Mixture of Ordered Scoring Experts for Cross-prompt Essay Trait Scoring

Po-Kai Chen[3], **Bo-Wei Tsai**[3], **Kuan-Wei Shao**[1],
Chien-Yao Wang[2], **Jia-Ching Wang**[3], and **Yi-Ting Huang**[1*]
[1]National Taiwan University of Science and Technology, Taiwan
{ythuang, shao455268}@mail.ntust.edu.tw
[2]Institute of Information Science, Academia Sinica, Taiwan
kinyiu@iis.sinica.edu.tw
[3]National Central University, Taiwan
{pokaichen, a112522133}@g.ncu.edu.tw, jcw@csie.ncu.edu.tw

Abstract

Automated Essay Scoring (AES) plays a crucial role in language assessment. In particular, cross-prompt essay trait scoring provides learners with valuable feedback to improve their writing skills. However, due to the scarcity of prompts, most existing methods overlook critical information, such as content from prompts or essays, resulting in incomplete assessment perspectives. In this paper, we propose a robust AES framework, the Mixture of Ordered Scoring Experts (MOOSE), which integrates information from both prompts and essays. MOOSE employs three specialized experts to evaluate (1) the overall quality of an essay, (2) the relative quality across multiple essays, and (3) the relevance between an essay and its prompt. MOOSE introduces the ordered aggregation of assessment results from these experts along with effective feature learning techniques. Experimental results demonstrate that MOOSE achieves exceptionally stable and state-of-the-art performance in both cross-prompt scoring and multi-trait scoring on the ASAP++ dataset. The source code is released at https://github.com/antslabtw/MOOSE-AES.

1 Introduction

Language education is essential in today's globalized world, facilitating cross-cultural communication and interaction. Automated Essay Scoring (AES) has gained significant attention for its ability to provide rapid, objective, and scalable assessments of written responses. AES rapidly provides objective writing assessments has gotten significant attention. Earlier AES methods adopt a prompt-specific training scheme (Taghipour and Ng, 2016; Dong and Zhang, 2016; Yang et al., 2020; Wang et al., 2022), which often limits their performance to prompts seen during training. When encountering unseen topics, these systems struggle to rate essays in alignment with an appropriate rubric. As a result, developing cross-prompt AES models (Jin et al., 2018; Li et al., 2020; Ridley et al., 2020) that can adapt to unseen topics is essential for improving the generalizability of AES.

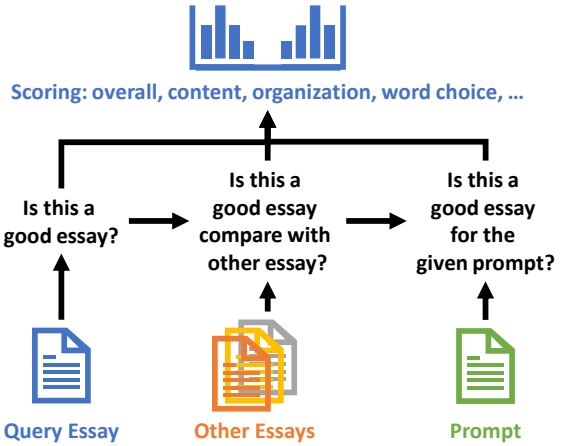

Figure 1: Concept of Ordered Scoring Experts (OSE).

To enhance the accuracy of the AES models, researchers have paid their attention on developing deep learning-based approaches (Jin et al., 2018; Li et al., 2020; Ridley et al., 2020; Mizumoto and Eguchi, 2023; Mansour et al., 2024; Lee et al., 2024; Stahl et al., 2024; Do et al., 2024). However, most of these studies generate only a holistic score, rather than providing detailed feedback to meet educational needs. Teachers and students require assessments that evaluate multiple facets of writing, such as content, organization, grammar, and vocabulary, to support pedagogical objectives and enhance learning outcomes. Thus, recent work on multi-trait scoring (Ridley et al., 2021; Chen and Li, 2023; Do et al., 2023; Xu et al., 2025) has gained more attention for providing more comprehensive writing feedback. Building on this foundation, our research focuses on multi-trait AES to ensure robust performance across prompts.

It is challenging to train a generalized cross-prompt essay trait scoring model with the limited number of available prompts. For example, Ridley et al. (2020) considers only essays as input to the

Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 18071–18084

model, narrowing the task to assessing writing style or surface-level quality. Although work like ProTACT (Do et al., 2023) incorporate essay-prompt correlation features, they rely solely on syntactic features to represent essays. These existing methods overlook critical information, such as the content of prompts or essays, leading to incomplete assessment perspectives.

More recently, Large Language Models (LLMs) have been introduced to address various AES tasks (Mizumoto and Eguchi, 2023; Lee et al., 2024; Stahl et al., 2024; Chu et al., 2025). For instance, Do et al. (2024) proposed the T5 (Do et al., 2024)-based ArTS for multi-trait scoring, while Mansour et al. (2024) explored the performance of ChatGPT and Llama-2 (Touvron et al., 2023) for essay scoring and feedback generation. Additionally, Xu et al. (2025) built EPCTS on top of Qianwen (Bai et al., 2023), achieving promising cross-prompt essay trait scoring results.

In this work, we draw on the concepts of LLMs but develop a robust cross-prompt essay trait scoring system without relying on LLMs. First, we establish a simple yet strong baseline, Multi-cunk BERT with Trait Attention (MBTA), by leveraging content features, and various linguistic features. We then propose the Ordered Scorer Experts (OSE), as shown in Figure 1, which models the evaluation process of professional human raters when scoring essays. Additionally, OSE addresses the inherent scarcity of prompts, a key factor limiting the robustness of AES on unseen prompts. To maximize the utility of the limited available prompts, we reformulate the training of the scoring module from learning how to score to learning how to select scoring cues. This change allows the model to focus on learning diverse scoring cues, which reducing overfitting to previously seen data. Building on these strategies, a Mixture of Experts (MoE) approach is introduced to dynamically select relevant scoring cues based on prompts, culminating in our AES framework, MOOSE. Our work makes the following key contributions:

- A strong baseline for cross-prompt essay trait scoring called Multi-chunk BERT with Trait Attention (MBTA) is developed.
- We reformulating the AES training objective from direct scoring to scoring cue retrieval, this change enhances the robustness of cross-prompt AES model.
- We decompose the scoring mechanism into three aspects, which are (1) essay-intrinsic quality, (2) cross-essay comparison, and (3) essay–prompt relevance, to strengthen the reasoning in cross-prompt essay trait scoring.
- The proposed MOOSE framework achieves state-of-the-art performance in cross-prompt AES with trait scoring.

2 Related Work

2.1 Cross-Prompt AES

In the early research of AES, they focused on prompt-specific settings (Taghipour and Ng, 2016; Dong and Zhang, 2016; Dong et al., 2017; Yang et al., 2020; Wang et al., 2022; Cao et al., 2020). These settings evaluate the performance of trained models on seen prompt in training stage. However, in the real world applications, it is impossible to take all of the prompts to train a model. Recently, more studies (Jin et al., 2018; Li et al., 2020; Ridley et al., 2020; Zhang et al., 2025) focus on cross-prompt setting which is more close to the real world settings. Cross-prompt setting trains the model on multiple prompts and uses unseen prompts for testing. Tackling on this setting, researchers do their efforts on designing unbiased feature extractor and devloping methods to find out the real adherence between prompt and essay. For example, PAES (Ridley et al., 2020) applies hierarchical-CNN-LSTM with Part-Of-Speech (POS) embedding and linguistic features to decrease the semantic bias from different prompts. PANN (Jiang et al., 2023) disentangles quality and content information in essay features for better finding out the fake correlation between essay and prompt adherence features.

