.e., 4-16 examples), we find that prefix tuning based approaches perform better than full finetuning; this suggests that prefix tuning approaches are more robust to changes in distribution (and possibly noise). We also note that BERT with truncation (maximum token length of 500) performs a lot better than initially reported in Chalkidis et al. (2019), who report an F1 worse than random guessing (macro F1 of 66.5 in ours, 17 in theirs). We believe this underperformance of finetuning BERT could be caused by a mistake in their training process.

In Figure 4, we show the trend of performance on Reddit data as the number of samples increases. Prefix domain adaptation is comparable to finetuning, consistently outperforming regular prefix tuning. As shown by the larger shaded area around the lines, the stability of finetuning is worse than prefix domain adaptation for this task. Performance gradually converges increases as more data is given to each method.

Larger models typically provide better performance on various tasks. Thus, we run experiments using RoBERTa-large (over 2x larger than RoBERTa-base) to see how our approach scales to larger models. As seen in Table 3, our approach is still comparable to or outperforms full finetuning with larger models. Impressively, in the fewshot sizes 32-128, prefix domain adaptation with RoBERTa-base is even comparable to full finetuning with RoBERTa-large. Additionally, we note that full domain adapation is more sensitive to learning rates in larger models, explaining weaker performance in fewshot sizes 32 and 64. Due to limitations in computational resources, we leave more extensive hyperparameter search as future work.

7.1 Calibration

While providing predictions to laypersons, it is vital that the distribution of the output logits accurately reflect the model's confidence. Thus, we use the expected calibration error (ECE) (Pakdaman Naeini et al., 2015) to measure the calibration of each model resulting from each method. We show that the calibration of our approach is better than finetuning across tasks, as seen in Table 4. Additionally, we observe that our approach is comparable to

	Reddit	LSE	ECHR
FT	$0.158_{0.012}$	$0.243_{0.015}$	$0.320_{0.037}$
LEGAL-BERT + FT	$0.454_{0.05}$	$\mathbf{0.165_{0.043}}$	$0.245_{0.042}$
Domain Adapt + FT	$0.152_{0.004}$	$0.214_{0.01}$	$0.320_{0.121}$
Prefix Domain Adapt	$0.133_{0.01}$	$0.242_{0.023}$	$\mathbf{0.214_{0.032}}$
Prefix Adapt	$\mathbf{0.104_{0.021}}$	$0.24_{0.008}$	$0.266_{0.063}$
P-Tuning v2	$0.412_{0.019}$	$0.263_{0.009}$	$0.253_{0.050}$

Table 4: Calibration, measured by the top-1 expected calibration error (ECE). "Reddit" is the ECE on our Legal Advice Reddit dataset (fewshot size of 256), "ECHR" is ECE on European Court of Human Rights dataset (fewshot size of 32), and "LSE" is ECE on the Law Stack Exchange dataset (fewshot size of 256). Lower is better, with **bold** being the best and underline being second best.

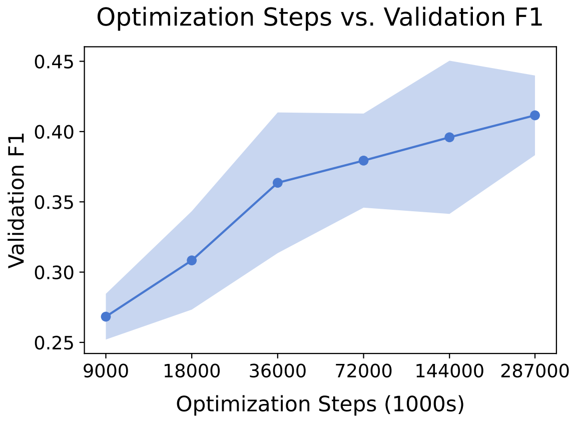


Figure 5: Performance of prefix domain adaptation after training the domain adapted prompt for a different number of training steps, performed on Reddit data with a fewshot size of 32 using RoBERTa-base. Shaded region represents standard deviation between five runs.

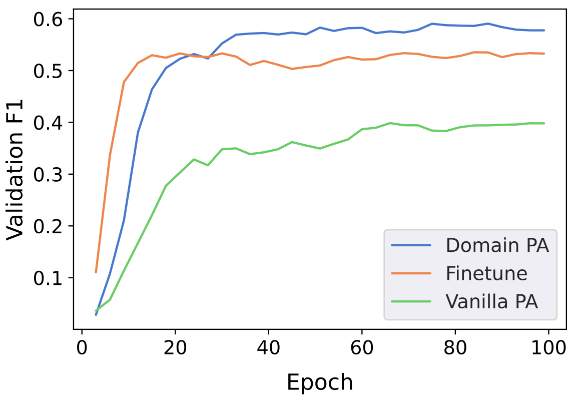


Figure 6: Convergence comparison of prefix domain adaptation ("Domain PA"), full finetuning ("Finetune"), and P-Tuning v2 on Reddit data, using a fewshot size of 64.

