Domain Knowledge Transferring for Pre-trained Language Model via Calibrated Activation Boundary Distillation

Dongha Choi[1], **HongSeok Choi**[2], and **Hyunju Lee**[1,2*]
[1]Artificial Intelligence Graduate School
[2]School of Electrical Engineering and Computer Science
Gwangju Institute of Science and Technology, Gwangju 61005, South Korea
dongha528@gm.gist.ac.kr, {hongking9,hyunjulee}@gist.ac.kr

Abstract

Since the development and wide use of pre-trained language models (PLMs), several approaches have been applied to boost their performance on downstream tasks in specific domains, such as biomedical or scientific domains. Additional pre-training with in-domain texts is the most common approach for providing domain-specific knowledge to PLMs. However, these pre-training methods require considerable in-domain data and training resources and a longer training time. Moreover, the training must be re-performed whenever a new PLM emerges. In this study, we propose a domain knowledge transferring (DoKTra) framework for PLMs without additional in-domain pre-training. Specifically, we extract the domain knowledge from an existing in-domain pre-trained language model and transfer it to other PLMs by applying knowledge distillation. In particular, we employ activation boundary distillation, which focuses on the activation of hidden neurons. We also apply an entropy regularization term in both teacher training and distillation to encourage the model to generate reliable output probabilities, and thus aid the distillation. By applying the proposed DoKTra framework to downstream tasks in the biomedical, clinical, and financial domains, our student models can retain a high percentage of teacher performance and even outperform the teachers in certain tasks. Our code is available at https://github.com/DMCB-GIST/DoKTra.

1 Introduction

Recently, transformer (Vaswani et al., 2017)-based language models have been successfully applied in the field of natural language processing (NLP). In particular, the two-stage approach of "pre-training and fine-tuning," such as BERT (Devlin et al., 2019), has become the standard for NLP applications. Generally, a transformer-based model is pre-trained with a large amount of text data in an unsupervised manner, and then fine-tuned with a small dataset for several downstream tasks. Further, advanced pre-trained language models (PLMs) with improved architectures or training methods continue to emerge, including ALBERT (Lan et al., 2019) or RoBERTa (Liu et al., 2019).

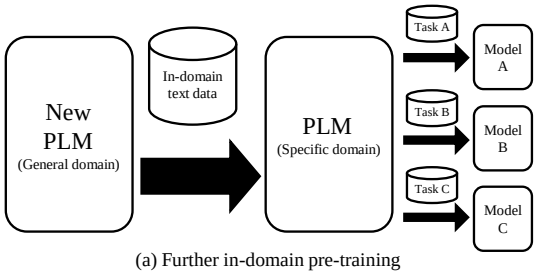
However, these models must be further improved for tasks requiring domain knowledge, such as those in the biomedical or financial domains, as the pre-training data usually consist of general domain text (e.g., Wikipedia). Additional pre-training with in-domain text has been proposed to provide the PLMs with domain-specific knowledge. For example, in the biomedical domain, several domain-specific PLMs trained with large biomedical texts, such as BioBERT (Lee et al., 2020), PubMedBERT (Gu et al., 2020) and BlueBERT (Peng et al., 2019), have been successfully used as strong baselines for several downstream tasks. Nevertheless, additional pre-training has several limitations, such as the need for sufficient training data and resources, and a longer training time. Furthermore, whenever a new PLM emerges, it must be re-trained to create more advanced domain-specific models.

To address this issue, we propose an efficient domain-knowledge transferring framework that does not require additional pre-training steps. Specifically, we focus on the applicability of knowledge distillation (Hinton et al., 2015) as a domain-knowledge transfer method, not only for model compression. Knowledge distillation is a well-known knowledge transfer method that is primarily used for model compression. The knowledge from a larger and more effective teacher model is distilled to a smaller student model by encouraging it to mimic the teacher characteristics, such as soft probabilities (Hinton et al., 2015) or hidden representations (Kim et al., 2018; Sun et al., 2019).

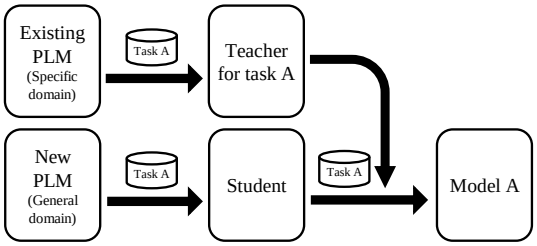
In this study, we propose a domain knowledge transfer (DoKTra) framework for an advanced PLM via calibrated activation boundary distilla-

*Hyunju Lee is the corresponding author.

Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 1658 - 1669



(a) Further in-domain pre-training



(b) DoKTra framework

Figure 1: Comparison between (a) an existing domain transfer method and (b) a proposed framework. The thickness of the arrow is proportional to the required training time.

tion. In contrast to the existing in-domain pre-training methods, we transfer domain knowledge to a new language model using only an existing in-domain pre-trained model, and without a time-consuming pre-training on the new model. For instance, BioBERT was pre-trained for 23 days on 8 NVIDIA V100 GPUs (Lee et al., 2020). We can estimate that if a new, larger language model is pre-trained with a large number of biomedical texts, its training duration would be longer than that of BioBERT. However, our framework can be executed in a few hours on a single 24 GB GPU. The comparison between our framework and a conventional approach is visualized in Figure 1.

Specifically, we apply the calibration method to generate a reliable and well-supervising teacher model. Then, we apply activation boundary distillation (Heo et al., 2019) to distill the domain knowledge to the student, which is more efficient with a small amount of training data. Moreover, by selecting language models more advanced than the teacher as students, we allow the student models to acquire additional domain knowledge while preserving its superiority.

