



OPEN

## Subset selection for domain adaptive pre-training of language model

JunHa Hwang, SeungDong Lee, HaNeul Kim & Young-Seob Jeong✉

Pre-trained language models have brought significant performance improvements in many natural language understanding tasks. Domain-adaptive language models, which are trained with a specific domain corpus, exhibit high performance in their target domains. However, pre-training these models with a large amount of domain-specific data requires a substantial computational budget and resources, necessitating the development of efficient pre-training methods. In this paper, we propose a novel subset selection method called AlignSet, which extracts an informative subset from a given domain dataset for efficient pre-training. Our goal is to extract an informative subset that enables faster learning of the language model compared to learning from the entire dataset. By experiments across multiple domains, we demonstrate that AlignSet generates better subsets than other methods.

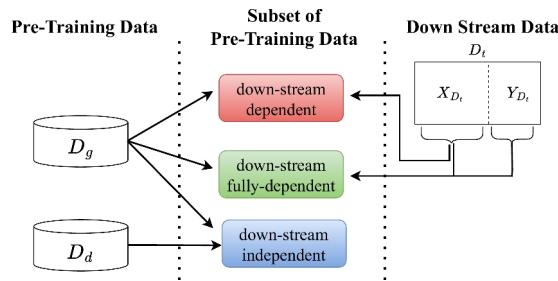
With the advent of Transformer<sup>1</sup>, pre-trained language models (PLMs) showed remarkable performance across a range of natural language understanding (NLU) tasks<sup>2–5</sup>. Especially, domain-adaptive LMs obtained through continual pre-training (CPT) were found to be effective for target domains<sup>6–8</sup>. Despite its effectiveness, the high cost of pre-training limits its applicability. For example, pre-training ROBERTa-large takes approximately one day using 1,024 V100 GPUs<sup>3</sup>. Therefore, it is necessary to develop an efficient way to develop domain-adaptive LMs.

There are mainly two approaches for efficiently pre-training domain-adaptive LMs with a limited computational budget: a model-centric and a data-centric. The model-centric approach involves designing a compact model such as ALBERT<sup>9</sup> and EarlyBERT<sup>10</sup>, whereas the data-centric approach focuses on data management or selection. As one of the data-centric approaches, subset selection refers to extracting only a portion of a large dataset that includes representative or informative instances, thereby creating a subset that is much smaller in size than the original dataset. Recently, there have been studies that efficiently train language models using subsets while maintaining or improving performance<sup>11–13</sup>. However, these studies have limitations, such as relying on downstream task datasets or not outperforming random sampling.

The subset selection methods can be categorized into three groups based on the data dependency: *down-stream dependent*, *down-stream fully-dependent*, and *down-stream independent group*, as depicted in Fig. 1. The three groups commonly utilize a large general pre-training corpus  $D_g$  that is used to pre-train general-purposed foundation models (e.g., Bidirectional encoder representations from transformers (BERT)<sup>2</sup>). The upper two groups in the figure exploit the down-stream dataset  $D_t = \{X_{D_t}, Y_{D_t}\}$  where  $X_{D_t}$  and  $Y_{D_t}$  indicate the input texts and the output labels, respectively. Specifically, the *down-stream dependent* group takes  $D_g$  and  $X_{D_t}$ , and the *down-stream fully-dependent* group employs  $D_g$ ,  $X_{D_t}$  and  $Y_{D_t}$ . The dependency to the down-stream dataset  $D_t$  may cause bias when we do not have a sufficient amount of the down-stream data. On the other hand, the *down-stream independent* group does not suffer from this issue as it takes only  $D_g$  and  $D_d$ , where  $D_d$  is a dataset of a target domain. Note that the  $D_d$  is unlabelled corpus in the target domain, whereas the general dataset  $D_g$  incorporates multiple domains.

In this study, we propose a new subset selection method called AlignSet, belonging to the *down-stream independent group*, to efficiently pre-train a domain-adapted LM within a limited computational budget while avoiding the aforementioned limitations. The main idea of AlignSet is to align sentence representations obtained from two different PLMs so that we can exploit the aligned representations to score examples. Based on the finding that there exist representative subsets for a target domain<sup>13</sup>, our goal is to extract an informative subset that makes the language model better work on domain-specific tasks. Note that our model is designed to extract a subset in such a way that the language model achieves better performance on target domain tasks compared to a randomly selected subset. However, it is not designed to outperform a language model trained on the entire dataset. Our contributions are as follows.

Department of Computer Engineering, Chungbuk National University, Cheongju 28644, Republic of Korea. ✉email: ysjay@chungbuk.ac.kr



**Fig. 1.** Three groups of subset selection methods according to the dependency to the down-stream dataset, where  $D_g$  represents the pre-training corpus of the general domain,  $D_d$  indicates the pre-training corpus of a target domain, and  $X_{D_t}$  and  $Y_{D_t}$  represent text and label in down-stream task dataset  $D_t$ , respectively.

- AlignSet: We propose a new subset selection method of *down-stream independent* group. It utilizes the similarity between aligned embeddings obtained from two PLMs.
- Experiments on multiple domains: Similar to Gururangan et al.<sup>14</sup>, we perform the experiments on several domains (e.g., biomedical science, computer science, news, and personality), and show that the proposed method outperforms other subset selection methods.