LEGAL-BERT across tasks. In the case where questions are well formulated (i.e., in the LSE dataset), we found that legal models are better calibrated. However, in Reddit data, which is central to helping laypersons with legal questions, we find that our approach is very competitive.

7.2 Sample Efficiency

We study the effect of training time (i.e., number of training steps) for the domain-adapted prompt on downstream performance. To analyze the effect of additional training steps on the domain adapted prefix's performance, we initialize models using pre-trained prefixes from specific steps and plot the performance (over five runs) in Figure 5. We find that more optimization steps during the prefix adaptation step lead to better downstream performance. Intuitively, this makes sense as a longer training time means the prefix starts closer to an ideal one for a downstream task.

Though each optimization step is faster with regular prefix tuning (Gu et al., 2022), it converges slowly and thus is not necessarily faster than finetuning. As shown in Figure 6, our approach converges faster than regular prefix tuning. Again, we argue that this is expected as the prompts are closer to a desired solution when compared to regular prefix tuning, meaning fewer training steps are needed to reach an effective solution.

8 Conclusions

In this paper, we propose a novel training framework, *prefix domain adaptation*, aiming to domain adapt a prompt using a large corpus of domain-specific text. We show that our approach matches or outperforms LEGAL-BERT or related techniques in performance while training fewer (0.1%) parameters. With our technique, we improve fewshot performance and convergence time compared to other parameter-efficient methods. We believe this will make fewshot data more usable (and thus reduce data labelling costs) while using parameter-efficient methods to reduce computational and storage costs.

Additionally, we introduce two new datasets (Legal Advice Reddit and Law Stack Exchange) to lay foundations for future work in legal decision-making systems; as opposed to formal documents in ECHR, our two datasets are closer to legal questions asked by laypersons, helping to promote access to justice for all.

References

Kate Barnes, Tiernon Riesenmy, Minh Duc Trinh, Eli Lleshi, Nora Balogh, and Roland Molontay. 2021. Dank or not? analyzing and predicting the popularity of memes on reddit. *Applied Network Science*, 6(1):21.

Marco Basaldella, Fangyu Liu, Ehsan Shareghi, and Nigel Collier. 2020. COMETA: A corpus for medical entity linking in the social media. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3122–3137, Online. Association for Computational Linguistics.

Jason Baumgartner, Savvas Zannettou, Brian Keegan, Megan Squire, and Jeremy Blackburn. 2020. The pushshift reddit dataset.

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.

Rohan Bhambhoria, Hui Liu, Samuel Dahan, and Xiaodan Zhu. 2022. Interpretable low-resource legal decision making. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(11):11819–11827.

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.

Ilias Chalkidis, Ion Androutsopoulos, and Nikolaos Aletras. 2019. Neural legal judgment prediction in English. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4317–4323, Florence, Italy. Association for Computational Linguistics.

Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. LEGAL-BERT: The muppets straight out of law school. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2898–2904, Online. Association for Computational Linguistics.

Ilias Chalkidis, Abhik Jana, Dirk Hartung, Michael Bommarito, Ion Androutsopoulos, Daniel Katz, and Nikolaos Aletras. 2022. LexGLUE: A benchmark dataset for legal language understanding in English. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4310–4330, Dublin, Ireland. 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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Chunjiang Ge, Rui Huang, Mixue Xie, Zihang Lai, Shiji Song, Shuang Li, and Gao Huang. 2022. Domain adaptation via prompt learning.

Yuxian Gu, Xu Han, Zhiyuan Liu, and Minlie Huang. 2022. PPT: Pre-trained prompt tuning for few-shot learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8410–8423, Dublin, Ireland. Association for Computational Linguistics.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 132–1330. PMLR.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp. In *International Conference on Machine Learning*, pages 2790–2799. PMLR.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. 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 4582–4597, Online. Association for Computational Linguistics.

Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. 2022. P-tuning: Prompt tuning can be comparable to fine-tuning across scales and tasks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 61–68, Dublin, Ireland. 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.

Nicholas Lourie, Ronan Le Bras, and Yejin Choi. 2021. Scruples: A corpus of community ethical judgments on 32,000 real-life anecdotes. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(15):13470–13479.

Mahdi Pakdaman Naeini, Gregory Cooper, and Milos Hauskrecht. 2015. Obtaining well calibrated probabilities using bayesian binning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 29(1).

Ethan Perez, Douwe Kiela, and Kyunghyun Cho. 2021. True few-shot learning with language models.

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.