We apply our framework to the biomedical domain and verify its effectiveness by conducting experiments on several biomedical and clinical downstream tasks. Consequent to applying our framework to ALBERT and RoBERTa student models, we were able to obtain models that retained most of the teacher model's performance with fewer model parameters (ALBERT), and models with a higher performance than both students and teachers (RoBERTa). We also investigate the general applicability of our framework by applying it to a financial domain PLM and downstream tasks. The contributions of this study can be summarized as follows:

- We propose a DoKTra framework for advanced PLM via calibrated activation boundary distillation, without additional time-consuming pre-training steps.
- We conduct experiments to demonstrate the efficacy of DoKTra, resulting in obtaining the student models that retain most of the performances of the teacher model while utilizing fewer parameters or achieve even higher performances than the teacher model.

2 Related Work

2.1 Pre-trained language model (PLM)

Most modern language models are based on the transformer (Vaswani et al., 2017) architecture. The PLMs generally use only the encoder block of the transformer, which consists of two sublayers: a self-attention layer and a feed-forward layer. BERT (Devlin et al., 2019) is the most widely used PLM, which consists of several layers of transformer encoders. It was pre-trained for 4 days with a large amount of text data, which consisted of 3.3 billion words, using masked language modeling and a next sentence prediction task in an unsupervised manner. This pre-trained model can be easily used in various downstream tasks by fine-tuning it with a labeled dataset. Following the success of BERT, a variety of similar PLMs have emerged. Lan et al. (2019) proposed ALBERT, which outperformed BERT with considerably fewer parameters. ALBERT's architecture is more complex than BERT's; however, by applying factorized embedding parameterization and cross-layer parameter sharing, the number of parameters can be reduced. Liu et al. (2019) observed that BERT is significantly undertrained, and proposed RoBERTa, a more robust and better-performing model, which is obtained by a longer pre-training with a larger dataset (approximately 10 times that of BERT) and the removal of next sentence prediction.

2.2 Domain knowledge transferring for PLMs

Despite the PLMs' excellent performances in several downstream tasks in the general domain, they have not exhibited a superior performance in specific domain tasks, such as in biomedicine. To provide domain-specific knowledge to PLMs, additional pre-training with in-domain data has been applied. BioBERT (Lee et al., 2020) further pre-trained BERT using biomedical text consisting of 18 billion words, such as literature abstracts. Peng et al. (2019) applied a similar approach with both biomedical and clinical text data. Differently, Gu et al. (2020) pre-trained BERT from scratch with only biomedical literature.

3 DoKTra framework

In this section, we introduce the DoKTra framework, which is the main approach to transfer domain-specific knowledge.

3.1 Overview

The main goal of the DoKTra framework is to produce a task-specific student model for each downstream task in a specific domain by distilling domain knowledge from a fine-tuned teacher model. Our framework consists of two main stages: calibrated teacher training and activation boundary distillation.

In calibrated teacher training, the teacher model is trained to distil its domain-specific and task-specific knowledge into the student model. We use an existing in-domain PLM as the initial teacher model. For each downstream task in the initial teacher's domain, the teacher model is fine-tuned with its training data. In this process, an entropy regularization term, called the confidence penalty loss (Pereyra et al., 2017), is added to the training loss. By adding the confidence regularizer, the fine-tuned teacher model can generate more reliable output prediction probabilities for the input data, and thus, have a positive effect on distillation.

In activation boundary distillation, the domain-specific knowledge of the teacher model is transferred to the student model. We use an existing PLM as the initial student model, which is only pre-trained in the general domain. First, the student model is fine-tuned for a downstream task. Subsequently, it mimics the activation pattern of the hidden neurons in the teacher model (Heo et al., 2019). By distilling the activation pattern, the activation boundary of the teacher model is transferred more precisely, and the domain-specific knowledge of the teacher is transferred to the student model. Additionally, the student model is refined over fewer epochs with a standard classification loss (Romero et al., 2014; Yim et al., 2017; Heo et al., 2019). Because the student model is already fine-tuned for the downstream task, any additional refinement may result in overconfidence (Guo et al., 2017; Nixon et al., 2019). To address this issue, we also add the confidence regularizer to the refinement step. The proposed framework is visualized in Figure 2.

3.2 Calibrated teacher training

In this step, a task-specific teacher model is generated for each in-domain downstream task using a fine-tuning approach. Specifically, we choose BioBERT-base (Lee et al., 2020) as the initial teacher model, which has been pre-trained with a large biomedical domain corpus, such as PubMed abstracts. Owing to the in-domain pre-training, the BioBERT model outperforms the BERT model in several biomedical downstream tasks.

Despite their high performance, modern deep neural networks are not well calibrated (Guo et al., 2017), which is similar to language models such as BERT. In other words, these models only predict overconfidently and cannot generate a reliable output probability for the given input. However, most distillation approaches encourage the use of softened probability because they contain more information and can better support the learning of the student model (Hinton et al., 2015; Cho and Hariharan, 2019). Moreover, Menon et al. (2021) demonstrated that a teacher model that estimates "good" probabilities can better supervise a student model. Based on this idea, we apply an entropy-regularizing term that penalizes overconfidence when fine-tuning the teacher model (Pereyra et al., 2017). Several previous studies have revealed that a confidence penalty improves both the calibration and performance of biomedical downstream tasks (Choi and Lee, 2020).