With advances in LLMs, some research (Mizumoto and Eguchi, 2023; Mansour et al., 2024; Lee et al., 2024; Stahl et al., 2024) started to apply LLMs for developing AES models. Do et al. (2024) proposed ArTS in auto-regressive manner for generating multi-trait scores. Lee et al. (2024) introduced multi-trait decomposition which generate specific prompt trait rubric by LLMs. Stahl et al. (2024) designed various kinds of prompt and instruction, and applying in-context learning method to enhance few-shot learning AES performance.

Most of the above cross-prompt studies are struggling with the imbalance performance across different prompts. In this paper, we address this research problem.

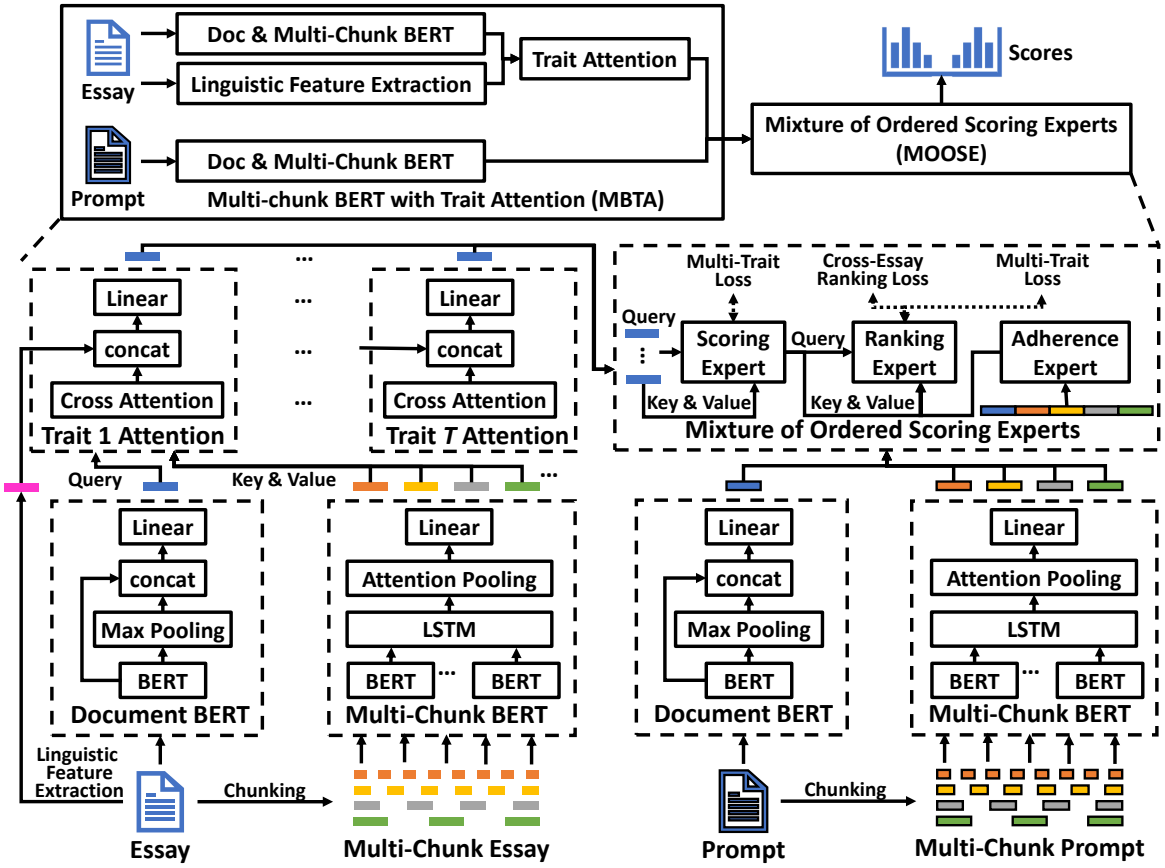

Figure 2: System overview of our proposed AES system for cross-prompt essay trait scoring.

2.2 Cross-Prompt AES with Trait Scoring

Research on cross-prompt AES trait scoring (Ridley et al., 2021; Chen and Li, 2023; Do et al., 2023; Sun et al., 2024; Xu et al., 2025) has gained significant attention, as it emphasizes the generalization ability of AES models to evaluate multiple writing traits on unseen prompts. For example, CTS (Ridley et al., 2021) employs a trait-attention mechanism to exchange cross-trait information and enhances the scoring performance of each trait. PMAES (Chen and Li, 2023) adopts a mapping learning strategy that aligns source and target prompts within a unified feature space to achieve stable cross-prompt scoring. It further tackles multi-trait scoring through a hierarchical encoder coupled with trait-specific dense layers and cross-trait correlation loss. ProTACT (Do et al., 2023) emphasizes prompt adherence by leveraging essay–prompt attention and incorporating a topic-coherence feature, along with a trait-similarity loss to promote integration and consistency in multi-trait scoring. However, these studies often discard crucial information from either the prompt or the essay, such as essay content, to prevent the trained model from overfitting to seen prompts.

Recent work proposed by Li and Ng (2024) demonstrates that performing feature selection for each prompt could further enhance scoring performance. While EPCTS (Xu et al., 2025) leverages LLMs to capture the nuanced relationship between the prompt and the essay via semantic segmentation and similarity computation, thereby effectively evaluating prompt relevance.

Inspired on these impressive works, our approach establishes a strong baseline that includes complete information of the prompt and essay. We then design a robust AES framework build upon it to imitate scoring process of professional human raters, which enable the model dynamically retrieve the scoring cues in an ordered point of view.

3 Methods

3.1 System Overview

Figure 2 illustrates our proposed AES system for cross-prompt essay trait scoring. The proposed AES system contains three main steps. First, we extract content features of the essay and prompt, with multi-granularity, via document BERT and multi-chunk BERT models (Wang et al., 2022). Additionally, various linguistic features of the essay are extracted as side information for trait scoring.

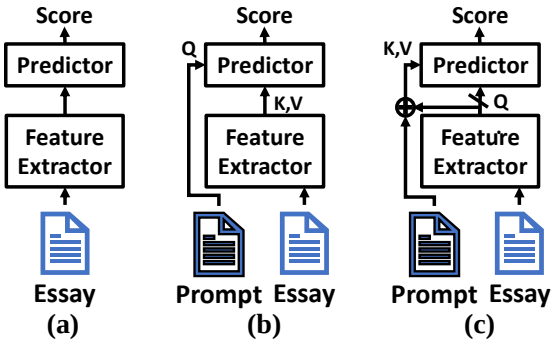

Figure 3: Mainstream and our AES approaches.

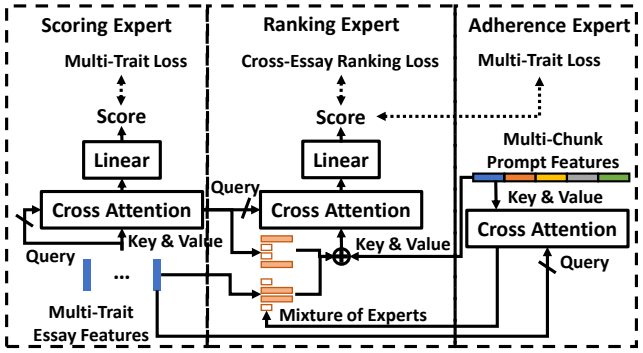

Figure 4: Mixture of Ordered Scoring Experts (MOOSE).