## Related works

### Pre-training language models

After Transformer<sup>1</sup> sheds light on natural language processing (NLP) field, language models (LMs) using Transformer architecture pre-trained on large datasets from the general domain have shown outstanding performance across various NLP tasks.<sup>2,4,15,16</sup> BERT<sup>2</sup>, one of the most well-known LMs, employs the encoder part of the Transformer, effectively capturing latent representations and making it applicable to various NLU tasks. Such encoder-based LMs are known as powerful to NLU tasks as it is designed to encode or comprehend the context of given texts. Indeed, compared to decoder-based LMs (e.g., Generative Pre-trained Transformer (GPT) series<sup>4,15,17,18</sup>, Llama series<sup>19–21</sup>), the encoder-based LMs mostly achieve the better performance on NLU tasks if the LMs are of the same scale (i.e., same model size). Furthermore, recent studies found that adapting LMs to a target domain through fine-tuning or continual pre-training (CPT) contributes to performance gain. In this paper, we focus on the encoder-based LMs and propose a new subset selection method for efficient CPT.

### Efficient pre-training

While the pre-training<sup>22,23</sup> or continual pre-training(CPT)<sup>14</sup> is a promising way to build a general or domain-adaptive language model, it demands significant computational resources.<sup>24</sup> To mitigate this issue, there have been studies on model-centric and data-centric approaches to efficiently pre-train language models with a limited computational budget. As a model-centric approach, Chen et al.<sup>10</sup> designed a way of training models based on the lottery ticket hypothesis. Lan et al.<sup>9</sup> and Dehghani et al.<sup>25</sup> introduced parameter sharing techniques for making the models to have fewer parameters. Such parameter sharing enables better performance in language modeling and allows to efficiently construct the language models. In this paper, we chose the ALBERT<sup>9</sup> utilizing such efficient parameter sharing technique as our base model for experiments.

As a data-centric approach, there have been subset selection techniques that extract informative examples from a large dataset to construct a small but effective or representative subset. The subset allows efficient training without losing much performance compared to the total dataset. Yao et al.<sup>11</sup> employed BM25<sup>26</sup> by treating the textual data of down-stream task as a query. This allows the similarity measurement between the pre-training data and the query to facilitate subset selection. Wang et al.<sup>12</sup> introduced a subset selection based on influence function<sup>27,28</sup>, and showed that their method outperforms other previous domain-specific language models. It finds samples that have the most positive impact on the down-stream task using the influence function. These studies commonly rely on the dataset of target task  $D_t$ ; they are of *down-stream dependent* or *down-stream fully-dependent* groups. Suzuki et al.<sup>13</sup>, belonging to *down-stream independent* group, defined scores based on the difference in length-normalized per-word cross-entropy between a domain-adaptive language model  $M_d$  and a general language model  $M_g$ . They proved that there exist representative subsets for a given domain, but their method did not exhibit much performance improvement over randomly extracted subsets. We believe that the biggest reason for their marginal performance gain might be that they strongly rely on the model loss, so it became difficult to distinguish between noisy samples and informative samples<sup>29,30</sup>.

## Method

Suppose we want to construct a domain-adaptive language model for a target domain  $d$ . The best way is to borrow a general-purposed language model (i.e., a language model pre-trained with general domains), and adapt it to the domain  $d$  through fine-tuning or continual pre-training (CPT). However, even if we have a pre-training corpus  $D_d$  of the domain  $d$ , it is expensive to perform CPT with the entire corpus  $D_d$ . Our subset selection method extracts a subset  $\hat{D}_d$  that comprises a certain portion of  $D_d$ , where the subset  $\hat{D}_d$  is much smaller than  $D_d$  but contains informative samples. In this paper, we arbitrarily set  $|\hat{D}_d| \approx (0.1 \sim 0.27) \times |D_d|$

based on our limited computation resources. Please refer to the subsection of Experimental settings and datasets for more details.

### AlignSet

Following Suzuki et al. <sup>13</sup>, we assume that we have two pre-trained language models (PLMs): a general-purposed language model  $M_g$  and an initial in-domain language model  $M_d$  of the target domain, as shown in Fig. 2. Any model can be chosen as the initial model  $M_d$  when it is pre-trained with any small amount of the target domain; in this paper, as random selection is the simplest way to get a subset, a language model pre-trained with a randomly selected subset of the target domain is employed as  $M_d$ . The goal of this study is to design a subset selection method for efficiently pre-training language models, so it might seem problematic that the initial model is already a PLM. It is worth noting that the initial model is pre-trained with an arbitrary in-domain subset (e.g., random subset), and we use such initial model as a proxy for the target domain to extract an *informative* subset that provides performance gain. Through experiments, we demonstrate that AlignSet is effective and efficient, even when considering the time required to build the initial model.

The two PLMs  $M_d$  and  $M_g$  take the same set of texts as input and yield sentence embeddings  $V_d \in \mathbb{R}^{N \times m_d}$  and  $V_g \in \mathbb{R}^{N \times m_g}$ , where  $N$  is the number of sentences in a batch. The embeddings can be any vectors derived from the PLMs (e.g., output vectors of [CLS] token, mean vectors of all tokens, etc). The two PLMs have different embedding spaces, so  $V_d$  and  $V_g$  must be different even for the same text. Inspired by Radford et al. <sup>41</sup> and Karpukhin et al. <sup>42</sup>, we design *align-layer* that projects  $V_d$  and  $V_g$  into an unified embedding space, where the projected embeddings  $U_d \in \mathbb{R}^{N \times m}$  and  $U_g \in \mathbb{R}^{N \times m}$ . The unified embedding space allows to get the similarity between sentence embeddings, where the diagonal values are self-similarities, as depicted in Fig. 2.