Since an overconfident classification model produces output probabilities close to 0 and 1, its probability distribution has a low entropy value. The confidence penalty loss (CPL) addresses this problem by minimizing the negative entropy of the output probability. Formally, the output probability of the model with parameters θ can be written as a conditional distribution $p_\theta(\mathbf{y}|\mathbf{x})$ through the soft-

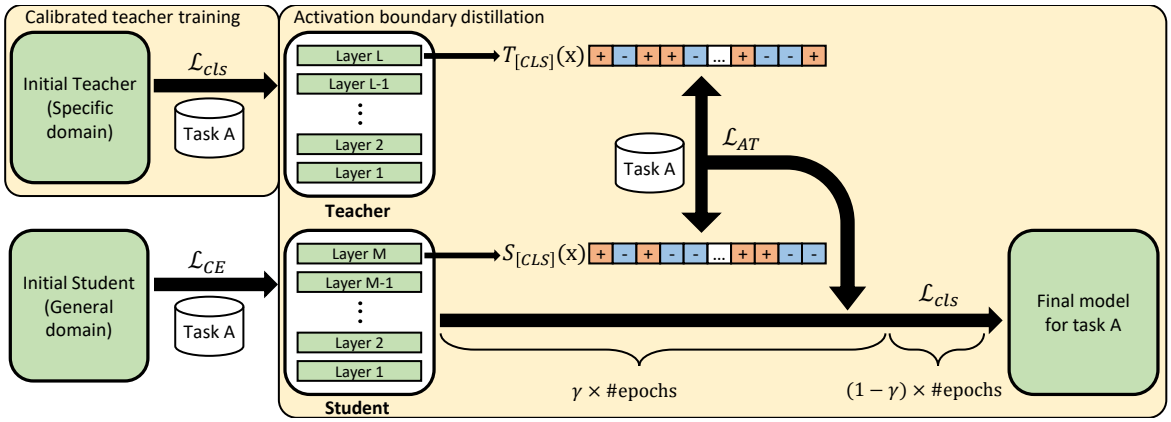


Figure 2: An overview of the DoKTra framework

max function for classes $\mathbf{y}$ and a given input $\mathbf{x}$. The entropy value of the output probability is given by

$$H(p_\theta(\mathbf{y}|\mathbf{x})) = -\sum_i p_\theta(\mathbf{y}_i|\mathbf{x}) \log(p_\theta(\mathbf{y}_i|\mathbf{x})), \tag{1}$$

where i denotes the class index. Finally, negative entropy is added to a regular cross-entropy loss $\mathcal{L}_{CE}$,

$$\mathcal{L}_{cls} = \mathcal{L}_{CE} - \beta H(p_\theta(\mathbf{y}|\mathbf{x})), \tag{2}$$

where β refers to a hyperparameter that controls the strength of entropy penalty.

3.3 Activation Boundary Distillation

Recently, Heo et al. (2019) has proposed a knowledge distillation method that only distils the activation boundary of the hidden representation of a deep neural network. Instead of distilling the magnitude of the neurons of the teacher network, Heo et al. (2019) designed the distillation loss to only transfer the activation of neurons and thus, allowed the activation boundary to be transferred. Since the decision boundary of a model, which consists of a combination of activation boundaries, is critical for the classification task, this method outperformed several distillation methods in image classification. Moreover, they also reported that the activation boundary distillation can learn rapidly and more efficiently with a small amount of training data. Thus, we select it as the domain-knowledge transferring method for our framework; this is beacuse the domain-specific downstream tasks usually consist of lesser training data than general domains.

To apply the activation boundary distillation to PLMs, we use classification embedding of the teacher and student as the distillation target. More precisely, the input sequence of a PLM such as BERT can be written as $[CLS], t_1, t_2, \ldots, [SEP]$, where t_i is the i-th token of the example. Then, the final output sequence is $h([CLS]), h(t_1), \ldots, h([SEP])$, where $h(t)$ indicates the hidden output of the last layer of the token t. For the classification task, the output embedding of the first special token("[CLS]," also known as the classification token) is generally used as the input of the classification layer. Thus, we apply activation boundary distillation to the classification embedding (output embedding of the classification token). For an input example $\mathbf{x}$, let $T_{[CLS]}(\mathbf{x}) \in \mathbb{R}^d$ and $S_{[CLS]}(\mathbf{x}) \in \mathbb{R}^d$ be the classification embedding vector ($h([CLS])$) of the teacher and student model, respectively. An element-wise activation indicator function can be defined to express the activation of a neuron:

$$\rho(x) = \begin{cases} 1, & \text{if } x > 0 \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

The loss function to transfer the activation of neurons is a l_1 norm of the difference between activations:

$$\mathcal{L}_{AT}(\mathbf{x}) = \|\rho(T_{[CLS]}(\mathbf{x})) - \rho(S_{[CLS]}(\mathbf{x}))\|_1. \tag{4}$$

However, this loss function cannot be minimized using gradient descent because ρ is a discrete function. To address this issue, Heo et al. (2019) has proposed an alternative loss function similar to hinge loss (Rosasco et al., 2004) with an activation function σ.

$$\mathcal{L}_{AT}(\mathbf{x}) = \|\rho(T_{[CLS]}(\mathbf{x})) \odot \sigma(\mu\mathbf{1} - (S_{[CLS]}(\mathbf{x}))) + (\mathbf{1} - \rho(T_{[CLS]}(\mathbf{x}))) \odot \sigma(\mu\mathbf{1} + (S_{[CLS]}(\mathbf{x})))\|_2^2, \quad (5)$$

where $\odot$ is the element-wise product and $\mathbf{1}$ is a d-dimensional vector, with all values equal to 1. μ is the margin, which is a hyperparameter for training stability.