Next, we adopt the architecture of trait attention mechanism proposed in ProTACT (Do et al., 2023) to learn trait-specific representations. Different from the ProTACT, we use the document BERT feature as the query and the multi-chunk BERT features as both key and value in the cross-attention layer to learn non-prompt specific representation of the essay. The attended representation is then concatenated with the non-prompt specific linguistic features, and fed into a linear layer to obtain trait-specific representations for each essay.

Finally, Both of the trait-specific essay representations and the prompt features are fed into our proposed MOOSE framework to predict the essay's trait scores. Notably, our approach differs from previous state-of-the-art methods (Do et al., 2023; Xu et al., 2025) that treat the prompt as the query. Instead, MOOSE uses the essay as the query to learn more robust representations and capture scoring cues with rich diversity. Existing SoTAs design their solutions from the prompt's perspective, aiming to determine whether a given essay is likely to receive a high score under the given prompt. In contrast, MOOSE transfers the scoring process into a scoring cue retrieval problem from the essay, and prompt features are treated as one kind of cue for selecting prompt-relation scoring cues. The following sections will detail our proposed problem transfer strategy and the MOOSE framework.

3.2 Essay as Queries

In the Transformer architecture (Vaswani et al., 2017), an attention score is defined as: $a_i = softmax(q_i K^T)V$. First, the relevance score for each query $q_i \in Q$ is computed against all keys $k_j \in K$. Next, the softmax function gives relative importance to the keys K based on their alignment with the given query q. Finally, the importance is used to estimate the distribution of the query q_i over the values V, thus producing an attention-informed representation of the query.

When aligning the Transformer process described above to the AES context, if prompt p is used as the query and essay e is used as the key and value, it is equivalent to reconstructing the prompt information with the components of the essay for scoring. This will only retain the parts of essay e's information that are highly correlated with prompt p, which will lead to biased scoring results. We proposed to use essay e as the query and concatenates prompt p and essay e as the key and value, as shown in Figure 3 (c). The estimated joint distribution of query over values will include both the degree of prompt adherence and the degree of essay for the content and organization, which have a more robust representation of the essay scoring task. Since the scoring of AES traits such as word choice, grammar, organization, etc., primarily relies on the content of the essay itself, using the essay as the query results in more robust performance.

3.3 From Scoring to Scoring Cue Retrieval

When training a cross-prompt AES model, the diversity of available prompts is often severely limited. Focusing exclusively on feature (query) learning can lead to overfitting on seen prompts. Therefore, we propose that the model should concentrate on learning comprehensive scoring cues (value). By expanding and refining these scoring cues, the model is more likely to remain robust when encountering unseen prompts. Specifically, in the score prediction module, we apply a stop-gradient operation on the query, allowing gradients to update only the key and value. The fixed query then serves the function of retrieving relevant scoring cues.

Likewise, in the multi-layer decoder, we aim to prevent the model overfitting to the seen prompts. We achieve this by reusing the same key and value across different decoder layers. Reuse of the key and value making the decoding process become a progressive feature selection process. Figure 3 illustrates mainstream approaches and our approach:

- Figure 3 (a): Early methods (Taghipour and Ng, 2016; Dong and Zhang, 2016; Yang et al., 2020; Cao et al., 2020; Ridley et al., 2020, 2021) often scored the essay solely based on its content of essay.
- Figure 3 (b): Since ProTACT (Do et al., 2023), subsequent methods use the prompt as the principal role for scoring. Currently, this approach needs LLMs to generalize the relationship between the prompt and the essay for stable results across diverse prompts, such as EPCTS (Xu et al., 2025).
- Figure 3 (c): We propose to treat the essay as the principal role and reformulating the scoring process as a scoring cue retrieval, leading to a robust cross-prompt AES approach.

3.4 Mixture of Ordered Scoring Experts

In our MOOSE framework, we aim for a scoring mechanism that imitates the reasoning process of human experts. We design three specialized "experts" to tackle the following aspects of scoring:

The inherent writing quality of the essay. To measure the inherent writing quality of the essay, we adopt the loss function proposed by ProTACT (Do et al., 2023), which considers both the final trait scores and the reasoning among different traits using Pearson correlation coefficient r. If the correlation exceeds the threshold ϵ, the model is encouraged to maintain similarity between the predicted trait scores. The trait similarity loss is shown in Equation (1):

$$\mathcal{L}_{\text{ts}}(y, \hat{y}) = \frac{1}{c} \sum_{j=2}^{M} \sum_{k=j+1}^{M} \text{ts}(\hat{\mathbf{y}}_j, \hat{\mathbf{y}}_k, \mathbf{y}_j, \mathbf{y}_k), \quad (1)$$

$$\text{ts} = \begin{cases} 1 - \cos(\hat{\mathbf{y}}_j, \hat{\mathbf{y}}_k), & \text{if } r(\mathbf{y}_j, \mathbf{y}_k) \geq \epsilon \\ 0, & \text{otherwise} \end{cases}, \quad (2)$$

where c is the number of calculated ts that is not 0, M is the number of traits, j and k are indexes of the selected trait. Note that the trait Overall ($j = 1$) is excluded, as its score has relatively low correlations than other traits, so the index of j is from 2.

Relative quality across multiple essays. To compare the relative quality of multiple essays, we use the pairwise ranking loss (Equation 3). For each essay pair (a, b) in a training batch, we compute the predicted scores $\hat{y}_a$ and $\hat{y}_b$ and their difference $\Delta\hat{y} = \hat{y}_a - \hat{y}_b$. If the ground-truth scores satisfy $y_a > y_b$, we set $\delta = 1$ (otherwise, $\delta = 0$). The ranking loss penalizes incorrect orderings and encourages a larger $\Delta\hat{y}$ for correct rankings. Additionally, $\log(1 + e^{-|\Delta\hat{y}|})$ smooths the penalty, preventing gradient vanishing. Notably, only scores associated with the same trait are paired and used to compute the ranking loss. When different prompts share the same trait, the scores for that trait are also paired and included in the ranking loss calculation.

$$\mathcal{L}_{\text{rank}} = \delta \cdot \big(\max(0, -\Delta\hat{y}) + \log(1 + e^{-|\Delta\hat{y}|}) \big). \quad (3)$$

Prompt adherence of the essay. We measure the prompt adherence by estimate the probability of essay on joint distribution of multi-trait essay representations and prompt features. To achieve this purpose, the multi-chunk BERT features extracted from the prompt are concatenated with the key and value of cross attention layer. The model then examines the degree to which the prompt value is attended, identifying potential off-topic situations.

By leveraging these three specialized experts and integrating their insights, MOOSE imitates the multiple perspectives scoring process of human raters, thereby enhancing both the accuracy and robustness of cross-prompt essay trait scoring.

