KAES: Multi-aspect Shared Knowledge Finding and Aligning for Cross-prompt Automated Scoring of Essay Traits

Xia Li[1,2], **Wenjing Pan**[1]

[1]School of Information Science and Technology, Guangdong University of Foreign Studies, Guangzhou, China
[2]Center for Linguistics and Applied Linguistics, Guangdong University of Foreign Studies, Guangzhou, China
xiali@gdufs.edu.cn, wjpan@gdufs.edu.cn

Abstract

Cross-prompt automated essay scoring (AES) aims to train models using essays from different source prompts and test them on new target prompt essays. A core challenge of the task is to learn as much shared knowledge as possible between essays from different prompts in order to better represent new prompt essays. Previous studies primarily focus on learning this knowledge on a general, coarse-grained level, ignoring that the shared knowledge among prompts is highly detailed and contains a more comprehensive range of information that is not fully investigated. In this paper, we propose a novel multi-aspect knowledge finding and aligning optimization strategy to better acquire this detailed various shared knowledge. We also introduce LLM to extract explicit, interpretable knowledge from implicit, multi-aspect shared knowledge and use this knowledge to improve the representation and evaluation performance of new prompt essays. We conduct extensive experiments on public datasets. The results show that our approach outperforms current state-of-the-art models and is effective on cross-prompt AES.

Introduction

Automated essay scoring (AES) aims to evaluate the overall quality or specific traits of a given essay automatically. AES systems are widely used in the field of education assessment. It can reduce teachers' workload, provide students with rich feedback, and improve the efficiency and fairness of grading (McNamara et al. 2015).

A majority of AES systems focus on scoring essays written in response to a specific essay prompt[1], which means that the training and testing essays are from the same prompt (prompt-specific AES). Early works (Larkey 1998; Rudner and Liang 2002; Yannakoudakis, Briscoe, and Medlock 2011; Chen and He 2013; Attali and Burstein 2004; Phandi, Chai, and Ng 2015) focus on extracting rich handcrafted features to train scoring models. With the advent of deep learning, many neural network-based architectures (Taghipour and Ng 2016; Dong and Zhang 2016; Dong, Zhang, and Yang 2017; Tay et al. 2018; Mesgar and Strube 2018; Wang et al. 2022; Shibata and Uto 2022; Uto et al. 2023) have been proposed and achieve promising results. To provide enhanced feedback, several studies explore scoring specific traits of essays (Persing, Davis, and Ng 2010; Persing and Ng 2014; Mathias and Bhattacharyya 2020; Hussein, Hassan, and Nassef 2020). For example, Song et al. (2021) assesses essay structure and coherence for organization, and Ke et al. (2018) evaluate argument persuasiveness and logic.

As obtaining sufficient essays with human-rated scores for a new prompt is often difficult and expensive, cross-prompt AES (Phandi, Chai, and Ng 2015; Cummins, Zhang, and Briscoe 2016; Jin et al. 2018; Cao et al. 2020; Ridley et al. 2020; Li, Chen, and Nie 2020; Ridley et al. 2021; Do, Kim, and Lee 2023; Chen and Li 2023, 2024) has gained increasing attention in recent years. These works employ different strategies, such as knowledge transfer, self-supervised learning, and prompt-agnostic handcrafted feature extraction, to learn as many shared features among different prompts as possible to better represent and score the target prompt essays.

Although previous cross-prompt AES studies have achieved encouraging results, they primarily focus on acquiring the shared knowledge on a general, coarse-grained level, lacking the acquisition and learning of multi-aspect

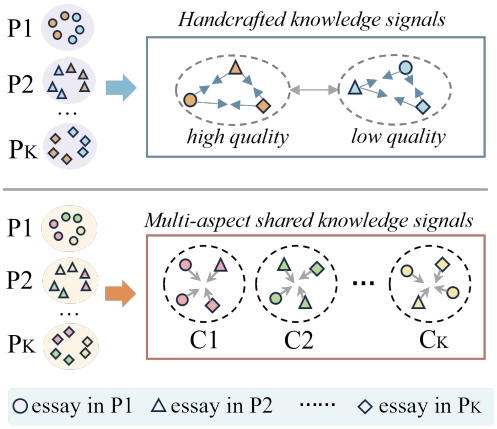

Figure 1: Illustration of handcrafted knowledge signals and multi-aspect shared knowledge signals retrieved by unsupervised clustering. $P_1 \sim P_K$ denotes prompt 1 to prompt K, and $C_1 \sim C_K$ denotes cluster 1 to cluster K.