Specifically, we select two PLMs as the initial student model: ALBERT-xlarge (Lan et al., 2019), which has a smaller number of parameters but performs better than BERT, and RoBERTa-large (Liu et al., 2019), which has a larger number of parameters and is known to outperform BERT significantly for most of the tasks. To distil the knowledge from a teacher model, we first fine-tune the student model to provide initial knowledge about the task. Then the student model is trained with $\mathcal{L}_{AT}$. We also add a few refinement steps to refine the classification layer of the student model. Because the student model is already fine-tuned before the distillation step, this additional refinement may cause overconfidence. Thus, we apply a confidence penalty regularization in the refinement step. Namely, the student is refined with $\mathcal{L}_{cls}$ after the distillation steps. We add a hyperparameter $\gamma \in [0, 1]$, which determines when the training loss is switched from distillation to refinement. The procedure of the DoKTra framework is summarized in Algorithm 1.

Algorithm 1 DoKTra framework

Input: Downstream task data $D = \{x_k, y_k\}_{k=1}^{N}$, hyperparameter β_1, β_2, γ

```
Fine-tune the teacher T with data D, using L_cls with β1
Fine-tune the student S with data D, using L_CE
epoch_switch = epochs_total × γ
for each epoch do
    if epoch < epoch_switch then
        Train S using L_AT
    else
        Train S using L_cls with β2
    end if
end for
return Student model S
```

Dataset	#Train	#Dev	#Test	Metrics	Domain
ChemProt	17865	11263	15583	micro F1	Biomed.
GAD	4796	-	534	F1	Biomed.
DDI	18779	7244	5761	micro F1	Biomed.
i2b2	22160	96	43000	micro F1	Clin.
HoC	10527	1496	2896	F1	Biomed.

Table 1: The statistics of the downstream task datasets

4 Experiments

4.1 Datasets

We evaluated our approach on several biomedical and clinical classification downstream tasks, including relation extraction and multi-label classification.

The relation extraction task aims to classify the relationship between two entities (e.g., gene, chemical, and disease) that are already annotated. The ChemProt (Krallinger et al., 2017) dataset contains PubMed abstracts with 10 types of chemical-protein interaction annotations and only five of the types are used for evaluation. The GAD dataset (Bravo et al., 2015) consists of gene-disease binary relation annotations. The DDI (Herrero-Zazo et al., 2013) dataset consists of text from the DrugBank database and Medline abstracts, with four types of drug-drug interaction annotations. In the clinical domain, the i2b2 dataset (Uzuner et al., 2011) contains texts from clinical documents, and eight types of relations between medical problems and treatments have been annotated. The HoC (Baker et al., 2016) corpus consists of PubMed abstracts with ten types of hallmarks of cancer annotation. Note that the HoC dataset is a multi-label document classification task predicting the combination of labels from an input text.

We pre-process every classification dataset except for GAD in the same manner as the BLUE (Peng et al., 2019) benchmark. In particular, entity anonymization is applied to all relation extraction datasets, which replace the entity mentions with anonymous tokens (e.g., @GENE$, @DISEASE$) to avoid confusion in using complex entity names. We use a pre-processed version of the GAD dataset provided by BioBERT, which is split for 10-fold cross-validation. The statistics of the pre-processed downstream task datasets are listed in Table 1.

Models	#Params.	ChemProt	GAD	DDI	i2b2	HoC	Avg.	Retain
BioBERT-ft (teacher)	110M	76.20±0.65	81.59±0.27	80.05±0.62	74.14±0.35	84.21±0.33	79.24	
ALBERT-ft (student)	60M	73.67±0.98	74.33±0.91	81.31±0.72	69.89±1.17	81.76±0.20	76.19	
ALBERT-DoKTra	60M	77.42±0.04	78.86±0.19	82.30±0.41	72.98±0.07	83.52±0.44	79.02	99.72%
RoBERTa-ft (student)	355M	75.75±0.35	77.84±1.80	80.71±1.56	72.51±1.80	83.98±0.44	78.16	
RoBERTa-DoKTra	355M	78.04±0.22	81.38±0.05	82.25±0.30	75.65±0.11	85.34±0.12	80.53	101.63%

Table 2: The DoKTra framework's main experimental results. (ft: fine-tuned)

4.2 Experimental details

For the experiments, we used the pre-trained BioBERT-base model (L=12, H=768, A=12) as the initial teacher model. We used two pre-trained models as the initial student model: ALBERT-xlarge (L=24, H=2048, A=32) and RoBERTa-large (L=24, H=1024, A=16). In the previous description, we have assumed that the embedding dimensions of teachers and students are identical. However, because the hidden embedding dimensions of teachers and students are different in our setting, we applied a linear transformation to the teacher's classification embedding to match the dimension with the student model.

In calibrated teacher training, we trained for 3-10 epochs with a learning rate of 2e-5. The hyperparameter β_1, the strength of the confidence penalty in teacher training, was chosen from {0, 0.3, 0.5, 0.7}. For activation boundary distillation, we first fine-tuned the initial student model for 5-10 epochs with learning rates of {6e-6, 8e-6, 1e-5}. Then, we distilled for 10 epochs with learning rates of {6e-6, 8e-6, 1e-5}. The confidence penalty strength β_2 in the refinement step and loss switch rate γ were chosen from {0, 0.3, 0.5, 0.7} and {0.6, 0.7 ,0.8, 0.9}, respectively. The margin μ of the activation transfer loss was set to 1.0. Every hyperparameter was tuned on the development set. The selected hyperparameters are shown in the Appendix.