Figure 4 illustrates the architecture of MOOSE, which is built upon a dual-layer cross-attention decoder and incorporates three sequential experts – scoring expert, ranking expert, and adherence expert, which forming an OSE. To transfer the training objective into a process of scoring cue retrieval, the stop-gradient operation is applied on the queries in the decoder. Then, the scoring expert takes only multi-trait essay features as input to learn essay inherent scoring cues. Following, the input query of ranking expert is dynamic selected by a Mixture of Expert (MoE) module, which determines whether to reuse the existing key–value pairs. The gating function of the MoE is calculted by attended feature of a cross attention layer which takes multi-trait essay representation as query and multi-chunk prompt feature as key and value. Finally, the adherence expert passing the multi-chunk prompt features to the ranking expert, making ranking expert to estimate whether the essay is addressed to the given prompt or not. By modeling the step-wise reasoning of human experts, MOOSE aims to achieve more accurate and robust essay trait scoring.

Next, we describe the implementation details of the Mixture-of-Experts (MoE) architecture within MOOSE. The general formula of MoE is given by:

Prompt	Essay Type	Content	Organization	Word Choice	Sentence Fluency	Conventions	Prompt Adherence	Language	Narrativity
1	Argumentative	✓	✓	✓	✓	✓			
2	Argumentative	✓	✓	✓	✓	✓			
3	Response (Source-Dependent)	✓					✓	✓	✓
4	Response (Source-Dependent)	✓					✓	✓	✓
5	Response (Source-Dependent)	✓					✓	✓	✓
6	Response (Source-Dependent)	✓					✓	✓	✓
7	Narrative	✓	✓			✓			
8	Narrative	✓	✓	✓	✓	✓			

Table 1: Essay type and trait information of the ASAP++ dataset. (Mathias and Bhattacharyya, 2018)

$$y = \sum_{i=1}^{n} G(x)_i \cdot E_i(x), \tag{4}$$

where n is the number of experts, G is the gating function, and E represents the expert network. In MOOSE, the MoE consists of two experts, both implemented as linear layers, and the gating function is a sigmoid gating function σ. To enhance the adaptability of the model, we replace the inputs of the gating function with the cross-attention outputs between the essay features and prompt features, allowing MOOSE to dynamically select the appropriate expert based on the essay-prompt relationship for score prediction. The MoE formula in the MOOSE could be rewritten as Equation (5):

$$\begin{aligned} y = {} & \sigma(CA(SG(F_{e1}), F_p)) \cdot E_1(F_{e1}) \\ & + (1 - \sigma(CA(SG(F_{e1}), F_p))) \cdot E_2(F_{e2}). \end{aligned} \tag{5}$$

where CA represents cross-attention, SG denotes stop gradient, and F_e, F_p correspond to essay and prompt features, respectively.

4 Experiments

4.1 Experimental Setup

In this study, we evaluate our model using the publicly available ASAP (Hamner et al., 2012) and ASAP++ (Mathias and Bhattacharyya, 2018) datasets. The original ASAP corpus comprises approximately 13,000 English essays across eight prompts, each assigned a holistic score. In contrast, ASAP++ augments these essay sets by providing additional trait-level scores. The trait information of ASAP++ dataset is shown in Table 1, and more information of the datasets are provided in Appendix B. To maintain the consistent comparison with previous works, the experimental settings of (Do et al., 2023) is adopted. The Quadratic Weighted Kappa (QWK) is a common metric in AES, which measures the agreement between our predicted scores and the ground-truth labels. In cross-prompt evaluation, essays from one prompt set serve as the test set, while essays from the remaining prompts are used for training.

The training details of MOOSE are as follows. All hyperparameters of MBTA follow the settings from the referenced papers. Multi-chunk BERT segments the input text into four fixed chunk sizes (10, 30, 90, 130), enabling the model to learn text features at different granularities. The LSTM dimension is set to 576, matching the BERT-base configuration. The feature dimension of Trait Attention is set to 256, which is also used for MOOSE. For the loss functions, the scoring expert and ranking expert use following weighted losses:

- Scoring expert: $0.7\mathcal{L}_{\text{mse}} + 0.3\mathcal{L}_{\text{ts}}$.
- Ranking expert: $0.5\mathcal{L}_{\text{mse}} + 0.2\mathcal{L}_{\text{ts}} + 0.3\mathcal{L}_{\text{rank}}$.

The overall loss is simply the summation of the scoring expert loss and ranking expert loss. More details about training hyper-parameters and used linguistic features are provided in Appendix A.

4.2 Comparison with State-of-The-Arts

We compare our proposed OSE and MOOSE with State-of-The-Arts (SoTAs) in cross-prompt essay trait scoring. Specifically, methods that rely solely on essay representations include PAES (Ridley et al., 2020), PMAES (Chen and Li, 2023), CTS (Ridley et al., 2021), and RDCTS (Sun et al., 2024). Methods that incorporate part-of-speech embeddings and linguistic features but do not leverage content information of essay are PAES (Ridley et al., 2020), PMAES (Chen and Li, 2023), CTS (Ridley et al., 2021), and ProTACT (Do et al., 2023). Additionally, EPCTS (Xu et al., 2025) is the method which use content features of both essay and prompt, it also enhances AES robustness by integrating LLMs (Bai et al., 2023). The comparison of the prompt-based and trait-specific evaluations are presented in Table 2 and Table 3, respectively.

Table 2 shows that proposed methods achieves the highest average performance, and gets best performance on 5 out of 8 prompts. MOOSE demonstrates the highest stability across all prompts with the lowest standard deviation (STD). We observe that except for EPCTS (Xu et al., 2025) which

Model	Prompt 1	Prompt 2	Prompt 3	Prompt 4	Prompt 5	Prompt 6	Prompt 7	Prompt 8	AVG	STD
PAES (Ridley et al., 2020)	.605	.522	.575	.606	.634	.545	.356	.447	.536	.088
PMAES (Chen and Li, 2023)	.656	.553	.598	.606	.626	.572	.386	.530	.566	.078
CTS (Ridley et al., 2021)	.623	.540	.592	.623	.613	.548	.384	.504	.553	.076
RDCTS (Sun et al., 2024)	.651	.553	.608	.623	.651	.580	.375	.529	.571	.085
ProTACT (Do et al., 2023)	.647	.587	.623	.632	.674	.584	.446	.541	.592	.067
EPCTS (Xu et al., 2025)	.659	.609	.619	**.686**	.671	**.629**	.555	**.630**	.632	.038
OSE (Ours)	.679	.612	**.660**	.660	.686	.596	.581	.627	.638	.037
MOOSE (Ours)	**.685**	**.613**	.657	.652	**.700**	.615	**.592**	.621	**.642**	**.036**

Table 2: Comparison of average QWK for each prompt on the ASAP++ dataset, **bold font** indicates best performance.

Model	Overall	Content	Organization	WC	SF	Convention	PA	Language	Narrativity	AVG	STD
PAES (Ridley et al., 2020)	.657	.539	.414	.531	.536	.367	.570	.531	.605	.527	.075
PMAES (Chen and Li, 2023)	.671	.567	.481	.584	.582	.421	.584	.545	.614	.561	.060
CTS (Ridley et al., 2021)	.670	.555	.458	.557	.545	.412	.565	.536	.608	.586	.062
RDCTS (Sun et al., 2024)	.673	.561	.480	.591	.576	.426	.609	.560	.634	.568	.065
ProTACT (Do et al., 2023)	.674	.596	.518	.599	.585	.450	.619	.596	.639	.586	.058
EPCTS (Xu et al., 2025)	**.728**	.630	.606	.614	.617	.525	.630	.613	.647	.623	.035
OSE (Ours)	.677	.643	.639	**.641**	.635	.575	.637	.610	.649	.634	.023
MOOSE (Ours)	.650	**.651**	**.652**	.634	**.643**	**.604**	**.649**	**.624**	**.665**	**.641**	**.018**

Table 3: Comparison of average QWK for each trait on the ASAP++ dataset, **bold font** indicates best performance.

leverages LLMs, all other models exhibit high sensitivity to the type of the prompt. In particular, for prompts 7 and 8, the open-ended nature of narrative essays leads to a significant drop in performance

Table 3 presents the performance of the score of traits in different models. Our proposed model outperforms all baselines in all of eight traits. Notably, previous methods struggle with the convention trait.Among previous SoTAs, except for EPCTS (Xu et al., 2025) which benefits from LLMs, all models exhibit weak performance in content and organization traits. Furthermore, POS embedding-based models, such as PAES and CTS, show inferior performance in sentence fluency. EPCTS achieves the highest QWK in overall score by leveraging prompt-relevant features extracted by LLMs. Overall, the proposed MOOSE demonstrating its highly consistent scoring ability across different traits via imitating scoring process of human expert, it achieves at least half of standard deviation (STD) than previous SoTAs.