Copyright © 2025, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

[1]In AES, the prompt refers to the writing theme of essays.

prompt-agnostic shared knowledge such as language grammar, text coherence, and writing style. While it is possible to tailor certain manual knowledge signals to guide the model in learning this kind of shared knowledge, the acquisition of more aspects of knowledge remains challenging due to its abstract nature and limited interpretability. For example, as shown in Figure 1 (upper part), we can set the essay qualities (e.g., high scores represent high quality and low scores represent low quality) as knowledge signals to align essays with similar quality. However, such manual designs may not be comprehensive; high-quality knowledge can also manifest in various aspects, such as having few grammatical errors (demonstrating knowledge of grammar) or displaying strong textual coherence (demonstrating knowledge of coherence). This suggests that the knowledge shared by different prompt essays can be multi-faceted. So, the questions are: 1) Can we develop an automated mechanism to help the model discover these various prompt-agnostic shared knowledge? 2) How to leverage these knowledge signals to optimize the representation of essays so that the model can effectively align essay representations of different prompts from much more diverse aspects, thereby enhancing the model's generalization capacity for representing and scoring of essays on new unseen prompts?

To address these two issues, we introduce a novel multi-aspect knowledge finding and aligning optimization strategy. The goal of the strategy is to automatically find the detailed multi-aspect knowledge and guide the model to learn the shared, consistent essay representation by aligning them in a prompt-independent manner. As shown in Figure 1 (lower part), we first use unsupervised clustering to automatically discover different clusters on essays sampled from all source prompts. Then, we optimize the model by aligning essays specific to the detailed, multi-aspect shared knowledge within the same cluster. In this way, the model autonomously learns consistent essay representations specific to different shared knowledge, independent of prompts.

Based on the presented strategy, we propose a novel dynamic multi-aspect knowledge optimization framework for cross-prompt automated essay scoring (KAES). Our proposed framework employs an iterative parameter update scheme. Specifically, we first train a scoring model over a certain number of epochs. Then, we use the encoder of the scoring model to represent the sampled essays from different source prompts and conduct finding and aligning optimization steps. The encoder is expected to be updated and guide the scoring model in light of those multi-aspect shared knowledge. After a certain number of iterations, the model will effectively learn the shared knowledge, and finally represent and score new prompt essays appropriately.

It is worth noting that, as the shared knowledge contained in each cluster is implicit and lacks interpretability, we propose to use LLM to explicitly capture this shared knowledge by describing the representative assays around the center of the cluster as feature text and representing it using BERT. In addition, instead of concatenating all the knowledge to the essay representation h_x, we select the cluster most similar to the essay representation h_x as its corresponding shared knowledge h_{x_shared}. We concatenate this explicit shared knowledge representation h_{x_shared} with the essay representation h_x to form the final representation h_{final} for scoring.

The main contribution of this paper can be summarized as follows:

- To the best of our knowledge, this is the first attempt to explore the learning of consistent representation of essays specific to multi-aspect shared knowledge among different prompts by introducing a finding, aligning, and selection strategy.
- We introduce LLM to extract explicit, interpretable knowledge from implicit, multi-aspect shared knowledge and use this knowledge to improve the representation and evaluation performance of new prompt essays.
- We conduct extensive experiments on the ASAP++ dataset, and the results show that our approach outperforms the state-of-the-art models and is effective in cross-prompt multi-trait AES.

Related Work

We will introduce related work from the perspectives of prompt-specific AES and cross-prompt AES.

Prompt-Specific AES. Prompt-specific AES methods focus on scoring essays belonging to the same prompt. The existing approaches employ various techniques to capture different features of the essay to improve scoring performance. For example, early studies (Rudner and Liang 2002; Chen and He 2013) focus on extracting rich handcrafted features to train scoring models. Later, neural network-based methods employ different strategies to advance the field. For example, Dong and Zhang (2016) models the hierarchical structure of the essay and Tay et al. (2018) captures sentence-level dependencies. Recent transformer-based models (Wang et al. 2022; Uto et al. 2023) further improve contextual understanding. To provide enhanced feedback, some studies explore scoring specific traits of essays, such as organization (Persing, Davis, and Ng 2010; Song et al. 2021), prompt adherence (Persing and Ng 2014) and argument persistence (Ke et al. 2018); others assess multiple traits of essays simultaneously (Mathias and Bhattacharyya 2020; Hussein, Hassan, and Nassef 2020). These advancements mark the shift from handcrafted features to neural models, enhancing scoring accuracy and feedback depth.

Cross-Prompt AES. Cross-prompt AES methods focus on scoring essays from different prompts. Some studies (Phandi, Chai, and Ng 2015; Cummins, Zhang, and Briscoe 2016; Cao et al. 2020) apply the transfer learning method to adapt models to new prompts for cross-prompt AES. For example, Jin et al. (2018) and Li, Chen, and Nie (2020) introduce a two-stage pseudo-labeling approach to improve cross-prompt AES performance. Other works explore prompt-related features for cross-prompt AES. For example, Do, Kim, and Lee (2023) employ a prompt-aware framework to improve scoring performance of the target prompt. Jiang et al. (2023) develop a representation learning method to separate the shared and prompt-specific features to improve cross-prompt generalization.

