Composable Cross-prompt Essay Scoring by Merging Models

Sanwoo Lee **Kun Liang** **Yunfang Wu***
School of Computer Science, Peking University
National Key Laboratory for Multimedia Information Processing, Peking University
MOE Key Laboratory of Computational Linguistics, Peking University
{sanwoo, wuyf}@pku.edu.cn kliang25@stu.pku.edu.cn

Abstract

Recent advances in cross-prompt automated essay scoring typically train models jointly on all available source domains, often requiring simultaneous access to unlabeled target domain samples. However, using all sources can lead to suboptimal transfer and high computational cost. Moreover, repeatedly accessing the source essays for continual adaptation raises privacy concerns. We propose a *source-free* adaptation approach that selectively *merges* the parameters of individually trained source models without further access to the source datasets. In particular, we mix the task vectors—the parameter updates from fine-tuning—via a weighted sum to efficiently simulate selective joint-training. We use Bayesian optimization to determine the mixing weights using our proposed **P**rior-encoded **I**nformation **M**aximization (**PIM**), an unsupervised objective which promotes score discriminability by leveraging useful priors pre-computed from the sources. Experimental results with LLMs on in-dataset and cross-dataset adaptation show that our method (1) consistently outperforms joint-training on all sources, (2) maintains superior robustness compared to other merging methods, (3) excels under severe distribution shifts where recent leading cross-prompt methods struggle, all while retaining computational efficiency.[1]

1 Introduction

Automated essay scoring (AES) is a machine learning task of developing a system that scores essays written in response to a given prompt (i.e., writing instructions). A prompt represents a domain as different prompts may have distinct topics. Early works had achieved success in prompt-specific AES (Chen and He, 2013; Taghipour and Ng, 2016; Dong et al., 2017; Farag et al., 2018) where test samples were assumed to belong to the same prompt as training samples. Yet prompt-specific models were found to struggle when tested on new prompts (Phandi et al., 2015), accelerating efforts on cross-prompt AES with domain adaptation or generalization techniques (Zesch et al., 2015; Jin et al., 2018; Chen and Li, 2023; Jiang et al., 2023).

Method	multi-source adaptation	leverages unlabeled target essays	no source essays for adaptation	supports source selection
Phandi et al. (2015)	✗	✗	✗	✗
Cao et al. (2020)	✗	✓	✗	✗
Ridley et al. (2020)	✓	✗	✓	✗
Chen and Li (2023)	✓	✓	✗	✗
Ours	✓	✓	✓	✓

Table 1: Comparison of adaptation settings among holistic cross-prompt AES methods based on key criteria. Our proposed setting satisfies all listed criteria.

Despite the adaptability to the data-scarce target prompt, current cross-prompt methods typically assume simultaneous access to the data from both the source and target prompts when the target samples are leveraged for adaptation (Cao et al., 2020; Chen and Li, 2023). Yet this assumption is often violated due to privacy concerns in releasing the source essays. Instead, models trained from the source prompts are safer to distribute. Hence adapting without source datasets holds great practical implications, which aligns with the unsupervised source-free domain adaptation (SFDA) paradigm (Liang et al., 2020; Wang et al., 2021; Huang et al., 2021).

On the other hand, selecting the most relevant source domains remains a crucial yet underexplored aspect in cross-prompt AES. Most works either adopt single-source adaptation setting (Phandi et al., 2015; Dong and Zhang, 2016; Cozma et al., 2018), or train the model jointly on all source domain datasets for multi-source adaptation (Jin et al., 2018; Ridley et al., 2021; Do et al., 2023; Chen and Li, 2024), the latter likely motivated by the belief

* Corresponding author.
[1]Code is available at https://github.com/sanwooo/composable-cross-prompt.

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 24386–24400
November 4-9, 2025

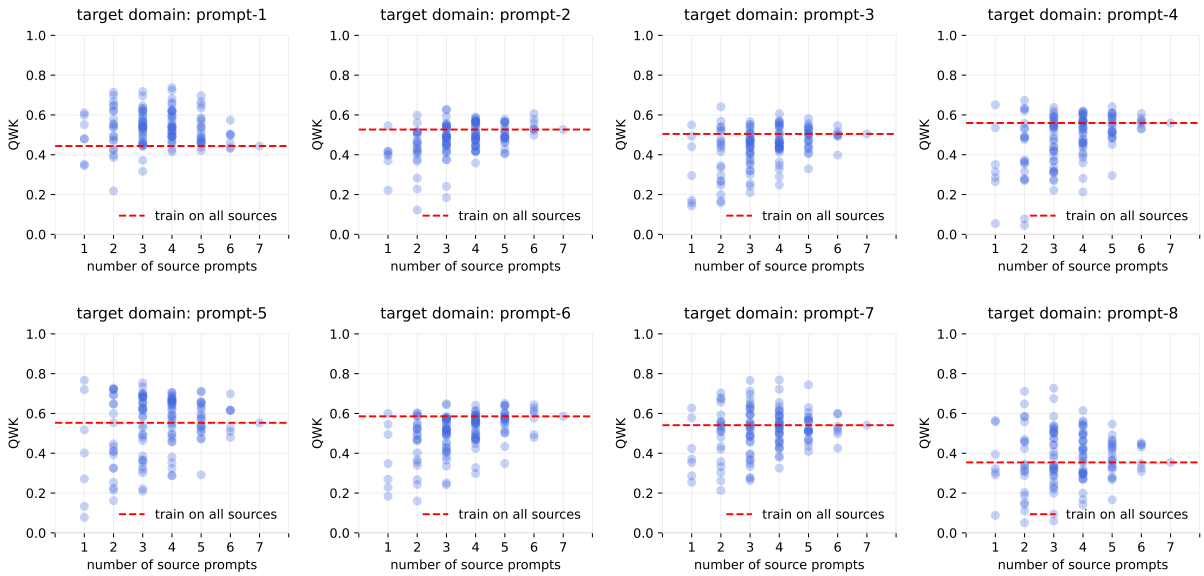

Figure 1: Agreement with human raters (QWK) on the target prompts of ASAP dataset, evaluated using BERT (Devlin et al., 2019) trained jointly on varying number of source prompt datasets. For each target prompt, the remaining prompts serve as source prompts. Training details are provided in Appendix A.

that more sources yield better performance. However, our pilot study in Figure 1 suggests that carefully selecting a subset of source prompts clearly outperforms training on all sources. Nevertheless, the high cost of joint training makes exhaustive search for the optimal subset impractical.

In this work, we explore *merging models* (Wortsman et al., 2022; Matena and Raffel, 2022; Ainsworth et al., 2023) as a scalable alternative to joint training for source-free domain adaptation in cross-prompt AES. That is, we combine models fine-tuned on individual source prompts without re-training. In particular, the weighted sum of the models' task vectors (Ilharco et al., 2023)—parameter updates after fine-tuning—is added back to the pre-trained model (Eq. 2), which effectively mimics joint training in a post-hoc fashion. It then allows for fast and iterative search over the mixing coefficients that (soft-) select the task vectors.

To guide this search in the absence of the target labels and source datasets, we propose **P**rior-encoded **I**nformation **M**aximization (**PIM**), an information-theoretic objective that leverages useful priors pre-computed from the labeled source domains, to enhance scoring performance (Sec. 3.1). The objective is coupled with Bayesian optimization for an efficient optimization of PIM (Sec. 3.2).

We merge lightweight LoRA adapters (Hu et al., 2022) of large language models (LLMs), motivated by LLMs' extensibility to generate rationales (Chu et al., 2025) and the surging efforts to build ever-stronger LLMs. Experiments on in- and cross-dataset settings show that: our method (1) outperforms training jointly on all sources, (2) surpasses other merging methods in a majority of cases, (3) remains robust under severe shifts (i.e., cross-dataset) where recent cross-prompt AES methods struggle, and (4) is time-saving than adaptation methods training jointly on all sources.

In summary, our contributions are as follows:

- We propose a domain-adaptive model merging approach for source-free cross-prompt AES. See Table 1 for the comparison of settings.
- We design an unsupervised objective which promotes model's score discriminability regularized by priors derived from the sources.
- Beyond in-dataset, we validate our method in cross-dataset adaptation, under which our method remains more robust than recent cross-prompt methods.