Based on above observations, we will conduct further experiments in Section 4.4 and Section 4.5 to analyze the proposed components contributing to the observed performance improvements.

4.3 Ablation Studies

Table 4 presents the results of our ablation studies, which evaluate the impact of each proposed component. Our developed baseline model, Multi-chunk BERT with Trait Attention (MBTA), processes essays as input and employs a single-layer transformer decoder for trait scoring. To systematically assess the contribution of different components in the proposed MOOSE, we conduct ablation experiments on six key components: Query Detach (QD), Dual-layer Head (DH), RanKing loss (RK), reuse of Key and Value (KV), Prompt feature integration in the Prediction layer (PP), and Ordered Scoring Experts (OSE). Each model variant was trained for 14 epochs with a learning rate of $1e^{-5}$. Finally, Hyper-parameter Alignment to ProTACT (HA) and Mixture of Experts (MoE) are applied on the ORE model and hyper-parameter aligned ORE model respectively as the final models. The HA involves increasing the learning rate to $1e^{-4}$ and extending saving checkpoints to 50, allowing for analysis on a consistent basis for making comparison with other SoTAs. The architectures of ablated model variants are shown in Appendix C.1.

Model	P1	P2	P3	P4	P5	P6	P7	P8	AVG	IMP
A:MBTA	.639	.593	.603	.604	.657	.555	.469	.589	.589	=
B:A+QD	.645	.616	.613	.617	.648	.553	.477	.600	.595	+.006
C:B+DH	.657	.606	.612	.629	.672	.538	.473	.595	.598	+.009
D:C+RK	.679	.612	.646	.655	.688	.560	.549	.461	.606	+.017
E:D+KV	.668	.604	.650	.658	.679	.570	.559	.495	.610	+.021
F:D+PP	.630	.583	.636	.656	.683	.575	.579	.514	.607	+.018
G:F+OSE	.675	**.617**	.654	**.668**	.686	.600	.528	.560	.624	+.034
H:G+HA	.679	.612	**.660**	.660	.686	.596	.581	**.627**	.638	+.048
I:H+MoE	**.685**	.613	.657	.652	**.700**	**.615**	**.592**	.621	**.642**	+.052

Table 4: Ablation studies of proposed components.

We first introduce QD, which reformulates the learning objective into a scoring cue retrieval task. This modification leads to a general performance improvement across most prompts. Next, we deepen the transformer decoder by adding an additional layer (DH), resulting in a slight performance

Model	P1	P2	P3	P4	P5	P6	P7	P8
prompt as query	**.677**	.611	.643	.664	.646	.576	.480	.427
essay as query	.675	**.617**	**.654**	**.668**	**.686**	**.600**	**.528**	**.560**

Table 5: Analysis of query type on each prompt.

Model	P1	P2	P3	P4	P5	P6	P7	P8
scoring	.639	.593	.603	.604	**.657**	**.555**	.469	.594
cue retrieval	**.645**	**.616**	**.613**	**.617**	.648	.553	**.477**	**.600**

Table 6: Analysis of learning goal on each prompt.

Model	P1	P2	P3	P4	P5	P6	P7	P8
scoring experts	.648	.608	.592	.638	.651	.535	.484	**.616**
ranking experts	.630	.583	.636	.656	.683	.575	**.579**	.514
ordered experts	**.675**	**.617**	**.654**	**.668**	**.686**	**.600**	.528	.560

Table 7: Analysis of expert type on each prompt.

Model	Overall	T1	T2	T3	T4	T5	T6	T7	T8
prompt as query	.631	.607	.547	.575	.552	.478	.628	.593	.645
essay as query	**.678**	**.627**	**.603**	**.634**	**.601**	**.522**	**.638**	**.610**	**.658**

Table 8: Analysis of query type on each trait.

Model	Overall	T1	T2	T3	T4	T5	T6	T7	T8
rank→score	.633	.595	**.581**	**.620**	**.619**	**.525**	.592	.588	.611
score→rank	**.649**	**.605**	.568	.577	.553	.506	**.622**	**.605**	**.646**

Table 9: Analysis of experts' order on each trait.

Model	Overall	T1	T2	T3	T4	T5	T6	T7	T8
scoring experts	.632	.603	.571	.628	.612	.509	.608	.591	.628
ranking experts	.666	.603	.569	.585	.567	.512	.613	.603	.628
ordered experts	**.678**	**.627**	**.603**	**.634**	**.601**	**.522**	**.638**	**.610**	**.658**

Table 10: Analysis of expert type on each trait.

gain. The incorporation of RK (ranking loss) further enhances scoring accuracy, however, we observe a performance drop on Prompt 8. We hypothesize that this decline is due to the open-ended nature of Prompt 8, where responses are based on personal experiences, making direct quality ranking inherently challenging. This observation underscores the importance of selecting appropriate features for different prompt types. To address this, we integrate KV to fine-tune the second-layer decoder as a feature selector, effectively reducing scoring sensitivity across prompts.

Furthermore, we investigate the role of prompt features in our model. While the introduction of PP does not improve overall performance, it enhances scoring consistency across prompts. This suggests that prompt adherence is crucial for reliable scoring. The introduction of OSE yields the most substantial performance gain, confirming that structuring the scoring process into specialized components - scoring expert, ranking expert, and adherence expert effectively improves scoring accuracy. At th final, we add MoE on OSE to construct MOOSE. Both ORE and MOOSE show excellent results in cross-prompt essay trait scoring. They achieve an average QWK of 0.638 and 0.642, respectively, which set the new SoTA on the ASAP++ dataset.

4.4 Analysis of Cross-Prompt Scoring

In this section, we analyze and discuss three key aspects of cross-prompt scoring, which are query selection, scoring cue retrieval, and expert decoders.

Query Type Table 5 compares using an essay or a prompt as a query. Results show that essay queries consistently improve performance, especially mitigating the underperformance of prompt queries on narrative prompts (P7-P8). We infer that, when the prompt serves as the query while the essay is treated as the value, the model essentially learns the distribution of prompt-related features within the essay. This approach tends to excel when the prompt strongly dictates the writing direction, but suffers for more open-ended topics. In such cases, essays under the same prompt may exhibit large intra-class variation, which cannot be effectively learned from the limited prompt distribution alone. Because there are significantly more essays than prompts, using the essay as the query enables the model to learn a more generalized distribution. Consequently, prompt-based queries achieve their best performance on highly topic-focused argumentative prompts (P1–P2), show only moderate effectiveness on partially constrained response prompts (P3–P6), and fail to capture the full spectrum for more open-ended narrative prompts (P7–P8).