Zein Shaheen, Gerhard Wohlgenannt, and Erwin Filtz. 2020. Large scale legal text classification using transformer models.

Lakshay Sharma, Laura Graesser, Nikita Nangia, and Utku Evci. 2019. Natural language understanding with the quora question pairs dataset.

Jerrold Soh, How Khang Lim, and Ian Ernst Chai. 2019. Legal area classification: A comparative study of text classifiers on Singapore Supreme Court judgments. In *Proceedings of the Natural Legal Language Processing Workshop 2019*, pages 67–77, Minneapolis, Minnesota. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need.

Tu Vu, Brian Lester, Noah Constant, Rami Al-Rfou', and Daniel Cer. 2022. SPoT: Better frozen model adaptation through soft prompt transfer. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5039–5059, Dublin, Ireland. Association for Computational Linguistics.

Liting Wang, Li Zhang, and Jing Jiang. 2020. Duplicate question detection with deep learning in stack overflow. *IEEE Access*, 8:25964–25975.

Rongsheng Zhang, Yinhe Zheng, Xiaoxi Mao, and Minlie Huang. 2021. Unsupervised domain adaptation with adapter.

Lucia Zheng, Neel Guha, Brandon R. Anderson, Peter Henderson, and Daniel E. Ho. 2021. When does pretraining help? assessing self-supervised learning for law and the casehold dataset of 53,000+ legal holdings. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Law*, ICAIL '21, page 159–168, New York, NY, USA. Association for Computing Machinery.

Configuration	Learning Rates
RoBERTa-base PT	5e-2, 3e-2, 2e-2, 5e-3, 5e-4
RoBERTa-large PT	5e-2, 3e-2, 2e-2, 5e-3, 5e-4
RoBERTa-base FT	1e-3, 5e-4, 2e-4, 1e-4, 5e-5
RoBERTa-large FT	1e-4, 5e-5, 2e-5, 1e-5, 5e-6

Table 5: Learning rates searched for each configuration. The suffix "PT" means for prompt tuning based methods, and "FT" for finetuning based methods.

A Additional Training Details

We use the AdamW optimizer and a grid search of learning rates as in Table 5, mostly following Gu et al. (2022). For all of our experiments, we truncate the sequence to a length of 500 tokens (as opposed to 512 tokens) to allow space for a tuned deep prefix prompt. We report the calibration and general results using the checkpoint with the best validation macro F1, for each fewshot size and method.

Given that RoBERTa-base (~125M parameters) and RoBERTa-large (~355M parameters) can fit in a single NVIDIA 1080Ti GPU (using a smaller batch size), we do not perform any model or data parallelism. We use an effective batch size (i.e., factoring in gradient accumulation steps) of 32 for experiments on roberta-base, and due to memory constraints, an effective batch size of 24 for experiments on roberta-large. As the number of samples is low, we train for 100 epochs. However, while performing domain adaptation and prefix adaptation training steps, we train for 20 epochs as much more data as available (and therefore, more optimization steps are run in each epoch).

We use a prefix length of 8. Including the tuned linear head for classification, the largest number of parameters we tune for RoBERTa-base is 160K (varies slightly for each task depending on the number of classes), or ~0.13% of the model's parameters.

Dataset Name	N_{train}	N_{dev}	N_{test}	Avg. Words
ECHR	7100	2998	1380	2105_{2489}
Legal Advice Reddit	9887	9987	79136	145_{117}
Law Stack Exchange	638	319	1596	244_{217}

Table 6: Sizes of datasets. $N_{train,dev,test}$ represent sizes of the train, development, and test sets respectively.

B Data Details

For Reddit data we take the top 11 classes that are not countries. We concatenate the title of the Reddit post and body text together, then use this combination to train our models for the masked language modelling and flair classification task.

For Stack Exchange data, we take only the questions with a single tag, and again. The stack exchange data, taken from Internet Archive[5], includes the post body in an HTML form. As our base models were not trained on HTML formatted text, we convert the HTML to Markdown footnote to make it much more similar to human readable text.

For the ECHR dataset, we use the non-anonymized variant and concatenate the title of the case with each fact from the legal case. Additionally, we found that some documents had numbered facts (such as "**1.** <fact>"), while some documents were not numbered. We used a simple regular expression to remove this inconsistency which could possibly create biases in the model (e.g., if numbered facts were more likely to mean a violation).

In our domain adaptation experiments, we use all the data (i.e., including questions/posts that were previously filtered out because they didn't have top tags) for each dataset. We use the domain adapated checkpoint with the best validation cross-entropy loss for downstream tasks.

The sizes of each split are listed in Table 6. Test split sizes for the Reddit and Stack Exchange dataset are intentionally larger than the validation and training set to better simulate true fewshot learning, as per Perez et al. (2021).

[5] https://archive.org/download/stackexchange