The motivation of AlignSet is as follows. First, we assume that if a particular example differs from or is more distinguishable than the others, then the example is more informative. To find such informative examples, we introduce a contrastive loss that measures the similarity of an example to itself and to other examples. This approach enables us to train a unified embedding space that captures the similarity between the two PLMs. After training with contrastive loss, a high similarity between positive pairs indicates that they are clearly distinguishable from other examples. In other words, examples with high similarity across the two PLMs are considered more distinctive or informative compared to others. Second, if we simply apply the contrastive loss to a target domain using a single model (i.e.,  $M_g = M_d$ ), it may lead to selecting only abnormal examples or outliers, as is often the case in the computer vision area <sup>43,44</sup>. Therefore, we design the aligned space by introducing  $M_g$  and  $M_d$ , where  $M_g$  has a base or general perspective (i.e., distribution) and  $M_d$  has a perspective of the target domain. We expect that the aligned space between the two PLMs will allow to find domain-informative examples that are not merely abnormal or outliers.

### Loss function

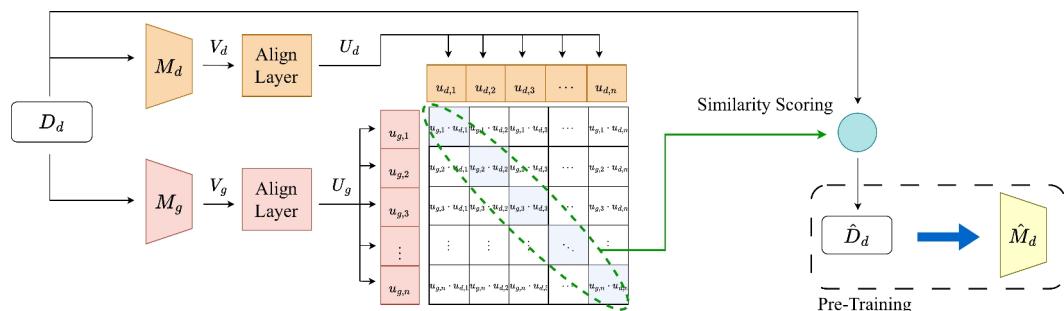
Following Radford et al. <sup>41</sup>, we utilize the InfoNCE loss function <sup>45</sup> to train the align-layer parameters:

$$L_{NCE}(v_i, W) = -\log \frac{e^{sim(v_i, w_i)/\tau}}{\sum_{w_j \in W} e^{sim(v_i, w_j)/\tau}} \quad (1)$$

where  $\tau$  is a temperature,  $w_i$  and  $v_i$  indicate  $i$ -th element of embeddings generated by the two language models within the batch, and  $W = \{w_1, \dots, w_{|W|}\}$ . The temperature parameter controls the range of the logits in the softmax function and is a trainable parameter that is directly optimized during training. In this paper, we employ cosine similarity as the similarity function  $sim$  between embeddings. We apply this loss both column-wise and row-wise, and obtain the total loss  $L$  as below.

$$L = (L_{NCE}(u_{g,i}, U_d) + L_{NCE}(u_{d,i}, U_g))/2 \quad (2)$$

The loss encourages maximizing the similarity of *positive pairs* while minimizing the similarity of *negative pairs*. In other words, positive pairs are pulled closer together, while negative pairs are pushed farther apart in aligned



**Fig. 2.** The overview of AlignSet, where  $\hat{D}_d$  is a subset of  $D_d$ ,  $\hat{M}_d$  is a domain-adaptive language model,  $V_d$  and  $V_g$  are embeddings generated by  $M_d$  and  $M_g$ , respectively. And  $U_d$  and  $U_g$  refer to the embeddings projected into the unified embedding space via the alignment layer.

space. The positive pairs, represented by the diagonals in Fig. 2, correspond to the embeddings of the same examples, whereas the negative pairs correspond to the embeddings of different examples. During the training phase,  $M_g$  and  $M_d$  are frozen, and the parameters of the align-layer are updated. After training, the positive pairs are used to compute the self-similarity scores. Specifically, self-similarity scores for positive pairs can be obtained from the trained model's unified embedding space as an inference result. We utilize this score to measure the informativeness of each example, and the top- $k$  examples are selected as the subset  $\hat{D}_d$ .

## Experiments

### Experimental settings and datasets

For our limited computational resource, we chose to use A Lite BERT (ALBERT)<sup>9</sup> for all models (e.g.,  $M_g$ ,  $M_d$ , and  $\hat{M}_d$ ). Our method only learns the aligned space between encoder-based models without modifying architecture, making it easily applicable to other encoder-based models. We utilized a publicly available general-purposed ALBERT checkpoint from HuggingFace as  $M_g$ . As explained earlier, any model can be chosen as  $M_d$  if it is pre-trained with a small subset of the target domain. In our experiments,  $M_d$  is obtained by performing continual pre-training (CPT) on  $M_g$  with a randomly selected subset of  $D_d$ . For AlignSet, we set the [CLS] token representations of the last layer as  $V_g$  and  $V_d$ , and employ a linear layer as the align-layer, which is randomly initialized. When we fine-tune the model and predict task labels, the representation of the [CLS] token is passed to the task-specific linear layer.

We conducted experiments on multiple domains: computer science (CS), biomedical science (BioMed), news, personality, and review, where the review domain is for cross-domain experiments. Table 1 summarizes the source and specification of the in-domain pre-training datasets  $D_d$  and down-stream task datasets  $D_t$ , where all datasets are in English. The subset ratio (i.e., 10~27%) is chosen based on the amount of time required to pre-train the models because it mostly takes too long for pre-training. For example, in the personality domain, it took 358 hours (15 days) for a single run of pre-training with the 10% subset. On the other hand, the dataset size of the news domain is much smaller than the other domains, so we arbitrarily set the subset size to be 10% of the personality domain subset, which corresponds to 27%. The task of  $D_t$  is a classification task, except for the First Impressions V2 dataset which is a regression task on OCEAN factors: openness, conscientiousness, extraversion, agreeableness, and neuroticism. Specifically, the OCEAN factor regression is to predict per-factor scores ranging from 0 to 1 for a given input text. Therefore, we took mean absolute error (MAE) as a metric for the OCEAN factor regression, while we report the classification results using macro-F1 and micro-F1 by following Gururangan et al.<sup>14</sup> and Beltagy et al.<sup>6</sup>. All experimental results are averages of five independent runs with random seeds.