The experiments were run on a single RTX 3090 24 GB GPU, and the training codes were implemented in PyTorch. All experiments were repeated three times with different random seeds, and the average performances and standard deviations have been reported.

4.3 Experimental results on downstream tasks

Table 2 shows the overall experimental F1 score results of the DoKTra framework on five biomedical and clinical classification tasks. The initially fine-tuned student models are in the second and fourth rows and the DoKTra framework is applied to both, as shown in the third and fifth rows.

As shown in the third and fifth rows, the classification performances of biomedical and clinical downstream tasks are significantly improved by applying our proposed framework, when compared to the initial student models. This implies that distilling the activation patterns of the neurons from the calibrated teacher model can transfer its domain-specific knowledge and thus improve the task performance in the domain on which the student has not yet been pre-trained.

By applying the DoKTra framework, the ALBERT-xlarge student model was able to retain 99.72% of the teacher model performance on an average. ALBERT has two advantages: a small number of parameters and high performance (Lan et al., 2019). Applying our framework to ALBERT allowed us to obtain a student model with performance comparable to that of the teacher with half the parameters. In other words, we successfully transferred domain-specific knowledge to ALBERT while maintaining its existing advantages. Consequently, the distilled ALBERT achieved a higher performance than the teacher model on ChemProt and DDI.

The RoBERTa model that was applied to the proposed framework outperformed the teacher model on an average, specifically in four of five downstream tasks (ChemProt, DDI, i2b2, and HoC). RoBERTa's performance was already similar to the teacher model in the initial fine-tuning stage because it was pre-trained with more data than BERT and exhibited a greater robustness. The results on RoBERTa imply that our proposed framework can be effectively applied to emerging and advanced pre-trained language models. In other words, domain-specific knowledge can be transferred into advanced models without a time-consuming pre-

Dataset	BioBERT -ft	RoBERTa -PM-ft	RoBERTa -DoKTra
ChemProt	76.20	**79.00**	78.04
GAD	**81.59**	81.16	81.38
DDI	80.05	81.39	**82.25**
i2b2	74.14	**78.83**	75.65
HoC	84.21	**86.11**	85.34
Avg.	79.24	**80.90**	80.53

Table 3: Performance comparison between existing pre-trained model and DoKTra. (**bold** for the best, underline for the second best)

Dataset	RoBERTa -ft	TAPT	TAPT (3xGPU)	RoBERTa -DoKTra
ChemProt	75.75	73.55	75.40	78.04
GAD	80.17	81.85	81.41	84.47
DDI	80.71	73.61	78.00	82.25
i2b2	72.51	70.95	72.42	75.65
HoC	83.98	86.39	86.45	85.34
Avg.	79.34	77.27	78.74	81.15

Table 4: Performance comparison between TAPT and DoKTra.

training and perturbing the model's efficacy in the general domain.

4.4 Performance comparisons

To compare our approach with the in-domain pre-training method, we used RoBERTa-PM-large (Lewis et al., 2020), which is a RoBERTa-large model additionally pre-trained with a large biomedical and clinical corpus consisting of 14 billion words. We fine-tuned the RoBERTa-PM for each task.

Table 3 shows the classification performance of BioBERT, RoBERTa-PM, and our approach in five biomedical and clinical tasks. As mentioned before, our best model outperformed the BioBERT (teacher) model on four of the five tasks. Notably, our approach even outperformed RoBERTa-PM on two tasks and demonstrated comparable performances on the others. These results are remarkable since our approach spent only a few hours on each task, whereas RoBERTa-PM may require several days and billions of words to be pre-trained. Note that RoBERTa-PM has an advantage in the i2b2 task since its pre-training data contains MIMIC-III clinical text data, while our teacher model was pre-trained with only biomedical texts. In other words, this implies our approach has a room for further improvement when a better in-domain model is set as a teacher.

We also compared our framework with task-adaptive pre-training (TAPT) (Gururangan et al., 2020), an additional pre-training method for PLMs. The TAPT approach additionally pre-trains an existing PLM before fine-tuning it with the training samples of each task. As both TAPT and DoKTra only utilize the task-specific training data, they can be fairly compared in terms of performance and training resources. For TAPT, we additionally pre-trained the RoBERTa-large model with each pre-processed downstream task's training data. We followed the hyperparameters used in TAPT except for batch size and the maximum sequence length because we used the same computing resource as DoKTra for a fair comparison. The possible maximum pre-training batch size with the given computing resource for the RoBERTa-large model was 36. Since the results of the RoBERTa-large model with a small batch size were unstable, we also performed a distributed training with three GPUs, resulting in a batch size of 108.

The comparison results are shown in Table 4. Note that the performance on GAD in Table 4 was evaluated with the first split of a 10-fold cross-validation, while the main result in Table 3 was evaluated with all splits. As revealed in the results, even though TAPT showed improved results in the original study with Google Cloud TPU, it was unstable with the small batch size and sequence length; the performances were even degraded in the general GPU environment. Although the TAPT performance improved when the batch size increased through distributed training, the improvement was inadequate. This may be because of the batch size being smaller than that in the TPU environment. Moreover, DoKTra required less training time than TAPT while both methods were task-specific. For instance, TAPT required a total of seven hours of training, while DoKTRa was completed in only 1.1 hours for the ChemProt task. This is because DoKTra leverages the knowledge of an existing in-domain PLM, thus requiring only a few fine-tuning and distillation steps. The comparison of TAPT and DoKTra using more advanced computing resources is left as a future work.