2 Preliminary

Problem Statement. This paper uses "prompt" and "domain" interchangeably. We consider the unsupervised source-free domain adaptation problem (Liang et al., 2020; Wang et al., 2021; Yang et al., 2022) with multiple source domains for cross-prompt AES, where the input $x \in \mathcal{V}^*$ is a sequence of tokens and the output $y \in \mathbb{Z}$ is an integer score. A pre-trained model $\mathcal{M}(\theta_{pre})$ is fine-tuned

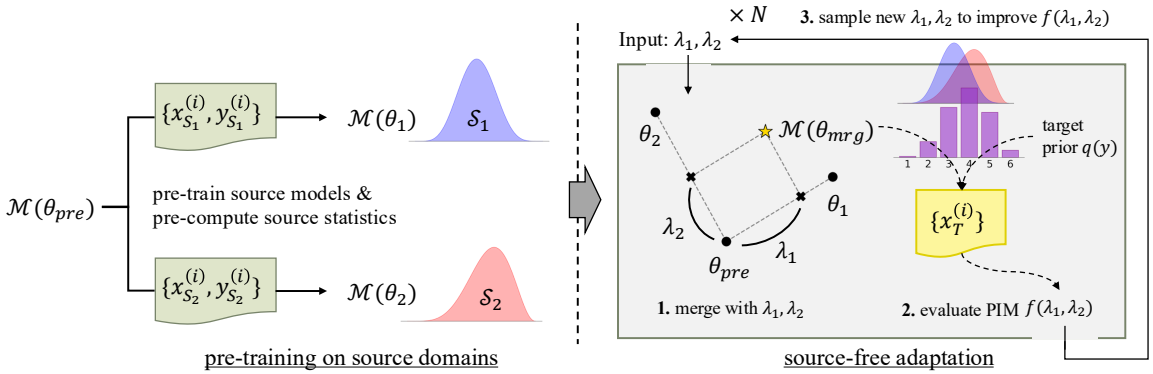

Figure 2: An illustration of our method for source-free cross-prompt AES. **Left**: Source models and statistics are pre-trained before adaptation. **Right**: During source-free adaptation, merging coefficients are optimized via Bayesian optimization to optimize the prior-encoded information maximization (PIM) criterion (Eq. 9).

on each one of the source datasets separately, and additional statistics $\mathcal{S}_j$ may be computed for each source, after which the source datasets become no longer available (Adachi et al., 2025). We denote $\mathcal{M}(\theta_j)$ as the model fine-tuned on j-the source dataset $\mathcal{D}_{S_j} = \{(x_{S_j}^{(i)}, y_{S_j}^{(i)})\}_{i=1}^{N_j}$, parameterized by $\theta_j \in \mathbb{R}^d$. Note that separating each source domain apart is not a requirement for source-free adaptation, but is a stricter setting we aim to address via model merging.

During the adaptation phase, we have access to the fine-tuned models $\{f(\theta_j)\}_{j=1}^{M}$, an unlabeled target dataset $\mathcal{D}_T = \{(x_T^{(i)})\}_{i=1}^{N_T}$ and optionally, the pre-computed statistics $\{\mathcal{S}_j\}_{j=1}^{M}$. We note that the score range may vary across domains (e.g., Table 2). During inference, we require the model to adapt to potentially novel score ranges, different from the common approach that normalizes all score ranges to a shared scale (Taghipour and Ng, 2016; Cozma et al., 2018; Wang and Liu, 2025).

LLM Ordinal Regression. We employ LLMs for scoring essays—an ordinal regression task implemented by autoregressive generation. Following Lukasik et al. (2025), we assume each score y corresponds to a unique string representation $\text{str}(y) \in \mathcal{V}^*$ (e.g., $2 \rightarrow$ "2"). Each input-output pair (x, y) is transformed into a formatted pair (x', y') using an instruction template (e.g., see Appendix C.1). The pre-trained model $\mathcal{M}(\theta_{pre})$ is then instruction-tuned on the source dataset $\mathcal{D}_j$ to maximize the likelihood of generating the answer:

$$\theta_j = \arg\max_{\theta} \mathop{\mathbb{E}}_{x,y} \left[p(y'|x', \theta)\right] \quad (1)$$

Model Merging. The seminal work of Ilharco et al. (2023) introduced the concept of a *task vector* $\tau_j := \theta_j - \theta_{pre}$ defined as the parameter updates obtained through fine-tuning. Interestingly, adding task vectors from multiple tasks to the pre-trained model has been shown to effectively approximate multi-task training with a performance drop, a finding further verified by follow-up studies (Yadav et al., 2023; Yu et al., 2024; Deep et al., 2024). We follow this merging framework and simulate selective joint-training through a weighted sum of the task vectors:

$$\theta_{mrg} = \theta_{pre} + \sum_{j=1}^{M} \lambda_i \tau_j \quad (2)$$

where $\{\lambda_j \in \mathbb{R}\}_{j=1}^{M}$ are the mixing coefficients.

Updating the entire parameters results in large task vectors, making merging less efficient. We instead adopt low-rank adaptation (LoRA) (Hu et al., 2022) and merge lightweight adapters. LoRA fine-tunes models by learning low-rank updates: for a weight matrix $W \in \mathbb{R}^{m \times n}$, the update is $W + \Delta W = W + BA$, where $B \in \mathbb{R}^{m \times r}$ and $A \in \mathbb{R}^{r \times n}$ are learnable low-rank matrices ($r \ll \min(m, n)$). Accordingly, we define task vectors in terms of LoRA adapters:

$$\tau_j = \theta_j - \theta_{pre} = \mathop{||}_{l=1}^{L} \text{flatten}(B_j^{(l)} A_j^{(l)}) \quad (3)$$

where $||$ denotes concatenation of the flattened vectors ($\text{flatten}(B_j^{(l)} A_j^{(l)})$) across layers. Layers without adapters contribute zeros.

Given this setup, merging models via a linear combination of LoRA adapters (Eq. 2, 3) reduces

our objective to selecting the mixing coefficients $\lambda_1, \ldots, \lambda_M$ that lead to an optimal performance on the target prompt.

3 Method

Given the absence of target labels in cross-prompt essay scoring, we establish an objective that promotes the model's scoring performance from an information-theoretic view (Sec. 3.1), and employ Bayesian optimization to maximize this objective without costly backpropagation (Sec. 3.2). Figure 2 illustrates an overview of our method.

3.1 Prior-encoded Information Maximization

In essay scoring where target labels are ordinal scores, a plausible scoring model would assign unambiguous labels for individual essays, while retaining the discriminability across different essays. In standard classification setup, this idea has been formalized as maximizing the mutual information (**MI**) between the input x and the output y (Bridle et al., 1991; Krause et al., 2010; Liang et al., 2020). In what follows, we revisit this principle, study which of its properties can be modified to be better applied in essay scoring, and propose our final objective.

The MI between input x and output y under a discriminative model $p(y|x, \theta)$ is given by:

$$\mathcal{I}(y; x) = \mathcal{H}(p(y|\theta)) - \mathcal{H}(p(y|x, \theta)) \quad (4)$$

where $\mathcal{H}(\cdot)$ denotes entropy. In classification, $\mathcal{I}(y; x)$ is empirically estimated as

$$\mathcal{I}(y; x) = \mathcal{H}(\frac{1}{N}\sum_{i=1}^{N} p(y|x^{(i)}, \theta)) - \frac{1}{N}\sum_{i=1}^{N} \mathcal{H}(p(y|x^{(i)}, \theta)) \quad (5)$$

in which $p(y|x^{(i)}, \theta) \in \mathbb{R}^C$ denotes the output probability of a sample $x^{(i)}$. Essentially, maximizing $\mathcal{I}(y; x)$ balances between sample-wise sharpness and global discriminability in predictions.

However, directly applying this criterion to ordinal regression can be problematic. Note that maximizing $\mathcal{H}(p(y|\theta)) = \log C - KL(p(y|\theta)||U)$ is equivalent to minimizing KL-divergence between $p(y|\theta)$ and a uniform distribution U. Given the classes are assumed to be sorted discrete scores, U is unlikely to be the true target prompt score distribution $p(y)$, since assigning extreme scores are less likely than the mid-range ones.

Based on this insight, we propose **P**rior-encoded **I**nformation **M**aximization (**PIM**), where we extract an informative prior $q(y)$ from the labeled source domains, and use it in place of U for MI maximization. To suit source-free adaptation, we pre-compute the marginal distribution over the scores for each source domain *before* removing the dataset. In what follows, we denote the subscript S_j as j for brevity. For j-th source domain, we first scale the scores $\{y_j^{(i)}\}_{i=1}^{N_j}$ to the $[0, 1]$ interval:

$$\tilde{y}_j^{(i)} = (y_j^{(i)} - a_j + 0.5)/(b_j - a_j + 1) \quad \forall i \quad (6)$$

where y_j is assumed to be an integer ranging from a_j to b_j. Next, we fit a Beta distribution $\mathbf{Beta}(\alpha_j, \beta_j)$ with the scaled scores via maximum likelihood estimation:

$$(\alpha_j, \beta_j) = \underset{(\alpha_j, \beta_j)}{\arg\max} \ \mathbb{E}_{\tilde{y}_j}\left[\mathbf{Beta}(\tilde{y}_j; \alpha_j, \beta_j)\right] \quad (7)$$

where $\mathbf{Beta}(\tilde{y}_j; \alpha_j, \beta_j)$ is the probability density at $\tilde{y}_j$. $\mathbf{Beta}(\alpha, \beta)$ is a suitable abstraction of a set of noisy (scaled) scores, given that it is unimodal when $\alpha > 1, \beta > 1$ and flexible in modeling the skewness of the distribution bounded by $[0, 1]$, just as essay scores being bounded and roughly unimodal.