In the pre-training steps, we used Adam optimizer<sup>46</sup> with decoupled weight decay regularization<sup>47</sup>, and the initial learning rate was 0.0001. We used the batch-size of 16, and the number of epochs was 20. We adopted a linear learning rate scheduler with 10,000 warm-up steps. In the fine-tuning steps, we used the batch-size of 32, and an initial learning rate of 0.00003 with 10 epochs. The experiments were conducted on 4 NVIDIA GeForce RTX 3090 GPUs.

## Results

### Main result

We compared the AlignSet with two rival methods: a randomly selected subset (RandomSet) and a subset obtained from the method of Suzuki et al.<sup>13</sup> (SuzukiSet). Table 2 summarizes the experimental results on multiple domains, where FullSet indicates the entire pre-training dataset  $D_d$ . We examined the FullSet only on the news domain due to our limited computational budget. Note that the domains of  $\hat{D}_d$  and the task dataset are the same for BioMed, CS, news, and personality domains, whereas cross-domain adaptation was examined on the review domain. The vanilla model indicates the general purposed model  $M_g$  without any continual pre-training. The subset selection methods generally outperformed the vanilla model, which is consistent with previous studies showing that domain-adaptive models are superior to general models for target domains. The

Purpose	Domain	Dataset	Task	Subset ratio	Size		
					Train	Val	Test
pre-training( $D_d$ )	CS, BIOMED	S2ORC <sup>31</sup>	–	10%	10,610,430	–	–
	News	CCNEWS <sup>32</sup>	–	27%	708,241	–	–
	Personality	Pandora <sup>33</sup>	–	10%	17,640,062	–	–
down-stream( $D_t$ )	CS	ACL-ARC <sup>34</sup>	Citation intent classification	–	1688	114	139
	BIOMED	RCT <sup>35</sup>	Abstract sentence roles classification	–	180,040	30,212	30,135
	News	AGNEWS <sup>36</sup>	Topic classification	–	115,000	5000	7600
		HYPERPARTISAN <sup>37</sup>	Partisanship classification	–	514	63	65
	Personality	First impressions V2 <sup>38</sup>	OCEAN factor regression	–	6000	2000	2000
	Review	HELPFULNESS <sup>39</sup>	Helpfulness classification	–	115,251	5000	25,000
		IMDB <sup>40</sup>	Sentiment classification	–	20,000	5000	25,000

**Table 1.** Data specification of the domain-specific datasets  $D_d$  and down-stream datasets  $D_t$  across multiple domains.

Domain adaptation	Task dataset	Vanilla model (w/o further pre-training)	RandomSet	SuzukiSet	AlignSet	FullSet
BioMed → BioMed	RCT	85.71 ± 0.06	86.47 ± 0.08	86.39 ± 0.08	<b>86.52 ± 0.04</b>	-
CS → CS	ACL-ARC	74.07 ± 5.60	75.71 ± 2.01	75.56 ± 2.48	<b>76.15 ± 1.51</b>	-
News → News	AGNEWS	93.73 ± 0.19	93.90 ± 0.20	93.99 ± 0.06	<b>94.12 ± 0.13</b>	93.84 ± 0.13
	HYPERTPARTISAN	81.81 ± 4.67	84.84 ± 4.35	84.95 ± 1.26	<b>86.45 ± 1.98</b>	82.63 ± 2.10
Personality → Personality	First Impressions V2	0.1128 ± 0.0013	0.1120 ± 0.0004	0.1121 ± 0.0004	<b>0.1118 ± 0.0001</b>	-
† Personality → Review	HELPFULNESS	68.40 ± 0.65	68.21 ± 0.39	68.19 ± 0.28	<b>68.42 ± 0.46</b>	-
	IMDB	<b>93.64 ± 0.01</b>	93.32 ± 0.20	90.95 ± 0.24	93.35 ± 0.04	-

**Table 2.** Averaged results with standard deviations on four domains, where  $d_1 \rightarrow d_2$  indicates the domain adaptation from  $d_1$  (domain of  $D_d$ ) to  $d_2$  (domain of  $D_t$ ). We used macro-F1 for the classification tasks, but micro-F1 for the RCT dataset. Mean absolute error (MAE) was utilized for the First Impressions V2 dataset. † indicates cross-domain adaptation. Significant values are in bold.

Domain	RandomSet	SuzukiSet	AlignSet
CS, BioMed	4 s	31 h	74 h
News	1 s	2 h	6.5 h
Personality	4 s	74 h	103 h

**Table 3.** Elapsed time of subset selection methods.

AlignSet achieved the best performance amongst them, implying that the AlignSet finds better subsets using the aligned embedding space.

### Impact of in-domain pre-training dataset

Although the subset selection method is mainly for efficient pre-training, we examined the task performance of the FullSet on the news domain. We found that the RandomSet was comparable to the FullSet, but the other subset selection methods exhibited better performance than the FullSet. This result is associated with the size of  $D_d$ . As shown in Table 1,  $D_d$  of the news domain is only 4~6% of other domains, causing a greater distribution shift during the subset selection. Specifically, we speculate that RandomSet performed better than FullSet because it coincidentally happened to be closer to the distribution of the fine-tuning dataset. The performance gap caused by the distribution shift will be relieved as the size of  $D_d$  gets larger<sup>48,49</sup>. We also conducted *cross-domain adaptation* (i.e., *personality* → *review*) to check the importance of the in-domain pre-training dataset  $D_d$  and how well our method works for unseen domains. As shown at the bottom of Table 2, the subset selection methods were comparable to or worse than the vanilla model because the domains of  $D_d$  and  $D_t$  are different. This implies that selecting an appropriate  $D_d$  is crucial for performance.