Dataset	DoKTra - CTT		DoKTra	
	F1(%)	$\mathcal{L}_{AT}$	F1(%)	$\mathcal{L}_{AT}$
ChemProt	76.20±0.20	193.75	77.42±0.04	139.79
GAD	77.26±0.94	331.50	78.86±0.19	268.95
DDI	82.16±0.63	131.62	82.30±0.41	98.97
i2b2	72.82±0.30	123.29	72.98±0.07	92.20

Table 5: Comparison of average classification performance and loss values with or without teacher calibration. (CTT: calibrated teacher training)

Models	F1 (%)	Improvement
BioBERT-ft (teacher)	76.20±0.65	
ALBERT-ft (student)	73.67±0.98	
+KLD	76.40±0.36	+2.73
+CTT+KLD	76.87±0.49	+3.20
+ABD	76.20±0.24	+2.53
+CTT+ABD (proposed method)	**77.42±0.04**	**+3.75**
ALBERT-ft+CPL	74.04±0.43	+0.37

Table 6: Ablation study on the ChemProt dataset. (ft: fine-tuned, KLD: KL-divergence-based distillation, CTT: calibrated teacher training, ABD: activation boundary distillation, ft+CPL: fine-tuned with confidence penalty loss)

4.5 Efficacy of combining calibration and activation boundary distillation

We conducted an experiment to verify the positive effect of combining calibrated teacher training and activation boundary distillation. Because the entropy regularizer in calibrated teacher training issues penalties based on the output probability distribution, it is difficult to intuitively understand how it positively affects activation boundary distillation, which uses hidden representation. Thus, we ablate the calibrated teacher training steps in our framework and compare the final performances and loss values.

Irrespective of the use of an alternative version (Equation 5) during the training, the extent to which the activation pattern is distilled can be intuitively observed by calculating the original "activation transfer loss" (Equation 4). The value of Equation 4 directly refers to the number of neurons activated differently than the teacher model. For instance, if $\mathcal{L}_{AT} = 500$ for an ALBERT model (H=2,048), it indicates that 500 of the 2,048 elements in the hidden representation vector exhibited signs different to those of the teacher.

Table 5 shows the experimental results on four relation extraction tasks with ALBERT students. As shown in Table 5, the application of the calibrated teacher training reduces the $\mathcal{L}_{AT}$ and improves the classification performance. In other words, calibration on the teacher training clearly aids the supervision of the teacher in activation boundary distillation, even though the output probability information is not directly used in distillation.

4.6 Ablation study

To observe how each component contributed to the proposed framework, we conducted an ablation study. We ablated two major components: calibrated teacher training (CTT) and activation boundary distillation (ABD). The experiments were performed on the ChemProt dataset, using the ALBERT-xlarge model as the student architecture. To ablate the calibrated teacher training, we trained the teacher model using only $\mathcal{L}_{CE}$. We compared the activation boundary distillation with KL-divergence based distillation (KLD), which penalizes the difference between the output probability distributions of the two models.

Table 6 presents the results of the ablation study. As we proposed, applying both calibrated teacher training and activation boundary distillation resulted in a superior performance. In particular, the calibrated teacher model was able to distil its activation boundary to the student model much more effectively, thus improving the performance of the student model, as we hypothesized in the previous section. Applying KL-divergence-based distillation yielded positive results in terms of classification performance. Notably, calibrated teacher training also improved the KL-divergence-based distillation because it enabled the distillation of a considerably more reliable output probability, as reported in Menon et al. (2021). Note that applying the confidence regularizer to the fine-tuning of the student model only slightly improved the performance, suggesting that the observed gains in our model are only partially because of the calibration regularizer.

4.7 Experimental results on financial domain

To verify the general applicability of our approach, we conducted experiments on financial sentiment classification tasks. Financial sentiment analysis

Models	#Params	FPB	FTS	Avg.	Retain
FinBERT-ft (teacher)	110M	85.70±0.59	85.88±0.48	85.79	
ALBERT-ft (student)	60M	83.85±1.65	80.79±1.94	82.32	
ALBERT-DoKTra	60M	86.25±0.19	86.08±1.82	86.17	100.44%
RoBERTa-ft (student)	125M	85.78±0.29	81.76±0.48	83.77	
RoBERTa-DoKTra	125M	87.21±0.29	85.10±0.19	86.16	100.43%

Table 7: Experimental results of DoKTra framework on financial domain. (ft: fine-tuned)

aims to classify the polarity of financial-related text, such as financial news or tweets. Since financial text usually contains specialized language, several pre-training approaches have emerged (Araci, 2019; Yang et al., 2020; Liu et al., 2021) to fill the gap between the general and financial domains.

In this study, we selected the FinBERT (Yang et al., 2020) model as a teacher in the DoKTra framework and evaluated our approach on two tasks, the Financial PhraseBank (FPB) and FinTextSen (FTS). The Financial PhraseBank (FPB) (Malo et al., 2014) contains sentences from financial news annotated for positive, neutral, and negative sentiments. The FinTextSen (FTS) (Cortis et al., 2017) consists of financial tweets from Twitter and StockTwits with real-valued sentiment scores. To transform it into a classification task, we clustered the sentiment score into a 3-class label, following Daudert et al. (2018). The Financial PhraseBank dataset contains 4,846 sentences, and we set 10% of the examples as the test set while preserving the label distribution. The FinTextSen originally includes 2,488 tweets, but only 1,700 tweets are available now. We set 10% of the entire data as the test set, which is similar to FPB.

As shown in Table 7, ALBERT-DoKTRa and RoBERTa-DoKTRa outperformed the FinBERT-ft teacher on financial downstream tasks. Note that we used the RoBERTa-base model in this section because of the training stability. This result suggests that DoKTra can be applied regardless of the domain and can be an efficient alternative to in-domain pre-training.