During the *adaptation* stage, we unify all source Beta distributions into a single $\mathbf{Beta}(\alpha_S, \beta_S)$ to further reduce domain-specific noise. Essentially, we consider the mean μ and variance σ^2 of the mixture $1/M \sum_{j=1}^{M} \mathbf{Beta}(\alpha_j, \beta_j)$ and set $\mathbf{Beta}(\alpha_S, \beta_S)$ such that its mean and variance equal to μ and σ^2 (derivations in Appendix B.1). This unified distribution is then discretized into a categorical distribution $q(y) \in \mathbb{R}^{C_T}$ over the target prompt (sorted) score classes:

$$q_c(y) = \int_{\frac{c-1}{C_T}}^{\frac{c}{C_T}} \mathbf{Beta}(y; \alpha_S, \beta_S)\, dy, \text{for } c \in 1{:}C_T \quad (8)$$

yielding source-informed prior probabilities over C_T evenly spaced bins. Our intuition is that $q(y)$ offers better approximation to the true distribution $p(y)$ than U in general.

Finally, we define our PIM objective $f(\lambda)$ as maximizing the prior-encoded mutual information by inserting $q(y)$ in place of U:

$$f(\lambda) = -KL(p(y|\lambda)||q(y)) - \mathcal{H}(p(y|x, \lambda)) \quad (9)$$

Here, the objective is written in terms of the merging coefficients $\lambda = [\lambda_1, \ldots, \lambda_M]^\top \in \mathbb{R}^M$ to

explicitly state that the parameter θ is solely determined by λ in our chosen merging framework (Eq. 2). In the context of using LLMs, we obtain $p(y|x^{(i)}, \lambda) \in \mathbb{R}^{C_T}$ by truncating the next-token probabilities of the score-predicting token (the <assistant> token in our case) to C_T score tokens (e.g., "1", "2", "3"). The truncated probabilities are then normalized to form a valid distribution which sum to 1.

3.2 Bayesian Optimization

In determining the mixing coefficients λ, recent studies have shown success in using Bayesian optimization (Jang et al., 2024; Liu et al., 2024) which is computationally less demanding than training the coefficients (Wortsman et al., 2022). Following this approach, we leverage Bayesian optimization to maximize $f(\lambda)$ (Eq. 9) in terms of $\lambda \in \mathbb{R}^M$ without costly backpropagation.

Essentially, the algorithm treats $f(\lambda)$ as a black-box function and constructs a surrogate of $f(\lambda)$ as a sample from a Gaussian Process—a distribution over functions (Williams and Rasmussen, 2006). Given the prior mean function μ_0, covariance function Σ_0 and k observations $f(\lambda^{(1:k)}) := \{f(\lambda^{(i)})\}_{i=1}^{k}$, it updates the *posterior* distribution over the function value at the current $(k+1)$-th iteration, i.e., $f(\lambda^{(k+1)})|f(\lambda^{(1:k)})$. Next, the *acquisition function* determines where to sample $\lambda^{(k+1)}$ based on the posterior. In particular, we use Expected Improvement (**EI**) which maximizes the expected gain over the current best value $f^*(k) := \max_{\lambda^{(i)}}\{f(\lambda^{(i)})\}_{i=1}^{k}$:

$$\arg\max_{\lambda^{(k+1)}} \mathop{\mathbb{E}}_{f(\lambda^{(k+1)})}\Big[\max(f(\lambda^{(k+1)}) - f^*(k), 0)\Big] \tag{10}$$

This process of posterior estimation and next point sampling is repeated until convergence. The final solution is $\arg\max_{\lambda^{(i)}}\{f(\lambda^{(i)})\}_{i=1}^{N}$ for N total iterations. See Appendix B.2 for additional details.

4 Experiment

4.1 Experimental Setup

Datasets. We validate our approach on two scenarios: (1) in-dataset cross-prompt scoring and (2) cross-dataset cross-prompt scoring. All samples are formatted using a simple instruction template, as in Appendix C.1.

In-dataset cross-prompt scoring follows the standard setup (Jin et al., 2018; Ridley et al., 2020; Li and Ng, 2024) where each prompt in a dataset is held out as a target domain and the remaining prompts serve as source domains. We use ASAP[2] (Hamner et al., 2012) dataset, which includes essays written by students from grade 7 to 10 in response to 8 prompts accross various genres and score ranges. We adopt the same dataset splits from Ridley et al. (2021) where each prompt is split into training and validation sets approximately by 5.6 : 1. When a prompt serves as the target domain, its two splits are combined into the test set. Dataset statistics are shown in Table 2.

Dataset	Prompt	#Essay	Genre	Avg Len	Range
ASAP	1	1783	ARG	427	2-12
	2	1800	ARG	432	1-6
	3	1726	RES	124	0-3
	4	1772	RES	106	0-3
	5	1805	RES	142	0-4
	6	1800	RES	173	0-4
	7	1569	NAR	206	0-30
	8	723	NAR	725	0-60
PERSUADE2.0	1	1656	ARG	339	1-6
	2	2157	ARG	641	1-6
	3	1670	ARG	552	1-6
	4	1552	ARG	573	1-6
	5	1372	RES	330	1-6
	6	2046	RES	455	1-6
	7	1862	RES	399	1-6
	8	1583	RES	381	1-6

Table 2: Dataset Statistics. **Genre**: ARG (argumentative), RES (source-dependent), NAR (narrative). **Avg Len**: Average essay length in words. **Range**: Score range.

Cross-dataset cross-prompt scoring is a new setting we introduce to validate our approach and the baselines under severer distribution shifts. In particular, we use all prompts from ASAP as source domains and treat each prompt from PERSUADE2.0 (Crossley et al., 2024) as the target domain. PERSUADE2.0 contains essays written by U.S. students in response to 15 prompts, among which we choose 4 from independent writing and another 4 from source-based writing during evaluation. Essay topics of the prompts are listed in Appendix C.2.

Models & Adapters. We conduct supervised fine-tuning on Llama-3.1-8B-Instruct (Grattafiori et al., 2024) (8 billion parameters) and Phi-4-mini-instruct (4 billion parameters) (Microsoft et al., 2025), for each prompt from the source domains independently. Details of LoRA fine-tuning are in Appendix C.3. At inference, we use greedy decoding and parse the score from the model's response.

[2] https://www.kaggle.com/c/asap-aes/data

Model	Scheme	Method	P1	P2	P3	P4	P5	P6	P7	P8	Avg.
gpt-4.1-mini	zero-shot	-	0.063	0.423	0.459	0.672	0.480	0.624	0.321	0.390	0.429
llama-3.1-8b-it	zero-shot	-	0.109	0.246	0.239	0.240	0.361	0.407	0.321	0.484	0.301
	merge	Averaging	0.526	0.465	0.527	0.593	0.720	0.738	0.608	0.163	0.542*
		Fisher Merging	0.437	0.541	0.521	0.590	0.670	0.724	0.562	0.167	0.526*
		RegMean	0.482	0.461	0.526	0.580	0.724	0.731	0.580	0.135	0.527*
		Task Arithmetic	0.787	0.368	0.604	0.632	0.772	0.741	0.627	0.120	0.581*
		TIES-Merging	0.582	0.527	0.532	0.619	0.711	0.752	0.595	0.155	0.559*
		AdaMerging	0.756	0.285	0.577	0.619	0.767	0.664	0.661	0.059	0.548*
		PIM (Ours)	0.682	0.562	0.612	0.647	0.762	0.690	0.711	0.152	**0.602**
	joint-train	-	0.606	0.512	0.611	0.656	0.743	0.760	0.666	0.257	0.601
phi-4-mini-it	zero-shot	-	0.084	0.305	0.238	0.479	0.367	0.350	0.131	0.184	0.267
	merge	Averaging	0.383	0.613	0.494	0.625	0.528	0.652	0.396	0.334	0.503*
		Fisher Merging	0.348	0.625	0.490	0.617	0.503	0.637	0.389	0.299	0.489*
		RegMean	0.348	0.607	0.507	0.619	0.577	0.666	0.378	0.286	0.498*
		Task Arithmetic	0.772	0.334	0.618	0.654	0.690	0.684	0.683	0.211	0.581*
		TIES-Merging	0.532	0.568	0.512	0.625	0.542	0.681	0.448	0.291	0.525*
		AdaMerging	0.742	0.316	0.569	0.618	0.645	0.650	0.608	0.227	0.547*
		PIM (Ours)	0.737	0.585	0.637	0.612	0.731	0.654	0.692	0.387	**0.629**
	joint-train	-	0.578	0.469	0.622	0.655	0.668	0.740	0.590	0.376	0.587*

Table 3: **In-dataset** cross-prompt evaluation results on **ASAP** $\rightarrow$ **ASAP**, measured by QWK. **P1-8** denotes Prompt 1-8. For each held-out target prompt, other 7 prompts constitute the source domains. *: Significant improvement ($p < 0.05$) of our method over a merging baseline/joint-training in Avg. QWK. The best average QWK is boldfaced.