### Efficiency

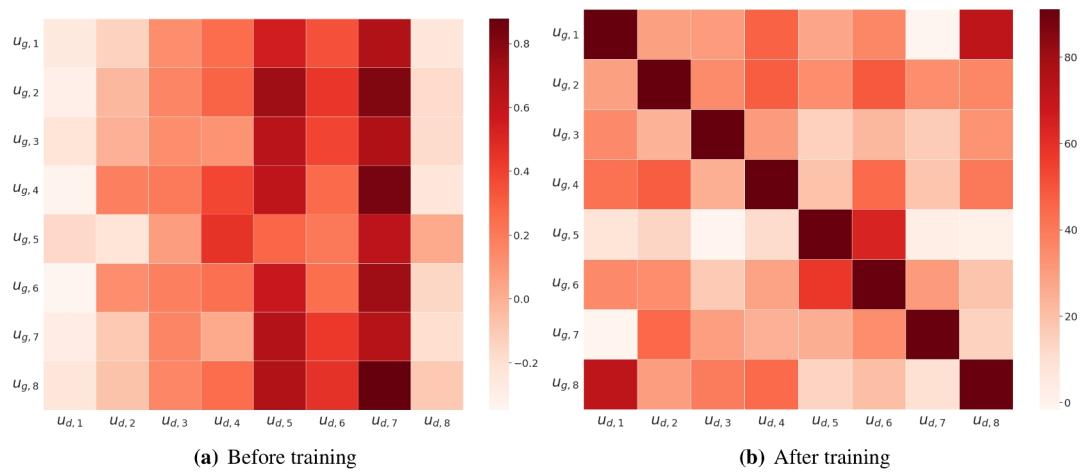
As the essential goal of the subset selection is efficient pre-training, we examined how long it takes to construct the subsets. Table 3 summarizes the elapsed time of different subset selection methods. The elapsed time for AlignSet includes align-layer training (e.g. 88 h) and data selection (e.g. 15 h). The RandomSet was the fastest because it just picks arbitrary examples. Even though the AlignSet takes longer than the SuzukiSet, the AlignSet is still more efficient than the FullSet. Assuming that we build a domain-adaptive LM for the personality domain. If we perform continual pre-training (CPT) for 20 epochs with FullSet, then it takes longer than 150 days using 4 NVIDIA GeForce RTX 3090 GPUs. On the other hand, the AlignSet will take only 34 days, which is almost one-fifth of the FullSet. That is, it takes 15 days to prepare the in-domain initial model  $M_d$  (i.e., pre-training  $M_d$  using the RandomSet), 103 hours (4.3 days) for subset selection, and another 15 days for pre-training the domain-adaptive LM  $\hat{M}_d$ . We also investigate the memory consumption efficiency of each subset selection method using Big-O notation. For RandomSet, since samples are randomly selected from the entire dataset, it requires  $O(1)$ . On the other hand, SuzukiSet and AlignSet operate in small batches of size  $B$ , requiring  $O(B^2)$ . This memory consumption does not account for the complexity of model training and applies only to additional memory complexity.

### Aligned space

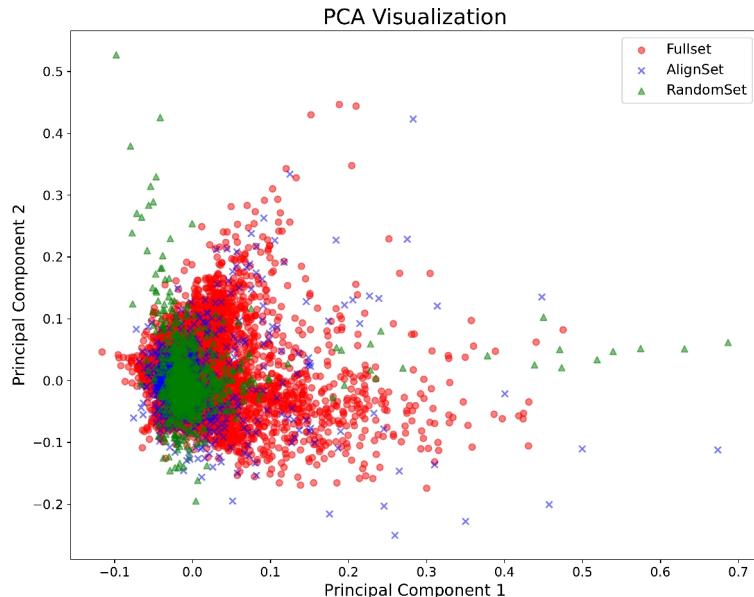
To check if the AlignSet forms the aligned embedding space well, we visualized the embedding space using arbitrary eight examples, as shown in Fig. 3 where the columns and rows of the heatmaps denote  $U_d$  and  $U_g$  within Fig. 2, respectively. In Fig. 3b, aligned diagonal values imply that the align-layer is well-trained and properly maps the sentence representations to the aligned space.

### Subset visualization

To assess the quality of the subsets, we visualized AlignSet, RandomSet, and FullSet using PCA (principal component analysis) plots and presented in Fig. 4. For visualization, we randomly selected 10,000 samples from



**Fig. 3.** Heatmaps of aligned embedding space, where the columns and rows correspond to  $U_d$  and  $U_g$ , respectively.