Scoring Cue Retrieval Table 6 examines Query Detach (QD), which reformulates learning as a scoring cue retrieval task. QD yields slight but consistent improvements, enhancing robustness in cross-prompt AES settings. This confirms that making the model learning to select scoring cues lead the model more robust when the prompts are scarce.

Expert Decoders Table 7 presents results of different type of expert decoders. OSE outperforms other combinations and shows strong consistency. Ranking experts perform poorly on argumentative prompts (P1-P2) but excel when reference materials are available (P3-P6). Narrative questions are closely related to personal experiences according to the subject matter, so scoring experts and ranking experts have their own strengths in different topics.

4.5 Analysis of Trait Scoring

In this section, we analyze and discuss serveral key different between our proposed method and existing method in cross-prompt trait scoring.

Query Type Table 8 compare "prompt as query" with "essay as query". The result shows that "essay as query" increase scoring ability in all of the traits. The trait with the smallest difference is prompt adherence. This suggests that using the prompt as a query relies too heavily on the relationship between the prompt and the essay.

Order of Experts Table 9 shows the results of swapping the order of experts. It is obvious that the scoring of the traits important for response prompts (PA, Lang, and Nar) perform better when performing ranking experts at last. The scoring of the traits that are important for argumentative prompts (Org, WC, SF, Conv) perform better when performing scoring experts at last. This is consistent with the conclusions in cross-prompt scoring analysis.

Expert Decoders Finally, Table 10 summarizes how different expert combinations perform across various decoders. The proposed ordered scoring experts achieve the best results for all traits, confirming that imitates the human scoring process is a promising strategy for AES. Notably, the ranking expert exhibits weaker performance on WC and SF, primarily due to prompt 8. This underscores an inherent issue with the ranking expert approach: when the prompt type in the training data differs substantially from that in the test data, the relative-quality features it learns do not generalize. Taken together with the ranking expert's less favorable performance on argumentative prompts, these findings highlight the need for additional research on loss functions that more effectively capture relative essay quality in a broader range of scenarios.

4.6 Visualization of Mixture of Experts

Figure 5 visualizes the outputs of the gating functions when using prompt features to guide gating functions. The darker the color, the higher the probability of selecting tokens refined by scoring experts as input for ranking experts. When prompt feature is used as guidence, the choice of experts is highly correlated with the prompt type. For Narative prompts (P7–P8), the probability of selecting refined tokens is the highest, followed by Argumentative prompts (P1–P2), while Response (Source-Dependent) prompts (P3–P6) almost completely

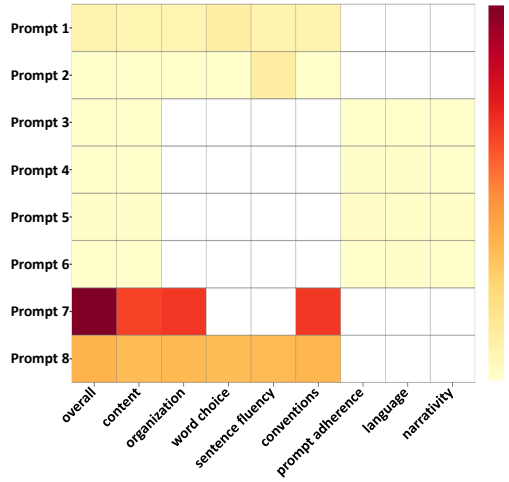

Figure 5: Outputs of the gating functions when using prompt feature as keys and values.

select original tokens. The result is consistent with previous observations. Narative prompts are more open-ended and require high-level semantic features to determine the relevance between essays and prompts. Argumentative prompts have positive and negative opinions, so sometimes high-level semantic features are needed to assist in determining the relevance between essays and prompts. For Response (Source-Dependent) prompts, since the reference source of the essay is specified, using the essay itself can highly reflect the relevance between the essay and the prompt.

5 Conclusions

This paper proposes an AES framework – MOOSE, that imitates the scoring process of human experts for cross-prompt essay trait scoring. MOOSE aims to learn robust content features that generalize to unseen prompts while predicting corresponding trait scores to guide language learning. To imitate the human scoring process, we design three types of experts: a scoring expert to assess the inherent quality of the essay, a ranking expert to compare relative quality across different essays, and an adherence expert to measure the relation between the essay-prompt pair. These experts are integrated into an ordered mixture of experts.

To learn robust features, we introduce essay query, query detach, and key value reuse techniques, which enable the model to capture fine-grained features and focus on retrieving useful scoring cues. Thereby reducing overfitting to the seen prompts. Consequently, MOOSE achieves impressive performance on the ASAP++ cross-prompt essay trait scoring task, surpassing current SoTA approaches built on LLMs.

Limitations

Our method enhances AES robustness in cross-prompt scoring and multi-trait scoring, yet several limitations persist. First, expert performance varies across prompts, underscoring the need for a dynamic selection mechanism that adapts scoring strategies to prompt characteristics. Second, expert combination order significantly impacts trait-specific scoring, highlighting the importance of integrating trait-aware expert training.

Additionally, prompt types influence trait scoring, yet our approach prioritizes cross-prompt robustness while overlooking prompt-type-specific scoring nuances. Incorporating these factors into loss functions remains an open challenge.

Lastly, our method struggles to capture high-level semantic relationships. While LLMs do not necessarily excel in multi-trait scoring, they outperform in holistic assessment, suggesting that human raters consider hidden factors beyond predefined traits. Enabling AES with self-reasoning capabilities could bridge this gap by uncovering deeper semantic structures.

Acknowledgments

The authors acknowledge the support from the National Science and Technology Council (NSTC) in Taiwan under grant number NSTC 112-2222-E-011-011-MY2. The authors would also like to express their appreciation for the support from the Academia Sinica in Taiwan under grand challenge seed program – SiliconMind. Furthermore, the authors extend their gratitude to the National Center for Highperformance Computing (NCHC) for providing computational and storage resources.

References

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Yue Cao, Hanqi Jin, Xiaojun Wan, and Zhiwei Yu. 2020. Domain-adaptive neural automated essay scoring. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieva*, pages 1011–1020.

Xinlei Chen and Kaiming He. 2021. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15750–15758.

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

Alexis Chevalier, Alexander Wettig, Anirudh Ajith, and Danqi Chen. 2023. Adapting language models to compress contexts. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3829–3846.

SeongYeub Chu, JongWoo 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 *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*.

Gilad Deutch, Nadav Magar, Tomer Natan, and Guy Dar. 2024. In-context learning and gradient descent revisited. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers) (NAACL)*, pages 1017–1028.

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.

Heejin Do, Yunsu Kim, and Gary Geunbae Lee. 2024. Autoregressive score generation for multi-trait essay scoring. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 1659–1666.

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

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

Ben Hamner, Jaison Morgan, lynnvandev, Mark Shermis, and Tom Vander Ark. 2012. The Hewlett foundation: Automated essay scoring. `https://kaggle.com/competitions/asap-aes`. Kaggle.

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 Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12456–12470.

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 Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1088–1097.

Sanwoo Lee, Yida Cai, Desong Meng, Ziyang Wang, and Yunfang Wu. 2024. Unleashing large language models' proficiency in zero-shot essay scoring. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 181–198.

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.