Baselines. We compare our approach against recent merging methods which we apply to source-free adaptation setting: Averaging (Wortsman et al., 2022), Fisher Merging (Matena and Raffel, 2022), RegMean (Jin et al., 2023), Task Arithmetic (Ilharco et al., 2023), TIES-Merging (Yadav et al., 2023) and AdaMerging (Yang et al., 2024c).

In addition, we report performance of joint-training on all sources, and top-performing cross-prompt methods—PAES(Ridley et al., 2020) and PMAES (Chen and Li, 2023), both of which train the model jointly on all source domains. Particularly, PMAES requires unlabeled target samples simultaneously. See Appendix C.4 for the descriptions and implementation details of the baselines.

Evaluation Metric. Following the standard evaluation protocol (Phandi et al., 2015; Cao et al., 2020; Chen and Li, 2023), we use the Quadratic Weighted Kappa (QWK) to measure the agreement between human-rated scores and predicted scores.

Implementation Details. We use Bayesian Optimization toolkit (Nogueira, 2014–), with the Expected Improvement ($\xi = 0.01$) acquisition function. We initially let the algorithm probe 10 random points, and iterate through 30 subsequent steps. Each coefficient λ_j is bounded by $[0, 1]$. We randomly select 64 unlabeled target prompt samples fixed across all iterations, and compute $p(y|x^{(i)}, \lambda)$ on those samples. If not otherwise stated, experimental results are averaged over 5 random seeds. Our method is robust to the choice of the number of iterations and unlabeled target prompt samples, as detailed in Appendix C.5.

4.2 Main Results

In-dataset Cross-prompt. Results are shown in Table 3. First, our merging approach matches or surpasses joint training on all sources, validating the effectiveness of selecting beneficial source domains for adaptation. In general, linear combination of task vectors underperforms its joint-training counterpart (Ilharco et al., 2023), which renders the progress of our method over joint-training impressive. Second, our method exceeds all merging baselines in average QWK, with the improvements observed to be statistically significant, highlighting the importance of merging strategy specifically designed for domain adaptation. Third, our method brings a notable improvement over zero-shot baselines, with average gains of 0.301 on Llama-3.1-8B-it and 0.362 on Phi-4-mini-it. It also outperforms zero-shot GPT-4.1-mini by 0.200, demonstrating the effectiveness of our method against an advanced LLM.

Cross-dataset Cross-prompt. Table 4 reports results on ASAP $\rightarrow$ PERSUADE2.0 transfer. Consistent with the in-dataset setting, our method shows improvements over joint training, e.g., by

Model	Scheme	Method	P1	P2	P3	P4	P5	P6	P7	P8	Avg.
llama-3.1-8b-it	zero-shot	-	0.136	0.363	0.278	0.309	0.043	0.120	0.166	0.139	0.194
	merge	Averaging	0.365	0.529	0.407	0.397	0.222	0.463	0.342	0.296	0.378*
		Fisher Merging	0.420	0.599	0.472	0.479	0.248	0.497	0.392	0.348	0.432*
		RegMean	0.340	0.513	0.391	0.399	0.208	0.441	0.322	0.291	0.363*
		Task Arithmetic	0.412	0.377	0.310	0.311	0.164	0.294	0.264	0.340	0.309*
		TIES-Merging	0.474	0.551	0.438	0.427	0.275	0.511	0.368	0.343	0.423*
		AdaMerging	0.396	0.122	0.116	0.151	0.153	0.216	0.206	0.270	0.204*
		PIM (Ours)	0.504	0.734	0.674	0.652	0.197	0.464	0.420	0.449	**0.512**
	joint-train	-	0.515	0.406	0.438	0.448	0.243	0.367	0.342	0.454	0.401*
phi-4-mini-it	zero-shot	-	0.181	0.216	0.397	0.515	0.341	0.345	0.352	0.386	0.342
	merge	Averaging	0.520	0.581	0.547	0.572	0.214	0.567	0.522	0.489	0.502
		Fisher Merging	0.533	0.594	0.564	0.579	0.224	0.577	0.529	0.500	**0.512**
		RegMean	0.502	0.557	0.530	0.567	0.206	0.568	0.508	0.488	0.491
		Task Arithmetic	0.337	0.353	0.347	0.295	0.110	0.330	0.274	0.278	0.291*
		TIES-Merging	0.513	0.613	0.543	0.518	0.199	0.565	0.505	0.490	0.493
		AdaMerging	0.474	0.149	0.200	0.156	0.162	0.360	0.407	0.365	0.284*
		PIM (Ours)	0.438	0.581	0.671	0.668	0.186	0.480	0.407	0.434	0.483
	joint-train	-	0.545	0.439	0.509	0.564	0.217	0.447	0.422	0.480	0.453*

Table 4: **Cross-dataset** cross-prompt evaluation results on **ASAP** → **PERSUADE2.0**, measured by QWK. For each target prompt in PERSUADE2.0, all 8 prompts in ASAP constitute the source domains.

0.111 on Llama3.1-8B-it and 0.030 on Phi-4-mini-it. Compared to the merging baselines, our method achieves the highest average QWK on Llama-3.1-8B-it and falls slightly short of some baselines on Phi-4-mini-it. Nevertheless, these baselines show limited generalizability, as they underperform on the other 3 settings (Table 3, 4) with larger drops, while our method maintains the best results. Overall, our approach shows robust adaptation on different types of domain shifts and LLMs.

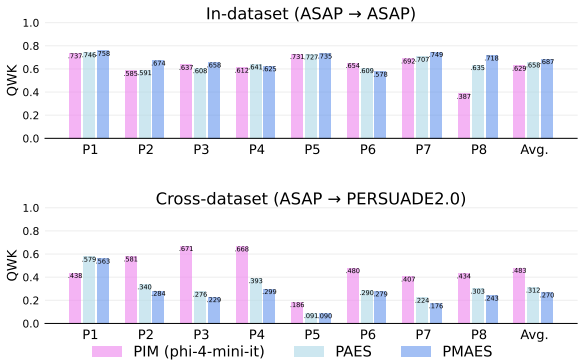

Figure 3: Comparison of PIM (phi-4-mini-it) with top-performing cross-prompt methods (PAES and PMAES). Similar trends for llama-3.1-8b-it (Appendix D).

Comparison with Leading Cross-prompt Methods. Figure 3 presents a comparison between our method against PAES (Ridley et al., 2020) and PMAES (Chen and Li, 2023), two strong baselines on **ASAP** → **ASAP**. PAES is a regression model that combines hand-crafted features and CNN+LSTM representations, while PMAES extends PAES with a domain adaptation strategy. We highlight that LLM-based autoregressive scoring does not inherently outperform PAES despite the larger parameter size, as seen when comparing the joint-train baseline (Table 3) with PAES (Figure 3 **top**). Similarly, we found that even a competitive encoder model, DeBERTaV3-base (He et al., 2021), underperforms PAES in the in-dataset setting with an average QWK of 0.528. This is largely due to the carefully designed hand-crafted features which alone achieves 0.641 average QWK (Ridley et al., 2020).

Under the in-dataset setting (Figure 3 **top**), our method achieves QWKs close to those of both baselines across most prompts, though it falls slightly short on average. In the more challenging cross-dataset setting (Figure 3 **bottom**), however, our method notably outperforms PAES and PMAES with larger margins than the in-dataset's case. This change in relative performance under larger distribution shifts may stem from the reliance of PAES and PMAES on the domain-sensitive feature engineering and all-source joint-training, which are potentially prone to negative transfer. By contrast, our method adaptively selects source domains to mitigate negative transfer.