**Fig. 4.** PCA visualization of FullSet, RandomSet, and AlignSet in the CS domain, where circles represent FullSet, and triangles and X markers indicate RandomSet and AlignSet, respectively.

the FullSet and 1000 samples from each remaining subset in the CS domain. In the figure, FullSet is represented by red circles, RandomSet by green triangles, and AlignSet by blue X markers. AlignSet appears to better reflect the distribution of FullSet compared to RandomSet. Specifically, while RandomSet captures FullSet in a more localized manner, AlignSet preserves relatively more essential characteristics of FullSet.

### Limitation

This study has a limitation that we utilized the ALBERT checkpoint for all the experiments due to our limited computational budget; at the early stage of this study, we found that the estimated time for pre-training RoBERTa<sup>3</sup> or BigBird<sup>50</sup> with RandomSet of the personality domain takes longer than 50 days.

Another limitation of this paper is that we did not conduct experiments with decoder-based models (e.g., GPT<sup>4,51</sup> or Llama<sup>52,53</sup> series). Previous articles discovered the potential of domain-adaptive generative models<sup>54,55</sup>, and we believe that the AlignSet will be able to contribute to the efficient development of domain-adaptive decoder-based models. The AlignSet is just a subset selection method, so it will be applicable to any other domains and multimodal tasks. However, the low-resource languages (LRL) have relatively small datasets, so dataset augmentation techniques will be more appropriate instead of subset selection methods.

Furthermore, we chose the subset size based on the time required to pre-train the models. Based on the previous analysis<sup>1</sup>, the time complexity of training transformer is  $O(N \cdot L^2 \cdot d)$ , where  $L$  represents sequence

length,  $d$  and  $N$  indicate dimension and data size, respectively. Since the actual training time is influenced by the hardware environment, we manually checked the training time on our machine and determined the subset ratio accordingly.

## Conclusion

To reduce pre-training costs for a target domain, we proposed a novel subset selection method AlignSet. The AlignSet is designed to align the sentence representations of two different PLMs by a unified embedding space, and examples with high self-similarity scores are selected as a subset. By experimental results on multiple domains, we showed that the AlignSet allows to efficiently pre-train domain-adaptive LMs. As on-device language models (i.e., small language models) are having attention lately due to the heavy computational cost of training language models, we believe that our subset selection method will contribute to developing small but effective language models for a target domain. We also plan to extend the AlignSet by applying it iteratively; when we get the domain-adaptive LM  $M_d$ , then it can be used as the initial model  $M_d$  for another round of AlignSet. We expect that through this process, the quality of AlignSet will improve. However, this will also lead to an increase in training time. In other words, such an iterative way may degrade in terms of efficiency, but we expect to find a suitable balance between efficiency and effectiveness.

## Data availability

All datasets used in the current study, except for Pandora and First Impression V2, are publicly available: S2ORC (<https://huggingface.co/datasets/allenai/peS2o>), CCNEWS ([https://huggingface.co/datasets/vblagoje/cc\\_news](https://huggingface.co/datasets/vblagoje/cc_news)), others (<https://github.com/allenai/dont-stop-pretraining>). The Pandora and First Impressions V2 datasets are available upon reasonable request and with permission from each author: Pandora (<https://psy.takelab.fer.hr/datasets/all/pandora/>), First Impressions V2 (<https://chalearnlap.cvc.uab.cat/dataset/24/description/>).