Xia Li, Minping Chen, and Jian-Yun Nie. 2020. SEDNN: Shared and enhanced deep neural network model for cross-prompt automated essay scoring. *Knowledge-Based Systems*, 210:106491.

Watheq Mansour, Salam Albatarni, Sohaila Eltanbouly, and Tamer Elsayed. 2024. Can large language models automatically score proficiency of written essays? In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 2777–2786.

Sandeep Mathias and Pushpak Bhattacharyya. 2018. ASAP++: Enriching the ASAP automated essay grading dataset with essay attribute scores. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Yu Meng, Chenyan Xiong, Payal Bajaj, Paul Bennett, Jiawei Han, Xia Song, et al. 2021. Coco-lm: Correcting and contrasting text sequences for language model pretraining. *Advances in Neural Information Processing Systems (NeurIPS)*, 34:23102–23114.

Atsushi Mizumoto and Masaki Eguchi. 2023. Exploring the potential of using an AI language model for automated essay scoring. *Research Methods in Applied Linguistics*, 2(2):100050.

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*.

Maja Stahl, Leon Biermann, Andreas Nehring, and Henning Wachsmuth. 2024. Exploring LLM prompting strategies for joint essay scoring and feedback generation. In *Proceedings of the 19th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2024)*, pages 283–298.

Jingbo Sun, Weiming Peng, Tianbao Song, Haitao Liu, Shuqin Zhu, and Jihua Song. 2024. Enhanced cross-prompt trait scoring via syntactic feature fusion and contrastive learning. *The Journal of Supercomputing*, 80(4):5390–5407.

Kaveh Taghipour and Hwee Tou Ng. 2016. A neural approach to automated essay scoring. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1882–1891.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in Neural Information Processing Systems (NeurIPS)*.

Yongjie Wang, Chuan Wang, Ruobing Li, and Hui Lin. 2022. On the use of BERT for automated essay scoring: Joint learning of multi-scale essay representation. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*, pages 3416–3425.

Jiangsong Xu, Jian Liu, Mingwei Lin, Jiayin Lin, Shenbao Yu, Liang Zhao, and Jun Shen. 2025. EPCTS: Enhanced prompt-aware cross-prompt essay trait scoring. *Neurocomputing*, 621:129283.

Ruosong Yang, Jiannong Cao, Zhiyuan Wen, Youzheng Wu, and Xiaodong He. 2020. Enhancing automated essay scoring performance via fine-tuning pre-trained language models with combination of regression and ranking. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1560–1569.

Chunyun Zhang, Jiqin Deng, Xiaolin Dong, Hongyan Zhao, Kailin Liu, and Chaoran Cui. 2025. Pairwise dual-level alignment for cross-prompt automated essay scoring. *Expert Systems with Applications*, 265:125924.

A Training Details

A.1 Hyper-Parameter Settings

Table 11 shows the hyper-parameters setting for training our ablated and final models.

Hyper-Parameter	Ablated Model	Final Model
epoch	14	14
optimizer	AdamW	AdamW
learning rate	$1e^{-5}$	$1e^{-4}$
weight decay	0.0	0.001
batch size	8	8
encoder	BERT-base	BERT-base
chunk size	{10,30,90,130}	{10,30,90,130}
multi-head num	2	2
hidden dim	256	256
checkpoints	14	56
scoring loss	$.7L_{mse}$+$.3L_{ts}$	$.7L_{mse}$+$.3L_{ts}$
ranking loss	$.5L_{mse}$+$.2L_{ts}$+$.3L_{rank}$	$.5L_{mse}$+$.2L_{ts}$+$.3L_{rank}$

Table 11: Hyper-parameter settings.

A.2 List of Linguistic Features

We extract 86 dimension of linguistic features, and the details are shown in Table 12.

Type	Feature (86)
readability grades (9)	Kincaid, ARI, Coleman-Liau, Flesch Reading Ease, Gunning Fog Index, LIX, SMOG Index, RIX, Dale Chall Index.
sentence information (14)	characters per word, syll per word, words per sentence, sentence per paragraph, type token ratio, chracters, syllables, words, wordtypes, sentences, paragraphs, long words, complex words, complex words dc.
word usage (6)	tobeverb, auxverb, conjunction, pronoun, preposition, nominalization.
sentence beginnings (6)	pronoun, interrogative, article, subordination, conjunction, preposition.
part-of-speech (27)	VB, JJR, WP, PRP$, VBN, VBG, IN, CC, JJS, PRP, MD, WRB, RB, VBD, RBR, VBZ, NNP, POS, WDT, DT, CD, NN, TO, JJ, VBP, RP, NMS.
other hand craft (24)	mean word, word variance, mean sentence, sentence variance, essay character length, word count, prep comma, unique word, clause per sentence, mean clause length, nax clause in sentence, spelling error, sentence average depth, average leaf depth, automated readability, linsear write, stop prop, positive sentence prop, nagative sentence prop, overall positivity score, overall negativity score, number of ",", number of ".".

Table 12: List of lingustic features.

B Information of ASAP++ Dataset

B.1 Statistical Information of ASAP++

The statistics of ASAP++ is shown in Table 13.

Prompt	Essay Type	Essay Num	Avg Len	Max Len
1	Argumentative	1785	350	960
2	Argumentative	1800	350	1173
3	Response	1726	150	415
4	Response	1772	150	395
5	Response	1805	150	474
6	Response	1800	150	502
7	Narrative	1569	300	732
8	Narrative	723	650	1146

Table 13: Statistical information of the ASAP++ dataset.

B.2 Prompt List of ASAP++

Prompts of ASAP++ are listed in Table 14.

ID	Prompt
1	More and more people use computers, but not everyone agrees that this benefits society. Those who support advances in technology believe that computers have a positive effect on people. They teach hand-eye coordination, give people the ability to learn about faraway places and people, and even allow people to talk online with other people. Others have different ideas. Some experts are concerned that people are spending too much time on their computers and less time exercising, enjoying nature, and interacting with family and friends.
2	Censorship in the Libraries "All of us can think of a book that we hope none of our children or any other children have taken off the shelf. But if I have the right to remove that book from the shelf – that work I abhor – then you also have exactly the same right and so does everyone else. And then we have no books left on the shelf for any of us." –Katherine Paterson, Author Write a persuasive essay to a newspaper reflecting your vies on censorship in libraries. Do you believe that certain materials, such as books, music, movies, magazines, etc., should be removed from the shelves if they are found offensive? Support your position with convincing arguments from your own experience, observations, and/or reading.
3	Write a response that explains how the features of the setting affect the cyclist. In your response, include examples from the essay that support your conclusion.
4	Read the last paragraph of the story. "When they come back, Saeng vowed silently to herself, in the spring, when the snows melt and the geese return and this hibiscus is budding, then I will take that test again." Write a response that explains why the author concludes the story with this paragraph. In your response, include details and examples from the story that support your ideas.
5	Describe the mood created by the author in the memoir. Support your answer with relevant and specific information from the memoir.
6	Based on the excerpt, describe the obstacles the builders of the Empire State Building faced in attempting to allow dirigibles to dock there. Support your answer with relevant and specific information from the excerpt.
7	Write about patience. Being patient means that you are understanding and tolerant. A patient person experience difficulties without complaining. Do only one of the following: write a story about a time when you were patient OR write a story about a time when someone you know was patient OR write a story in your own way about patience.
8	We all understand the benefits of laughter. For example, someone once said, "Laughter is the shortest distance between two people." Many other people believe that laughter is an important part of any relationship. Tell a true story in which laughter was one element or part.