4.3 Ablation Study

We conduct an ablation study of the components of our method, as in Table 5. First, reverting the useful prior back to the uniform distribution ($q(y) \rightarrow U$)

Method	Phi4-mini	L3.1-8B
PIM	**0.629**	0.602
$q(y) \rightarrow U$	0.594	0.590
w/o $-\mathcal{H}(p(y\|x,\lambda))$	0.620	**0.617**
w/o $-KL(p(y\|\lambda)\|\|q(y))$	0.542	0.552
BayesOpt $\rightarrow$ Random	0.611	0.595
joint-train	0.587	0.601

Table 5: Ablation study of PIM (Eq. 9), evaluated on ASAP $\rightarrow$ ASAP, measured by average QWK. w/o: without; BayesOpt $\rightarrow$ Random: same number of iterations with random search.

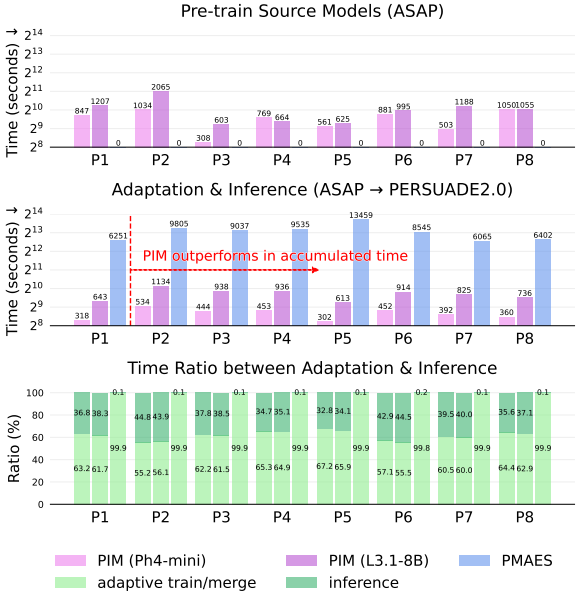

Figure 4: Log-scale time (y-axis) for pre-training source models on ASAP (**top**), followed by adaptation and inference on PERSUADE2.0 (**middle**), along with the time ratios between adaptation and inference (**bottom**). Results are from a single run on an NVIDIA A40 GPU.

leads to a notable drop in performance, which aligns with our motivation that U may not appropriately represent the target's marginal distribution $p(y)$ in essay scoring. The prior $q(y)$ derived from the source prompts serves as a transferrable supervision signal, without any expert knowledge. Second, we challenge the MI maximization principle (Krause et al., 2010), either by removing the sharpness term (i.e., w/o $-\mathcal{H}(p(y|x,\lambda))$) or the separation term (i.e., w/o $-KL(p(y|\lambda)||q(y))$). Interestingly, the former does not necessarily lead to performance drop, with Llama-3.1-8B-it in fact achieving some gains. One hypothesis is that each source model (LoRA adapter) yield sufficiently sharp predictions on the target samples, and that merging the source models via linear combination preserves this property. In contrast, removing separation term leads to a significant drop, possibly due to its tendency to favor an overconfident model which lacks diversity in predictions. Third, given the same number of iterations, Bayesian optimization yields higher QWK than random search, confirming guided exploration.

4.4 Cost Analysis

In Figure 4, we analyze the wall-clock time of PIM compared with PMAES, throughout the entire course of ASAP $\rightarrow$ PERSUADE2.0 adaptation. We note that PMAES joint-trains the model for all available sources for each target prompt. Notably, PIM requires substantially less time for adaptation and inference combined than PMAES, once all source models are pre-trained. When accounting for the accumulated time from pre-training, PIM begins to outperform PMAES from the second target prompt, with the gap widening as more target prompts are introduced. This highlights PIM's scalability to new prompts, despite using considerably larger models (LLMs) than PMAES (CNN+LSTM). The efficiency arises from the constant reuse of individual source models and efficient merging. On the other hand, PMAES retrains the entire model for each target, leading to relatively high time cost for adaptation.

5 Related Work

Cross-prompt Essay Scoring. Cross-prompt AES transfers models trained on source prompts to unseen ones (Dong and Zhang, 2016; Ridley et al., 2021). Early work used manual features and domain adaptation with a few labeled target samples (Phandi et al., 2015; Cummins et al., 2016). Later neural methods improved generalization by incorporating prompt-agnostic objectives (Ridley et al., 2020) or learning multi-prompt joint representations (Cao et al., 2020; Chen and Li, 2023), but rely on static data fusion and full retraining, which limits the scalability. Our approach differs by enabling dynamic and selective utilization of source-domain knowledge without joint-training.

Model Merging. Model merging linearly combines parameters from same-architecture networks while preserving properties (Neyshabur et al., 2020; Zhou et al., 2023). Current methods include magnitude pruning (Yadav et al., 2023; Yu et al., 2024; Deep et al., 2024; Gargiulo et al., 2025; Marczak et al., 2025) to reduce parameter conflicts; activa-

tion merging (Yang et al., 2024a,b; Xu et al., 2025) to align features; optimization merging (Matena and Raffel, 2022; Jin et al., 2023; Yang et al., 2024c) to adjust weights via optimization. Emerging studies (Team et al., 2025; Sun et al., 2025) have demonstrated the effectiveness of model merging on LLMs, presenting a potential pathway for enhancing out-of-distribution performance.

Source-free Domain Adaptation. SFDA transfers models pretrained on labeled source domain(s) to unlabeled target domain without source data (Sun et al., 2020). One approach includes compensating for the absence of source data by generating virtual samples (Tian et al., 2022; Ding et al., 2022) or computing summary statistics (Adachi et al., 2025). Another approach is adapting solely with unlabeled target data by minimizing entropy (Wang et al., 2021; Niu et al., 2022), prompting prediction diversity (Liang et al., 2020; Dong et al., 2021), or Bayesian calibration (Zhou and Levine, 2021). Applications of SFDA in NLP are few, but growing (Zhang et al., 2021; Yin et al., 2024). This paper addresses SFDA for essay scoring by leveraging source statistics, with model merging as a scalable alternative to training for adaptation.

6 Conclusion

In this paper, we propose a domain-adaptive model merging approach for source-free cross-prompt AES. Our pilot study suggests suboptimality of training on all source domains. Inspired by this, we shift to selecting beneficial source domains, and approximate costly joint training by merging task vectors through a linear combination. In optimizing the combination's coefficients, we resort to our proposed prior-encoded information maximization (PIM), an unsupervised objective which encourages score discriminability regularized by priors pre-computed from the sources. Experimental results with LLMs on in- and cross-dataset settings show that our method consistently outperforms joint training on all sources, surpasses other merging methods in numerous cases, maintains robustness under severe distribution shifts where leading cross-prompt methods struggle, all while remaining computationally efficient.

Limitations

We elucidate the limitations of this work as follows: First, both mutual information maximization and our improved PIM rely on the assumption that at least one source model provides reasonable predictions (e.g., with sufficient diversity) for the target prompt. If all source models fail to capture the semantics of the target prompt, optimizing the PIM objective may degrade performance arbitrarily, as encouraging discriminability and sharpness becomes meaningless without meaningful initial predictions. Second, while our adaptation process remains efficient (using a fixed small sample of target essays, e.g., 64), PIM is designed for LLMs, which incur significantly higher inference latency compared to conventional encoder-based AES models. This may limit scalability when each target prompt contains a very large volume of essays to be tested, despite the adaptation itself being sample-efficient. Third, although LLMs enable adaptation to novel score ranges, extreme deviations between source and target ranges can lead to suboptimal predictions. For instance, on P8 of ASAP with score range of $[0, 60]$, some source models lacked diverse predictions.

Ethics Statement

Potential Risks This work aims to improve cross-prompt AES performance of LLMs. However, our method does not guarantee the model's fairness of scoring. For instance, it is possible that the adapted model assigns the scores in favor of a certain social group, such as the essay writer's first language background, gender, etc. In addition, since the source datasets may disproportionately represent certain social groups, models trained on these datasets could reproduce the biases embedded in the datasets in their predictions. There are ongoing works analyzing the fairness of AES systems (Loukina et al., 2019; Schaller et al., 2024), and it is recommended to refer to this field before the deployment of the system.

Use of Scientific Artifacts For the datasets, we used ASAP (Hamner et al., 2012)'s publicly available text corpora, and used PERSUADE2.0 (Crossley et al., 2024) which is an open source corpus under CC BY-NC-SA 4.0 license. Both ASAP and PERSUADE2.0 have anonymized personally identifying information from the essays. For the models, Llama-3.1-8B-Instruct (Grattafiori et al., 2024) is under Llama3.1 Community License, and Phi-4-mini-instruct (Microsoft et al., 2025) is under MIT license. In addition, Bayesian Optimization (Nogueira, 2014–) toolkit is under MIT license. All of these artifacts is applicable for research use.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (62076008).

References

Kazuki Adachi, Shin'ya Yamaguchi, Atsutoshi Kumagai, and Tomoki Hamagami. 2025. Test-time adaptation for regression by subspace alignment. In *The Thirteenth International Conference on Learning Representations*.