Received: 26 August 2024; Accepted: 11 March 2025

Published online: 19 March 2025

## References

1. Vaswani, A. et al. Attention is all you need. In (Guyon, I. et al. eds.) *Advances in Neural Information Processing Systems*. Vol. 30 (Curran Associates, Inc., 2017).
2. Devlin, J., Chang, M.-W., Lee, K. & Toutanova, K. BERT: Pre-training of deep bidirectional transformers for language understanding. In (Burstein, J., Doran, C. & Solorio, T. eds.) *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Vol. 1 (Long and Short Papers). 4171–4186. <https://doi.org/10.18653/v1/N19-1423> (Association for Computational Linguistics, 2019).
3. Liu, Y. et al. Roberta A robustly optimized bert pretraining approach. [arXiv:1907.11692](https://arxiv.org/abs/1907.11692) (2019).
4. Brown, T. et al. Language models are few-shot learners. In (Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M. & Lin, H. eds.) *Advances in Neural Information Processing Systems*. Vol. 33. 1877–1901 (Curran Associates, Inc., 2020).
5. Raffel, C. et al. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.* **21**, 1–67 (2020).
6. Beltagy, I., Lo, K. & Cohen, A. SciBERT: A pretrained language model for scientific text. In (Inui, K., Jiang, J., Ng, V. & Wan, X. eds.) *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. 3615–3620. <https://doi.org/10.18653/v1/D19-1371> (Association for Computational Linguistics, 2019).
7. Lee, J. et al. BioBERT: A pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics* **36**, 1234–1240. <https://doi.org/10.1093/bioinformatics/btz682>. [https://academic.oup.com/bioinformatics/article-pdf/36/4/1234/48983216/bioinformatics\\_36\\_4\\_1234.pdf](https://academic.oup.com/bioinformatics/article-pdf/36/4/1234/48983216/bioinformatics_36_4_1234.pdf) (2019).
8. Gu, Y. et al. Domain-specific language model pretraining for biomedical natural language processing. *ACM Trans. Comput. Healthcare* **3**. <https://doi.org/10.1145/3458754> (2021).
9. Lan, Z. et al. ALBERT: A lite BERT for self-supervised learning of language representations. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26–30, 2020* (OpenReview.net, 2020).
10. Chen, X. et al. EarlyBERT: Efficient BERT training via early-bird lottery tickets. In (Zong, C., Xia, F., Li, W. & Navigli, R. eds.) *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)*. 2195–2207. <https://doi.org/10.18653/v1/2021.acl-long.171> (Association for Computational Linguistics, Online, 2021).
11. Yao, X., Zheng, Y., Yang, X. & Yang, Z. NLP from scratch without large-scale pretraining: A simple and efficient framework. In (Chaudhuri, K. et al. eds.) *Proceedings of the 39th International Conference on Machine Learning*. Vol. 162. *Proceedings of Machine Learning Research*. 25438–25451 (PMLR, 2022).
12. Wang, X. et al. Farewell to aimless large-scale pretraining: Influential subset selection for language model. In (Rogers, A., Boyd-Graber, J. & Okazaki, N. eds.) *Findings of the Association for Computational Linguistics: ACL 2023*. 555–568. <https://doi.org/10.18653/v1/2023.findings-acl.35> (Association for Computational Linguistics, 2023).
13. Suzuki, J., Zen, H. & Kazawa, H. Extracting representative subset from extensive text data for training pre-trained language models. *Inf. Process. Manag.* **60**, 103249. <https://doi.org/10.1016/j.ipm.2022.103249> (2023).
14. Gururangan, S. et al. Don't stop pretraining: Adapt language models to domains and tasks. In (Jurafsky, D., Chai, J., Schluter, N. & Tetreault, J. eds.) *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 8342–8360. <https://doi.org/10.18653/v1/2020.acl-main.740> (Association for Computational Linguistics, Online, 2020).
15. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I. et al. Improving language understanding by generative pre-training (2018).
16. Raffel, C. et al. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.* **21**, 1–67 (2020).
17. Radford, A. et al. Language models are unsupervised multitask learners (2019).
18. Ouyang, L. et al. Training language models to follow instructions with human feedback. In (Koyejo, S. et al. eds.) *Advances in Neural Information Processing Systems*. Vol. 35. 27730–27744 (Curran Associates, Inc., 2022).
19. Touvron, H. et al. Llama: Open and efficient foundation language models. [arXiv:2302.13971](https://arxiv.org/abs/2302.13971) (2023).
20. Touvron, H. et al. Llama 2: Open foundation and fine-tuned chat models. [arXiv:2307.09288](https://arxiv.org/abs/2307.09288) (2023).
21. Dubey, A. et al. The llama 3 herd of models. [arXiv:2407.21783](https://arxiv.org/abs/2407.21783) (2024).
22. Dai, A.M. & Le, Q. V. Semi-supervised sequence learning. In (Cortes, C., Lawrence, N., Lee, D., Sugiyama, M. & Garnett, R. eds.) *Advances in Neural Information Processing Systems*. Vol. 28 (Curran Associates, Inc., 2015).