5 Conclusion

In this study, we proposed the DoKTra framework as a domain knowledge transfer method for PLMs. The experimental results from the biomedical, clinical, and financial domain downstream tasks demonstrated that our proposed framework could transfer domain-specific knowledge into a PLM, while preserving its own expressive advantages without any further pre-training with additional in-domain data. We employed advanced models as the student model and verified the future applicability of our framework to emerging language models by achieving even higher performances than the teacher model. However, the limitations of our approach are that it is task-specific and was evaluated only in classification tasks. Our future studies would focus on developing the proposed framework as a task-agnostic method and evaluating it on various tasks.

Acknowledgements

This research was supported by the Bio-Synergy Research Project (NRF-2016M3A9C4939665) of the Ministry of Science and ICT through the National Research Foundation of Korea (NRF) and the NRF grant funded by the Korean government (Ministry of Science and ICT) (NRF-2018M3C7A1054932), and partly supported by Institute for Information and communications Technology Promotion (IITP) grant funded by the Korea government (MSIP) [No. 2019-0-01842, Artificial Intelligence Graduate School Program (GIST)].

References

Dogu Araci. 2019. Finbert: Financial sentiment analysis with pre-trained language models. *arXiv preprint arXiv:1908.10063*.

Simon Baker, Ilona Silins, Yufan Guo, Imran Ali, Johan Högberg, Ulla Stenius, and Anna Korhonen. 2016. Automatic semantic classification of scientific literature according to the hallmarks of cancer. *Bioinformatics*, 32(3):432–440.

Àlex Bravo, Janet Piñero, Núria Queralt-Rosinach, Michael Rautschka, and Laura I Furlong. 2015. Extraction of relations between genes and diseases from text and large-scale data analysis: implications for translational research. *BMC bioinformatics*, 16(1):1–17.

Jang Hyun Cho and Bharath Hariharan. 2019. On the efficacy of knowledge distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4794–4802.

Dongha Choi and Hyunju Lee. 2020. Extracting chemical–protein interactions via calibrated deep neural network and self-training. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 2086–2095.

Keith Cortis, André Freitas, Tobias Daudert, Manuela Huerlimann, Manel Zarrouk, Siegfried Handschuh, and Brian Davis. 2017. Semeval-2017 task 5: Fine-grained sentiment analysis on financial microblogs and news. Association for Computational Linguistics (ACL).

Tobias Daudert, Paul Buitelaar, and Sapna Negi. 2018. Leveraging news sentiment to improve microblog sentiment classification in the financial domain. In *Proceedings of the First Workshop on Economics and Natural Language Processing*, pages 49–54.

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.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2020. Domain-specific language model pretraining for biomedical natural language processing. *arXiv preprint arXiv:2007.15779*.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. 2017. On calibration of modern neural networks. In *International Conference on Machine Learning*, pages 1321–1330. PMLR.

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

Byeongho Heo, Minsik Lee, Sangdoo Yun, and Jin Young Choi. 2019. Knowledge transfer via distillation of activation boundaries formed by hidden neurons. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3779–3787.

María Herrero-Zazo, Isabel Segura-Bedmar, Paloma Martínez, and Thierry Declerck. 2013. The ddi corpus: An annotated corpus with pharmacological substances and drug–drug interactions. *Journal of biomedical informatics*, 46(5):914–920.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Jangho Kim, SeongUk Park, and Nojun Kwak. 2018. Paraphrasing complex network: network compression via factor transfer. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 2765–2774.

Martin Krallinger, Obdulia Rabal, Saber A Akhondi, Martın Pérez Pérez, Jesús Santamaría, Gael Pérez Rodríguez, Georgios Tsatsaronis, and Ander Intxaurrondo. 2017. Overview of the biocreative vi chemical-protein interaction track. In *Proceedings of the sixth BioCreative challenge evaluation workshop*, volume 1, pages 141–146.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Patrick Lewis, Myle Ott, Jingfei Du, and Veselin Stoyanov. 2020. Pretrained language models for biomedical and clinical tasks: Understanding and extending the state-of-the-art. In *Proceedings of the 3rd Clinical Natural Language Processing Workshop*, pages 146–157.

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*.

Zhuang Liu, Degen Huang, Kaiyu Huang, Zhuang Li, and Jun Zhao. 2021. Finbert: A pre-trained financial language representation model for financial text mining. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 4513–4519.

Pekka Malo, Ankur Sinha, Pekka Korhonen, Jyrki Wallenius, and Pyry Takala. 2014. Good debt or bad debt: Detecting semantic orientations in economic texts. *Journal of the Association for Information Science and Technology*, 65(4):782–796.

Aditya K Menon, Ankit Singh Rawat, Sashank Reddi, Seungyeon Kim, and Sanjiv Kumar. 2021. A statistical perspective on distillation. In *International Conference on Machine Learning*, pages 7632–7642. PMLR.

Jeremy Nixon, Michael W Dusenberry, Linchuan Zhang, Ghassen Jerfel, and Dustin Tran. 2019. Measuring calibration in deep learning. In *CVPR Workshops*, volume 2.

Yifan Peng, Shankai Yan, and Zhiyong Lu. 2019. Transfer learning in biomedical natural language processing: An evaluation of bert and elmo on ten benchmarking datasets. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 58–65.

Gabriel Pereyra, George Tucker, Jan Chorowski, Łukasz Kaiser, and Geoffrey Hinton. 2017. Regularizing neural networks by penalizing confident output distributions. *arXiv preprint arXiv:1701.06548*.

Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. 2014. Fitnets: Hints for thin deep nets. *arXiv preprint arXiv:1412.6550*.

Lorenzo Rosasco, Ernesto De Vito, Andrea Caponnetto, Michele Piana, and Alessandro Verri. 2004. Are loss functions all the same? *Neural computation*, 16(5):1063–1076.

Siqi Sun, Yu Cheng, Zhe Gan, and Jingjing Liu. 2019. Patient knowledge distillation for bert model compression. 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 4323–4332.

Özlem Uzuner, Brett R South, Shuying Shen, and Scott L DuVall. 2011. 2010 i2b2/va challenge on concepts, assertions, and relations in clinical text. *Journal of the American Medical Informatics Association*, 18(5):552–556.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Yi Yang, Mark Christopher Siy Uy, and Allen Huang. 2020. Finbert: A pretrained language model for financial communications. *arXiv preprint arXiv:2006.08097*.

Junho Yim, Donggyu Joo, Jihoon Bae, and Junmo Kim. 2017. A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4133–4141.

A Appendix

A.1 Hyperparameter setting

In this section, we report the searching scheme and actual values of the hyperparameters used by us. In all cases, we set the batch size to the maximum that a single GPU can process, with 128 being the maximum sequence length.

In calibrated teacher training, we first select the number of epochs and the learning rate as the default values of the BioBERT code and slightly change the number of epochs (e) for the unreported tasks from BioBERT. Then, we select the strength of the confidence regularization (β_1) by a grid search in terms of the F1 score and expected calibration error (ECE) on the development set. The formula for calculating ECE is as follows:

$$\text{acc}(B_m) = \frac{1}{|B_m|} \sum_{i \in B_m} \mathbf{1}(\hat{y}_i = y_i),$$

$$\text{conf}(B_m) = \frac{1}{|B_m|} \sum_{i \in B_m} \hat{p}_i,$$

$$\text{ECE} = \sum_{m=1}^{M} \frac{|B_m|}{n} |\text{acc}(B_m) - \text{conf}(B_m)|,$$

where B_m is the m-th bin, $\hat{y_i}$ and y_i indicate the predicted and true labels of the i-th sample in the bin, and $\hat{pi}$ is the output prediction probability. n is the number of total examples. A low ECE value implies that the model generates an output probability similar to its accuracy, and thus, is well-calibrated. The actual values of the hyperparameters for the calibrated teacher training are summarized in Table A1.

In activation boundary distillation, we perform a grid search to determine the number of epochs (e_1) and learning rate (lr_1) for initial student fine-tuning. Then, we conduct another grid search of the learning rate (lr_2), number of epochs (e_2), weight of the confidence penalty (β_2), and loss switch rate (γ) for the distillation and refinement steps. Both searches are performed on the development set. The actual values of the hyperparameters for the ALBERT student are summarized in Table A2. For the RoBERTa model as a student, we use the same teacher with ALBERT. The hyperparameters of the activation boundary distillation for the RoBERTa student are searched in the same manner with the ALBERT and summarized in Table A3.

A.2 Experimental details for financial domain

In this section, we report on the details of two financial downstream task datasets, the experimental details, and hyperparameters of the financial task experiments.

we used the pre-trained FinBERT-base model (L=12, H=768, A=12) with the original vocabulary. We used ALBERT-xlarge (L=24, H=2048, A=32) and RoBERTa-base (L=12, H=768, A=12) as the students. The hyperparameters are searched in the

same way as the experiments for the biomedical domain. The actual values of the hyperparameters for the calibrated teacher training and activation boundary distillation with ALBERT and RoBERTa are summarized in Tables A4, A5, and A6.

Dataset	CTT	
	e	β_1
ChemProt	5	0.3
GAD	3	0.7
DDI	5	0.3
i2b2	5	0.3
HoC	10	0

Table A1: The hyperparameters for calibrated teacher training

Dataset	ABD					
	e_1	lr_1	e_2	lr_2	β_2	γ
ChemProt	10	6e-6	10	1e-5	0.5	0.9
GAD	5	6e-6	10	1e-5	0.3	0.9
DDI	10	8e-6	10	1e-5	0.7	0.9
i2b2	10	1e-5	10	1e-5	0.5	0.9
HoC	10	1e-5	10	6e-6	0	0.6

Table A2: The hyperparameters for activation boundary distillation of the ALBERT model

Dataset	ABD					
	e_1	lr_1	e_2	lr_2	β_2	γ
ChemProt	5	1e-5	10	1e-5	0.5	0.8
GAD	5	1e-5	10	1e-5	0.5	0.9
DDI	10	1e-5	10	1e-5	0.7	0.8
i2b2	5	1e-5	10	1e-5	0.5	0.8
HoC	10	1e-5	10	6e-6	0	0.6

Table A3: The hyperparameters for activation boundary distillation of the RoBERTa model.

Dataset	CTT	
	e	β_1
FPB	5	0.7
FTS	5	0.3

Table A4: The hyperparameters for calibrated teacher training for the financial domain.

Dataset	ABD					
	e_1	lr_1	e_2	lr_2	β_2	γ
FPB	10	6e-6	10	1e-5	0.0	0.9
FTS	10	6e-6	10	6e-6	0.1	0.8

Table A5: The hyperparameters for activation boundary distillation of the ALBERT model for the financial domain.

Dataset	ABD					
	e_1	lr_1	e_2	lr_2	β_2	γ
FPB	5	1e-5	10	1e-5	0.0	0.9
FTS	10	1e-5	10	6e-6	0.5	0.9

Table A6: The hyperparameters for activation boundary distillation of the RoBERTa model for the financial domain.