Table 14: Prompts of ASAP++ dataset.

C Experiment Details

C.1 Ablated Architectures

Figure 6 illustrates the architectures in Section 4.3.

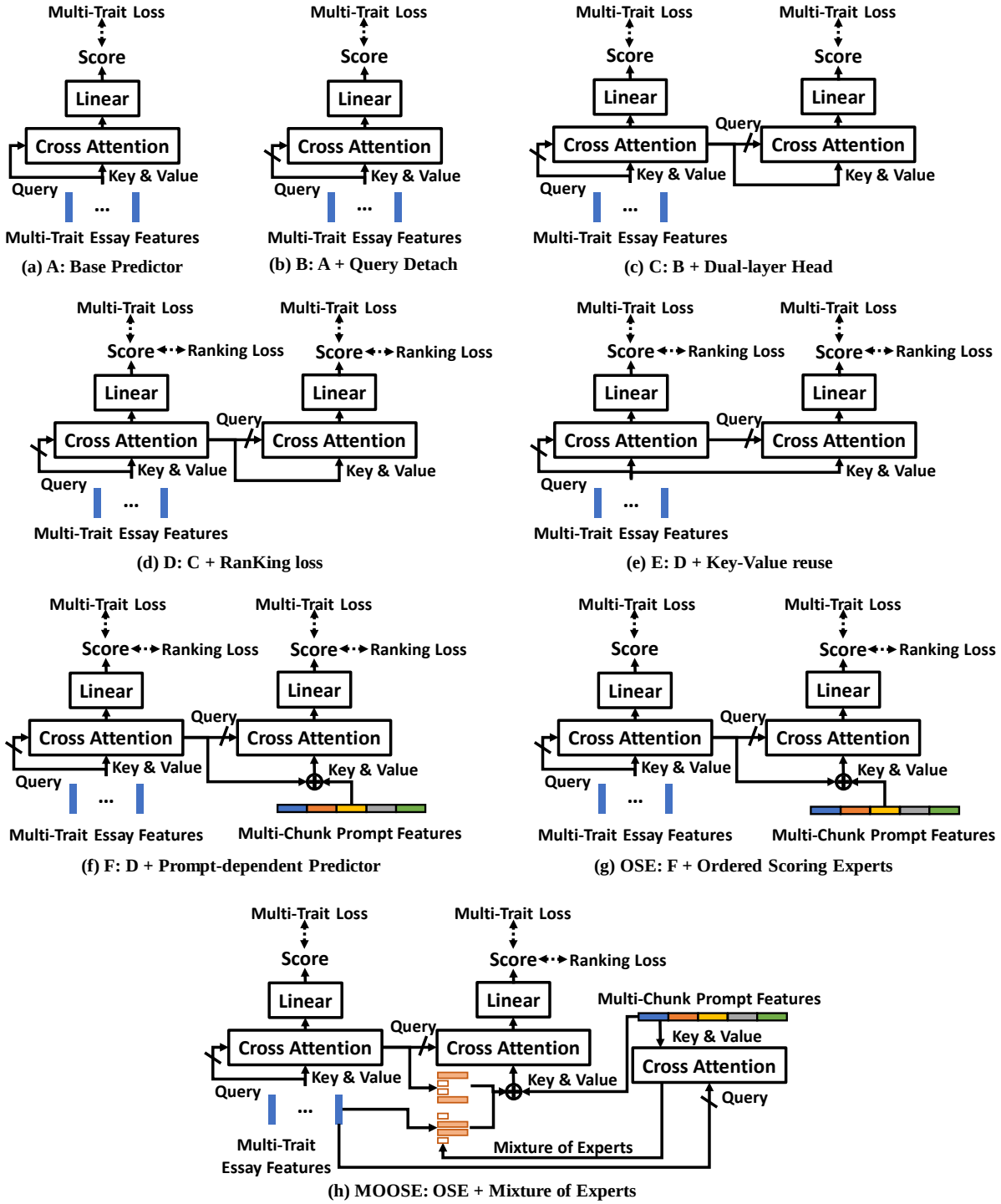

Figure 6: Ablated architectures in Section 4.3, including (a) base predictor, (b) Query Detach (QD), (c) Dual-layer Head (DH), (d) RanKing loss (RK), (e) Key-Value reuse (KV), (f) Prompt-dependent Predictor (PP), (g) Ordered Scoring Experts (OSE), and (h) Mixture Of Orderd Scoring Experts (MOOSE).

D How query detach (stop gradient) works

The scoring cue retrieval mechanism in MOOSE is highly related to the stop gradient operation, here we make a brief description to show how stop gradient works for MOOSE.

Stop gradient, also known as detach, is a common technique in training deep neural networks. It works by cutting off the gradient flow from a specific layer, preventing it from being used to update the weights and parameters of preceding layers. Below, we outline the effects of stop-gradient in different areas and its role in our work. In contrastive learning, stop gradient is often used to prevent mode collapse or to avoid having feature learning dominated by a small subset of neurons, such as (Chen and He, 2021). In Large Language Models (LLMs), it has been applied for progressive learning of language modeling tasks (Meng et al., 2021), independent layer updates (Deutch et al., 2024), and reducing training costs (Chevalier et al., 2023). The design of MOOSE is most similar to (Chen and He, 2021) and (Deutch et al., 2024), where stop-gradient is applied to the query branch to enable progressive learning via ordered scoring experts.

In standard Transformer training, both feature learning (query) and updates to the elements used for feature recomposition (values) occur simultaneously. However, in MOOSE, since the query undergoes a stop gradient operation, it remains unchanged and can be treated as a fixed input. In this setting, the values effectively serve as the necessary cues that transform the input into the targets. After training, the input will produce the consistent features and scoring cues through the trained model. Since the usage of values is positively correlated with their relevance to both the query and keys (which are identical to values), this mechanism aligns with retrieval tasks in NLP. As a result, we referred to this process as scoring cue retrieval.

E Efficiency of MOOSE

In terms of time efficiency, our method incurs lower computational cost compared to standard encoder-decoder models or large language models (LLMs). The critical inference path of our model includes BERT (12-layer Transformer), LSTM (sequence length / chunk size), Trait Attention (1-layer Transformer), and MOOSE (3-layer Transformer), totaling 16 Transformer layers and LSTM steps of the sequence length divided by chunk size. In comparison, encoder-decoder models like T5 have 12-layer Transformer encoders and 12-layer Transformer decoders, making both training and inference more expensive. If using the auto-regressive approach adopted by LLMs, the entire 12-layer Transformer decoder needs to run for each step in the sequence, resulting in a much higher inference cost.

The most expensive part of our architecture is the Multi-chunk BERT. The computational complexity of the Transformer is $O(n^2d + nd^2)$, where n is the sequence length and d the hidden dimension. A chunked Transformer with chunk size k has complexity $O((n/k)k^2d + kd^2)) = O(nkd + nd^2)$. In typical NLP tasks where the sequence length n is large, the complexity is dominated by sequence length, and thus chunking has minimal impact on overall inference speed. But in essay scoring, n is relatively small, so the hidden dimension d dominates the computational cost. Subsequent modules do not significantly add to the inference overhead because the multi-chunk BERT already uses attention pooling to reduce the sequence length.

For the ranking expert, training involves a cross-essay ranking loss to learn fine-grained distinctions between essays. At inference time, however, it operates on a single essay and produces a score prediction in real time.