Samuel Ainsworth, Jonathan Hayase, and Siddhartha Srinivasa. 2023. Git re-basin: Merging models modulo permutation symmetries. In *The Eleventh International Conference on Learning Representations*.

John Bridle, Anthony Heading, and David MacKay. 1991. Unsupervised classifiers, mutual information and 'phantom targets. In *Advances in Neural Information Processing Systems*, volume 4. Morgan-Kaufmann.

Yue Cao, Hanqi Jin, Xiaojun Wan, and Zhiwei Yu. 2020. Domain-adaptive neural automated essay scoring. In *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval*, pages 1011–1020.

Hongbo Chen and Ben He. 2013. Automated essay scoring by maximizing human-machine agreement. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1741–1752, Seattle, Washington, USA. Association for Computational Linguistics.

Yuan Chen and Xia Li. 2023. PMAES: Prompt-mapping contrastive learning for cross-prompt automated essay scoring. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1489–1503, Toronto, Canada. Association for Computational Linguistics.

Yuan Chen and Xia Li. 2024. PLAES: Prompt-generalized and level-aware learning framework for cross-prompt automated essay scoring. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 12775–12786, Torino, Italia. ELRA and ICCL.

SeongYeub Chu, Jong Woo Kim, Bryan Wong, and Mun Yong Yi. 2025. Rationale behind essay scores: Enhancing S-LLM's multi-trait essay scoring with rationale generated by LLMs. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 5796–5814, Albuquerque, New Mexico. Association for Computational Linguistics.

Mădălina Cozma, Andrei Butnaru, and Radu Tudor Ionescu. 2018. Automated essay scoring with string kernels and word embeddings. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 503–509, Melbourne, Australia. Association for Computational Linguistics.

S.A. Crossley, Y. Tian, P. Baffour, A. Franklin, M. Benner, and U. Boser. 2024. A large-scale corpus for assessing written argumentation: Persuade 2.0. *Assessing Writing*, 61:100865.

Ronan Cummins, Meng Zhang, and Ted Briscoe. 2016. Constrained Multi-Task Learning for Automated Essay Scoring. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 789–799, Berlin, Germany. Association for Computational Linguistics.

Pala Tej Deep, Rishabh Bhardwaj, and Soujanya Poria. 2024. Della-merging: Reducing interference in model merging through magnitude-based sampling. *CoRR*, abs/2406.11617.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Ning Ding, Yixing Xu, Yehui Tang, Chao Xu, Yunhe Wang, and Dacheng Tao. 2022. Source-free domain adaptation via distribution estimation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7202–7212.

Heejin Do, Yunsu Kim, and Gary Geunbae Lee. 2023. Prompt- and trait relation-aware cross-prompt essay trait scoring. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1538–1551, Toronto, Canada. Association for Computational Linguistics.

Fei Dong and Yue Zhang. 2016. Automatic features for essay scoring – an empirical study. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1072–1077, Austin, Texas. Association for Computational Linguistics.

Fei Dong, Yue Zhang, and Jie Yang. 2017. Attention-based recurrent convolutional neural network for automatic essay scoring. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 153–162, Vancouver, Canada. Association for Computational Linguistics.

Jiahua Dong, Zhen Fang, Anjin Liu, Gan Sun, and Tongliang Liu. 2021. Confident anchor-induced multi-source free domain adaptation. In *Advances in Neural Information Processing Systems*, volume 34, pages 2848–2860. Curran Associates, Inc.

Youmna Farag, Helen Yannakoudakis, and Ted Briscoe. 2018. Neural automated essay scoring and coherence modeling for adversarially crafted input. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 263–271, New Orleans, Louisiana. Association for Computational Linguistics.

Peter I Frazier. 2018. A tutorial on bayesian optimization. *arXiv preprint arXiv:1807.02811*.

Antonio Andrea Gargiulo, Donato Crisostomi, Maria Sofia Bucarelli, Simone Scardapane, and Emanuele Rodolà. 2025. Task Singular Vectors: Reducing Task Interference in Model Merging. *Preprint*, arXiv:2412.00081.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Ben Hamner, Jaison Morgan, lynnvandev, Mark Shermis, and Tom Vander Ark. 2012. The hewlett foundation: Automated essay scoring.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021. Debertav3: Improving deberta using electra-style pretraining with gradient-disentangled embedding sharing. *arXiv preprint arXiv:2111.09543*.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Jiaxing Huang, Dayan Guan, Aoran Xiao, and Shijian Lu. 2021. Model adaptation: Historical contrastive learning for unsupervised domain adaptation without source data. In *Advances in Neural Information Processing Systems*, volume 34, pages 3635–3649. Curran Associates, Inc.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. 2023. Editing models with task arithmetic. In *The Eleventh International Conference on Learning Representations*.

Chaeyun Jang, Hyungi Lee, Jungtaek Kim, and Juho Lee. 2024. Model fusion through bayesian optimization in language model fine-tuning. In *Advances in Neural Information Processing Systems*, volume 37, pages 29878–29912. Curran Associates, Inc.

Zhiwei Jiang, Tianyi Gao, Yafeng Yin, Meng Liu, Hua Yu, Zifeng Cheng, and Qing Gu. 2023. Improving domain generalization for prompt-aware essay scoring via disentangled representation learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12456–12470, Toronto, Canada. Association for Computational Linguistics.

Cancan Jin, Ben He, Kai Hui, and Le Sun. 2018. TDNN: A two-stage deep neural network for prompt-independent automated essay scoring. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1088–1097, Melbourne, Australia. Association for Computational Linguistics.

Xisen Jin, Xiang Ren, Daniel Preotiuc-Pietro, and Pengxiang Cheng. 2023. Dataless knowledge fusion by merging weights of language models. In *The Eleventh International Conference on Learning Representations*.

Andreas Krause, Pietro Perona, and Ryan Gomes. 2010. Discriminative clustering by regularized information maximization. In *Advances in Neural Information Processing Systems*, volume 23. Curran Associates, Inc.

Shengjie Li and Vincent Ng. 2024. Conundrums in cross-prompt automated essay scoring: Making sense of the state of the art. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7661–7681, Bangkok, Thailand. Association for Computational Linguistics.

Jian Liang, Dapeng Hu, and Jiashi Feng. 2020. Do we really need to access the source data? Source hypothesis transfer for unsupervised domain adaptation. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 6028–6039. PMLR.

Deyuan Liu, Zecheng Wang, Bingning Wang, Weipeng Chen, Chunshan Li, Zhiying Tu, Dianhui Chu, Bo Li, and Dianbo Sui. 2024. Checkpoint merging via bayesian optimization in llm pretraining. *arXiv preprint arXiv:2403.19390*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Anastassia Loukina, Nitin Madnani, and Klaus Zechner. 2019. The many dimensions of algorithmic fairness in educational applications. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 1–10, Florence, Italy. Association for Computational Linguistics.

Michal Lukasik, Zhao Meng, Harikrishna Narasimhan, Yin-Wen Chang, Aditya Krishna Menon, Felix Yu, and Sanjiv Kumar. 2025. Better autoregressive regression with LLMs via regression-aware fine-tuning. In *The Thirteenth International Conference on Learning Representations*.

Daniel Marczak, Simone Magistri, Sebastian Cygert, Bartłomiej Twardowski, Andrew D. Bagdanov, and Joost van de Weijer. 2025. No Task Left Behind: Isotropic Model Merging with Common and Task-Specific Subspaces. *Preprint*, arXiv:2502.04959.

Michael S Matena and Colin A Raffel. 2022. Merging models with fisher-weighted averaging. In *Advances in Neural Information Processing Systems*, volume 35, pages 17703–17716. Curran Associates, Inc.

Microsoft, :, Abdelrahman Abouelenin, Atabak Ashfaq, Adam Atkinson, Hany Awadalla, Nguyen Bach, Jianmin Bao, Alon Benhaim, Martin Cai, Vishrav Chaudhary, Congcong Chen, Dong Chen, Dongdong Chen, Junkun Chen, Weizhu Chen, Yen-Chun Chen, Yi ling Chen, Qi Dai, and 57 others. 2025. Phi-4-mini technical report: Compact yet powerful multimodal language models via mixture-of-loras. *Preprint*, arXiv:2503.01743.

Behnam Neyshabur, Hanie Sedghi, and Chiyuan Zhang. 2020. What is being transferred in transfer learning? *Advances in neural information processing systems*, 33:512–523.

Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Shijian Zheng, Peilin Zhao, and Mingkui Tan. 2022. Efficient test-time model adaptation without forgetting. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 16888–16905. PMLR.

Fernando Nogueira. 2014–. Bayesian Optimization: Open source constrained global optimization tool for Python.