23. Howard, J. & Ruder, S. Universal language model fine-tuning for text classification. In (Gurevych, I. & Miyao, Y. eds.) *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*. (Volume 1: Long Papers). 328–339. <https://doi.org/10.18653/v1/P18-1031> (Association for Computational Linguistics, 2018).
24. Kaplan, J. et al. Scaling laws for neural language models (2020). [arXiv:2001.08361](https://arxiv.org/abs/2001.08361).
25. Dehghani, M., Gouws, S., Vinyals, O., Uszkoreit, J. & Kaiser, L. Universal transformers. In *International Conference on Learning Representations* (2019).
26. Robertson, S. & Zaragoza, H. The probabilistic relevance framework: Bm25 and beyond. *Found. Trends Inf. Retr.* **3**, 333–389. <https://doi.org/10.1561/1500000009> (2009).
27. Cook, R.D. & Weisberg, S. Residuals and influence in regression (1982).
28. Koh, P.W. & Liang, P. Understanding black-box predictions via influence functions. In (Precup, D. & Teh, Y. W. eds.) *Proceedings of the 34th International Conference on Machine Learning*. Vol. 70. *Proceedings of Machine Learning Research*. 1885–1894 (PMLR, 2017).
29. Paul, M., Ganguli, S. & Dziugaite, G. K. Deep learning on a data diet: Finding important examples early in training. *Adv. Neural Inf. Process. Syst.* **34**, 20596–20607 (2021).
30. Fayyaz, M. et al. Bert on a data diet: Finding important examples by gradient-based pruning. [arXiv:2211.05610](https://arxiv.org/abs/2211.05610) (2022).
31. Lo, K., Wang, L.L., Neumann, M., Kinney, R. & Weld, D. S. S2orc: The semantic scholar open research corpus. [arXiv:1911.02782](https://arxiv.org/abs/1911.02782) (2020).
32. Hamborg, F., Meuschke, N., Breitinger, C. & Gipp, B. News-please: A generic news crawler and extractor. In *Proceedings of the 15th International Symposium of Information Science*. 218–223. <https://doi.org/10.5281/zenodo.4120316> (2017).
33. Gjurković, M., Karan, M., Vučkojević, I., Bošnjak, M. & Snajder, J. PANDORA talks: Personality and demographics on Reddit. In (Ku, L.-W. & Li, C.-T. eds.) *Proceedings of the Ninth International Workshop on Natural Language Processing for Social Media*. 138–152. <https://doi.org/10.18653/v1/2021.socialnlp-1.12> (Association for Computational Linguistics, Online, 2021).
34. Jurgens, D., Kumar, S., Hoover, R., McFarland, D. & Jurafsky, D. Measuring the evolution of a scientific field through citation frames. *Trans. Assoc. Comput. Linguist.* **6**, 391–406. [https://doi.org/10.1162/tacl\\_a\\_00028](https://doi.org/10.1162/tacl_a_00028) (2018).
35. Dernoncourt, F. & Lee, J. Y. PubMed 200k RCT: A dataset for sequential sentence classification in medical abstracts. In (Kondrak, G. & Watanabe, T. eds.) *Proceedings of the Eighth International Joint Conference on Natural Language Processing*. (Volume 2: Short Papers). 308–313 (Asian Federation of Natural Language Processing, 2017).
36. Zhang, X., Zhao, J. & LeCun, Y. Character-level convolutional networks for text classification. In (Cortes, C., Lawrence, N., Lee, D., Sugiyama, M. & Garnett, R. eds.) *Advances in Neural Information Processing Systems*. Vol. 28 (Curran Associates, Inc., 2015).
37. Kiesel, J. et al. SemEval-2019 task 4: Hyperpartisan news detection. In (May, J. et al. eds.) *Proceedings of the 13th International Workshop on Semantic Evaluation*. 829–839. <https://doi.org/10.18653/v1/S19-2145> (Association for Computational Linguistics, 2019).
38. Junior, J. C. S. J., Lapedriza, A., Palmero, C., Baro, X. & Escalera, S. Person perception biases exposed: Revisiting the first impressions dataset. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) Workshops*. 13–21 (2021).
39. McAuley, J., Targett, C., Shi, Q. & van den Hengel, A. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '15. 43–52. <https://doi.org/10.1145/2766462.2767755> (Association for Computing Machinery, 2015).
40. Maas, A.L. et al. Learning word vectors for sentiment analysis. In (Lin, D., Matsumoto, Y. & Mihalcea, R. eds.) *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. 142–150 (Association for Computational Linguistics, 2011).
41. Radford, A. et al. Learning transferable visual models from natural language supervision. In (Meila, M. & Zhang, T. eds.) *Proceedings of the 38th International Conference on Machine Learning*. Vol. 139. *Proceedings of Machine Learning Research*. 8748–8763 (PMLR, 2021).
42. Karpukhin, V. et al. Dense passage retrieval for open-domain question answering. In (Webber, B., Cohn, T., He, Y. & Liu, Y. eds.) *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 6769–6781. <https://doi.org/10.18653/v1/2020.emnlp-main.550> (Association for Computational Linguistics, Online, 2020).
43. Lamboley, C., Bitard-Feildel, T. & Chérubin, A. Anomaly detection in embedding space learned with a contrastive approach. In *Conference on Artificial Intelligence for Defense. Actes de la 4ème Conference on Artificial Intelligence for Defense (CAID 2022)* (DGA Maîtrise de l'Information, 2022).
44. Tack, J., Mo, S., Jeong, J. & Shin, J. CSI: Novelty detection via contrastive learning on distributionally shifted instances. In (Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M. & Lin, H. eds.) *Advances in Neural Information Processing Systems*. Vol. 33. 11839–11852 (Curran Associates, Inc., 2020).
45. van den Oord, A., Li, Y. & Vinyals, O. Representation learning with contrastive predictive coding. [arXiv:1807.03748](https://arxiv.org/abs/1807.03748) (2019).
46. Kingma, D. P. & Ba, J. A Method for Stochastic Optimization. Adam. [arXiv:1412.6980](https://arxiv.org/abs/1412.6980) (2017).
47. Loshchilov, I. & Hutter, F. Decoupled weight decay regularization. In *International Conference on Learning Representations* (2019).
48. Kouw, W. M. & Loog, M. An introduction to domain adaptation and transfer learning. [arXiv:1812.11806](https://arxiv.org/abs/1812.11806) (2019).
49. Isik, B. et al. Scaling laws for downstream task performance of large language models [arXiv:2402.04177](https://arxiv.org/abs/2402.04177) (2024).
50. Zaheer, M. et al. Big bird: Transformers for longer sequences. In (Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M. & Lin, H. eds.) *Advances in Neural Information Processing Systems*. Vol. 33. 17283–17297 (Curran Associates, Inc., 2020).
51. Black, S. et al. GPT-NeoX-20B: An open-source autoregressive language model. In (Fan, A., Ilic, S., Wolf, T. & Gallé, M. eds.) *Proceedings of BigScience Episode #5 – Workshop on Challenges & Perspectives in Creating Large Language Models*. 95–136. <https://doi.org/10.18653/v1/2022.bigsience-1.9> (Association for Computational Linguistics, virtual+Dublin, 2022).
52. Touvron, H. et al. Llama: Open and efficient foundation language models. [arXiv:2302.13971](https://arxiv.org/abs/2302.13971) (2023).
53. Taori, R. et al. Stanford alpaca: An instruction-following llama model. [https://github.com/tatsu-lab/stanford\\_alpaca](https://github.com/tatsu-lab/stanford_alpaca) (2023).
54. Wu, S. et al. Bloomberggpt: A large language model for finance. [arXiv:2303.17564](https://arxiv.org/abs/2303.17564) (2023).
55. Bolton, E. et al. Biomedlm. <https://github.com/stanford-crfm/BioMedLM> (2023).

## Acknowledgements

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (RS-2022-II220043, Adaptive Personality for Intelligent Agents). This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(RS-2020-NR049604).

## Author contributions

J.H.H and Y.S.J. conceptualize and wrote the main manuscript. J.H.H. conducted the experiments. J.H.H., S.D.L., and H.N.K. checked the experimental results. Y.S.J. was in charge of supervision and project administration. All authors reviewed the manuscript.

## Declarations

### Competing interests

All of the authors declare no competing interests.

### Additional information

**Correspondence** and requests for materials should be addressed to Y.-S.J.

**Reprints and permissions information** is available at [www.nature.com/reprints](http://www.nature.com/reprints).

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2025