Peter Phandi, Kian Ming A. Chai, and Hwee Tou Ng. 2015. Flexible Domain Adaptation for Automated Essay Scoring Using Correlated Linear Regression. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 431–439, Lisbon, Portugal. Association for Computational Linguistics.

Robert Ridley, Liang He, Xin-yu Dai, Shujian Huang, and Jiajun Chen. 2021. Automated cross-prompt scoring of essay traits. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 13745–13753.

Robert Ridley, Liang He, Xinyu Dai, Shujian Huang, and Jiajun Chen. 2020. Prompt agnostic essay scorer: a domain generalization approach to cross-prompt automated essay scoring. *arXiv preprint arXiv:2008.01441*.

Nils-Jonathan Schaller, Yuning Ding, Andrea Horbach, Jennifer Meyer, and Thorben Jansen. 2024. Fairness in automated essay scoring: A comparative analysis of algorithms on German learner essays from secondary education. In *Proceedings of the 19th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2024)*, pages 210–221, Mexico City, Mexico. Association for Computational Linguistics.

Lin Sun, Guangxiang Zhao, Xiaoqi Jian, Yuhan Wu, Weihong Lin, Yongfu Zhu, Change Jia, Linglin Zhang, Jinzhu Wu, Junfeng Ran, Sai-er Hu, Zihan Jiang, Junting Zhou, Wenrui Liu, Bin Cui, Tong Yang, and Xiangzheng Zhang. 2025. TinyR1-32B-Preview: Boosting Accuracy with Branch-Merge Distillation. *Preprint*, arXiv:2503.04872.

Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. 2020. Test-time training with self-supervision for generalization under distribution shifts. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 9229–9248. PMLR.

Kaveh Taghipour and Hwee Tou Ng. 2016. A neural approach to automated essay scoring. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1882–1891, Austin, Texas. Association for Computational Linguistics.

Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun Xiao, Chenzhuang Du, Chonghua Liao, Chuning Tang, Congcong Wang, Dehao Zhang, Enming Yuan, Enzhe Lu, Fengxiang Tang, Flood Sung, Guangda Wei, Guokun Lai, and 75 others. 2025. Kimi k1.5: Scaling Reinforcement Learning with LLMs. *Preprint*, arXiv:2501.12599.

Jiayi Tian, Jing Zhang, Wen Li, and Dong Xu. 2022. Vdm-da: Virtual domain modeling for source data-free domain adaptation. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(6):3749–3760.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. 2021. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*.

Jiong Wang and Jie Liu. 2025. T-MES: Trait-aware mix-of-experts representation learning for multi-trait essay scoring. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 1224–1236, Abu Dhabi, UAE. Association for Computational Linguistics.

Christopher KI Williams and Carl Edward Rasmussen. 2006. *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA.

Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, and 1 others. 2022. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *International conference on machine learning*, pages 23965–23998. PMLR.

Jing Xu, Jiazheng Li, and Jingzhao Zhang. 2025. Scalable Model Merging with Progressive Layer-wise Distillation. *Preprint*, arXiv:2502.12706.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin A Raffel, and Mohit Bansal. 2023. Ties-merging: Resolving interference when merging models. In *Advances in Neural Information Processing Systems*, volume 36, pages 7093–7115. Curran Associates, Inc.

Enneng Yang, Li Shen, Zhenyi Wang, Guibing Guo, Xiaojun Chen, Xingwei Wang, and Dacheng Tao. 2024a. Representation surgery for multi-task model merging. In *International Conference on Machine Learning*, pages 56332–56356. PMLR.

Enneng Yang, Li Shen, Zhenyi Wang, Guibing Guo, Xingwei Wang, Xiaocun Cao, Jie Zhang, and Dacheng Tao. 2024b. SurgeryV2: Bridging the Gap Between Model Merging and Multi-Task Learning with Deep Representation Surgery. *Preprint*, arXiv:2410.14389.

Enneng Yang, Zhenyi Wang, Li Shen, Shiwei Liu, Guibing Guo, Xingwei Wang, and Dacheng Tao. 2024c. Adamerging: Adaptive model merging for multi-task learning. In *The Twelfth International Conference on Learning Representations*.

Shiqi Yang, yaxing wang, kai wang, Shangling Jui, and Joost van de Weijer. 2022. Attracting and dispersing: A simple approach for source-free domain adaptation. In *Advances in Neural Information Processing Systems*, volume 35, pages 5802–5815. Curran Associates, Inc.

Maxwell Yin, Boyu Wang, and Charles Ling. 2024. Source-free unsupervised domain adaptation for question answering via prompt-assisted self-learning. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 700–713, Mexico City, Mexico. Association for Computational Linguistics.

Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. 2024. Language models are super mario: Absorbing abilities from homologous models as a free lunch. In *International Conference on Machine Learning*, pages 57755–57775. PMLR.

Torsten Zesch, Michael Wojatzki, and Dirk Scholten-Akoun. 2015. Task-independent features for automated essay grading. In *Proceedings of the Tenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 224–232, Denver, Colorado. Association for Computational Linguistics.

Bo Zhang, Xiaoming Zhang, Yun Liu, Lei Cheng, and Zhoujun Li. 2021. Matching distributions between model and data: Cross-domain knowledge distillation for unsupervised domain adaptation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5423–5433, Online. Association for Computational Linguistics.

Aurick Zhou and Sergey Levine. 2021. Bayesian adaptation for covariate shift. In *Advances in Neural Information Processing Systems*, volume 34, pages 914–927. Curran Associates, Inc.

Zhanpeng Zhou, Yongyi Yang, Xiaojiang Yang, Junchi Yan, and Wei Hu. 2023. Going beyond linear mode connectivity: The layerwise linear feature connectivity. *Advances in neural information processing systems*, 36:60853–60877.

A Details of Pilot Study

In the pilot study (Figure 1), we investigate the effect of jointly training on varying number of source datasets on the transfer performance. In this experiment, we train BERT with the following configurations: a regression head is placed on top of BERT encoder, which consists of a linear layer and a sigmoid activation; We train 30 epochs with a batch size of 128, a learning rate of $2 \cdot 10^{-5}$ with constant learning schedule; We choose the best checkpoint on the validation set among the epochs, with early stopping of 10 epochs.

B Supplementary Details of Method

B.1 Matching Mean and Variance of Beta Mixture

We describe how the unified $\mathbf{Beta}(\alpha_{\mathcal{S}}, \beta_{\mathcal{S}})$ is derived by matching its mean and variance to the Beta mixture $1/M \sum_{j=1}^{M} \mathbf{Beta}(\alpha_j, \beta_j)$. For each source distribution $\mathbf{Beta}(\alpha_j, \beta_j)$, the mean μ_j and variance σ_j^2 are

$$\mu_j = \frac{\alpha_j}{\alpha_j + \beta_j}, \sigma_j^2 = \frac{\alpha_j \beta_j}{(\alpha_j + \beta_j)^2(\alpha_j + \beta_j + 1)}.$$

The mean μ and variance σ^2 of the mixture is then given by

$$\mu = M^{-1} \sum_{j=1}^{M} \mu_j$$

$$\sigma^2 = M^{-1} \sum_{j=1}^{M} (\sigma_j^2 + \mu_j^2) - \mu^2.$$

Finally, we let $\mathbf{Beta}(\alpha_{\mathcal{S}}, \beta_{\mathcal{S}})$ have μ and σ^2 as its mean and variance, which is:

$$\alpha_{\mathcal{S}} = \mu \left(\frac{\mu(1-\mu)}{\sigma^2} - 1 \right)$$

$$\beta_{\mathcal{S}} = (1-\mu) \left(\frac{\mu(1-\mu)}{\sigma^2} - 1 \right).$$

B.2 Details of Bayesian Optimization

Bayesian optimization (Williams and Rasmussen, 2006) constructs a surrogate model of the black-box function $f(\lambda)$ as a sample from a Gaussian Process—a distribution over functions, and updates the posterior on f given observations $\{f(\lambda^{(i)})\}_{i=1}^{k}$. It then uses an *acquisition function* to determine where to sample $\lambda^{(k+1)}$ next. When the iteration terminates, $\lambda^* = \arg\max_{\lambda^{(i)}} f(\lambda^{(i)})$ is chosen as the final solution.

In detail, for $(k+1)$-iteration, the Gaussian prior is placed on the observations:

$$f(\lambda^{(1:k)}) \sim \mathcal{N}(\mu_0(\lambda^{(1:k)}), \Sigma_0(\lambda^{(1:k)}, \lambda^{(1:k)}))$$

where $\lambda^{(1:k)}$ is a compact notation for k points $\{f(\lambda^{(i)})\}_{i=1}^{k}$, and μ_0 and Σ_0 are the mean and covariance function of the Gaussian Process. We choose the commonly used $\mathbf{0}$ for μ_0 and Matern 2.5 kernel (Williams and Rasmussen, 2006) for Σ_0. Then the posterior on a new function value $f(\lambda^{(k+1)})$ given previous observations $f(\lambda^{(1:k)})$ is updated by the Bayes' rule (Frazier, 2018):

$$f(\lambda^{(k+1)})|f(\lambda^{(1:k)}) \sim \mathcal{N}(\mu_{k+1}(\lambda^{(k+1)}), \sigma^2_{k+1}(\lambda^{(k+1)}))$$

where

$$\mu_{k+1}(\lambda^{(k+1)}) = \Sigma_0(\lambda^{(k+1)}, \lambda^{(1:k)}) \cdot \Sigma_0(\lambda^{(1:k)}, \lambda^{(1:k)})^{-1} \cdot (f(\lambda^{(1:k)}) - \mu_0(\lambda^{(1:k)})) + \mu_0(\lambda^{(k+1)})$$

$$\sigma^2_{k+1}(\lambda^{(k+1)}) = \Sigma_0(\lambda^{(k+1)}, \lambda^{(k+1)}) - \Sigma_0(\lambda^{(k+1)}, \lambda^{(1:k)}) \cdot \Sigma_0(\lambda^{(1:k)}, \lambda^{(1:k)})^{-1}\Sigma_0(\lambda^{(1:k)}, \lambda^{(k+1)}).$$

Intuitively, the posterior mean $\mu_{k+1}(\lambda^{(k+1)})$ is a weighted sum between the prior $\mu_0(\lambda^{(k+1)})$ and a calibration term based on the data $f(\lambda^{(1:k)})$, and the posterior variance $\sigma^2_{k+1}(\lambda^{(k+1)})$ is given as the prior variance $\Sigma_0(\lambda^{(k+1)}, \lambda^{(k+1)})$ subtracted by the reduction in variance (uncertainty) after observing the data $f(\lambda^{(1:k)})$ (Frazier, 2018).

Next, the acquisition function specifies where to sample $\lambda^{(k+1)}$ based on the posterior. We use Expected Improvement (**EI**) which finds $\lambda^{(k+1)}$ such that the expected gain over the current best value $f^*(k) := \max_{\lambda^{(i)}}\{f(\lambda^{(i)})\}_{i=1}^{k}$ is maximized:

$$\arg\max_{\lambda^{(k+1)}} \mathop{\mathbb{E}}_{f(\lambda^{(k+1)})} \Big[\max(f(\lambda^{(k+1)}) - f^*(k), 0)\Big]$$

This process of posterior estimation and next point sampling is repeated until convergence.

C Additional Details of Experimental Setup

C.1 Instruction Template

We use the instruction template below throughout the experiments.

User Message

```
### Prompt:
{prompt}
### Student Essay:
{essay}
### Instruction:
Given the student's essay written in response to the prompt, assign a score within the range of {min_score} to {max_score}. Respond with only an integer score and no additional text.
```

Assistant Message

```
{score}
```

C.2 Prompt Topics

In Table 6, we specify the correspondence between the prompt IDs (Table. 2) and the prompt topics.

Dataset	Prompt	Topic
ASAP	1	Effects computers have on people
	2	Censorship in the libraries
	3	Impact of setting on the cyclist's experience
	4	The meaning of the ending in Winter Hibiscus
	5	The mood created in Narciso Rodriguez's memoir
	6	Obstacles to docking dirigibles
	7	A story about patience
	8	A story about laughter
PERSUADE2.0	1	Cell phones at school
	2	Distance learning
	3	Mandatory extracurricular activities
	4	Seeking multiple opinions
	5	"A Cowboy Who Rode the Waves"
	6	Does the electoral college work?
	7	Exploring Venus
	8	The Face on Mars

Table 6: Prompt Topics of ASAP and PERSUADE2.0.

C.3 Details of LoRA Fine-tuning

We use LoRA with $r = 16$, $\alpha = 32$, dropout $= 0.1$, targeting all linear layers in the transformer block (Vaswani et al., 2017). During fine-tuning we use the AdamW optimizer (Loshchilov and Hutter, 2019) with a batch size of 16, a learning rate of 10^{-4} and a cosine scheduler. The best checkpoint on the validation set is selected, with evaluation steps of 30 and early stopping patience of 3.

C.4 Descriptions and Implementation Details of Baselines

Averaging (Wortsman et al., 2022) simply averages the models' parameters. **Task Arithmetic** (Ilharco et al., 2023) adds a scaled sum of task vectors to the pre-trained model, and **TIES-Merging** (Yadav et al., 2023) pre-processes task vectors to resolve their interferences prior to merging. Following the recommended hyperparameters, we set the scaling factor of TA to $\lambda = 0.4$ and of TIES to $\lambda = 1.0$.

Fisher Merging (Matena and Raffel, 2022) improves Averaging by accounting for parameter-wise importance using Fisher information. Fisher information is estimated by sampling from the label distribution of samples from the validation set. **RegMean** (Jin et al., 2023) aims to minimize the layer-wise distance in activation between the merged model and all fine-tuned models. Following the original implementation on T5 models, we set the non-diagonal multiplier to $\alpha = 0.1$.

AdaMerging (Yang et al., 2024c) is a test time adaptation method which trains layer-wise coefficients for merging task vectors in order to minimize entropy on test samples. In our domain adaptation setting, test samples are the samples from the target domain. In contrast to other baselines, AdaMerging exploits the information of the target domain samples.

As for **Joint-train** baseline, we use the same configuration as C.3 except for batch size $= 64$ and early stopping patience $= 10$. For another joint-training baselines, **PAES** (Ridley et al., 2020) and **PMAES** (Chen and Li, 2023), we follow the original settings for model architecture and training hyperparameters. As the original implementation of PMAES does not specify the batch size, we set the combined batch size for the source and target domains data to 32, and allocate it proportionally based on the data ratio between the source and target domains.

C.5 Impact of Hyperparameter Choices

We examine the impact of hyperparameter choices on the performance of PIM. Specifically, we present results for varying the number of iterations in Table 7 and varying the number of unlabeled target prompt samples in Table 8. Table 7 shows that reducing half of the number of iterations ($40 \rightarrow 20$) yields a comparable average QWK, suggesting that convergence may occur as early as the 20th iteration. Table 8 indicates that PIM's performance is robust to the choice of the number of target prompt samples leveraged during Bayesian optimization.

Setting	Method	# of iterations	Avg. QWK
ASAP → ASAP	PIM (llama3.1-8b-it)	40 (10+30)	0.602
		20 (5+15)	0.600
	PIM (phi-4-mini-it)	40 (10+30)	0.629
		20 (5+15)	0.632
ASAP → PERSUADE2.0	PIM (llama3.1-8b-it)	40 (10+30)	0.512
		20 (5+15)	0.498
	PIM (phi-4-mini-it)	40 (10+30)	0.483
		20 (5+15)	0.486

Table 7: Impact of varying the number of iterations for on performance. The iterations consist of the initial random steps plus the optimization steps.

Setting	Method	# of samples	Avg. QWK
ASAP → ASAP	PIM (llama3.1-8b-it)	16	0.593
		32	0.603
		64	0.602
		128	0.603
	PIM (phi-4-mini-it)	16	0.629
		32	0.633
		64	0.629
		128	0.625
ASAP → PERSUADE2.0	PIM (llama3.1-8b-it)	16	0.528
		32	0.514
		64	0.512
		128	0.513
	PIM (phi-4-mini-it)	16	0.497
		32	0.483
		64	0.483
		128	0.489

Table 8: Impact of varying the number of (unlabeled) test samples on performance.

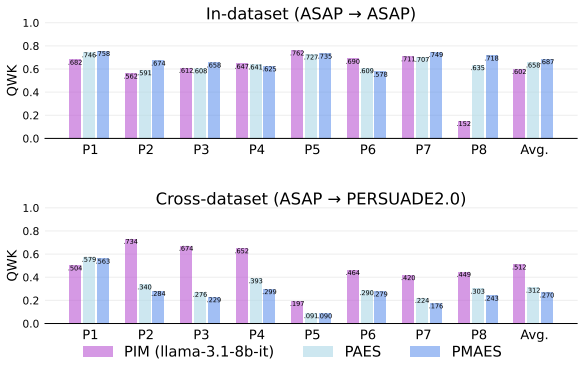

Figure 5: Comparison of PIM (llama-3.1-8b-it) with top-performing cross-prompt methods (PAES and PMAES).

D Additional Comparison with Leading Cross-prompt Methods

In Figure 5, we show additional comparison results between PIM (llama-3.1-8b-it) and leading cross-prompt AES methods—PAES (Ridley et al., 2020) and PMAES (Chen and Li, 2023). The overall trend is consistent with PIM (phi-4-mini-it).