The above cross-prompt AES methods mainly focus on the scoring of single attribute. In recent years, some cross-

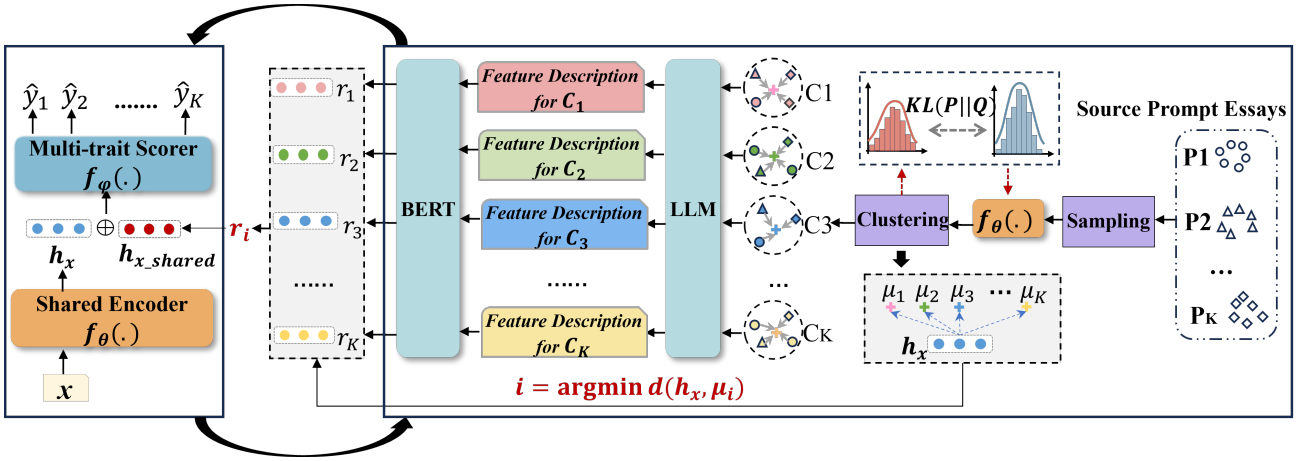

Figure 2: Overall architecture of KAES. During model training, multi-trait scoring training and knowledge finding and alignment are performed in alternating iterations. Through multi-trait scoring training, we obtain a shared encoder $f_\theta(.)$. For knowledge finding and alignment, we first sample essays from the source prompts, encode them using $f_\theta(.)$, and then perform clustering on these essays. We implicitly update θ to align essay representations in light of the knowledge signals from the clusters; We explicitly describe the features of the essays in the clusters using an LLM and obtain the knowledge representations using BERT. We dynamically select the knowledge representation h_{x_shared} corresponding to the centroid closest in distance to h_x for concatenation. During model inference, we select h_{x_shared} using the trained knowledge representations and centroids (shown in the gray box).

prompt AES models try to explore multi-trait scoring without using any target prompt essays during training. Ridley et al. (2020, 2021) incorporate handcrafted features from only source-prompt data into their model; Chen and Li (2023) employ a prompt-mapping strategy to learn shared features of source and target prompts; Chen and Li (2024) propose a prompt-generalized learning method based on meta-learning and a level-aware learning strategy.

Our work also focuses on cross-prompt multi-trait essay scoring and training the model without seeing any target prompt data. Although these previous cross-prompt multi-trait methods have achieved outstanding performance, they focus on capturing the generic shared knowledge across prompts. However, this is insufficient, as the shared knowledge is detailed and multifaceted. Unlike previous work, we propose a novel multi-aspect knowledge finding and aligning optimization strategy to automatically find and learn this knowledge and optimize our model in a prompt-independent manner.

Approach

As shown in Figure 2, the overall architecture of our KAES framework consists of the multi-trait scoring module, the knowledge finding and aligning module, and the training and inference module.

Task Definition

Our task focuses on cross-prompt AES, where a model is trained on source-prompt essays D_s and tested on unseen target-prompt essays D_t to predict multiple trait scores. Here, $D_s = \{(x_i, Y_i)|1 \leq i \leq N_{D_s}\}$, where x_i is the i-th source prompt essay and $Y_i = \left\{y_1^i, y_2^i, ..., y_{K_t}^i\right\}$ represents its K_t trait scores. Similarly, $D_t = \{(x_j, Y_j)|1 \leq j \leq N_{D_t}\}$, with x_j as the j-th target prompt essay and $Y_j = \left\{y_1^j, y_2^j, ..., y_{K_t}^j\right\}$ as its trait scores.

Multi-trait Scoring Module

Our scoring model, denoted as $F(.)$, consists of two components: the shared encoder $f_\theta(.)$ with parameters θ and the multi-trait scorer $f_\varphi(.)$ with parameters φ.

For the shared encoder, similar to previous studies, we use CNN-LSTM hierarchical structure encoders (Dong, Zhang, and Yang 2017) or finetuned BERT encoders (Devlin et al. 2019) as the shared encoder $f_\theta(.)$ to encode the essay, denoted as e. We use the same handcrafted features as Ridley et al. (2021), denoted as f. The representation of the essay is the concatenation of e and f, denoted as $h_x = [e; f]$.

In order to score multiple traits of an essay effectively, previous studies (Ridley et al. 2021; He et al. 2022) demonstrate that leveraging the mutual information of different traits is beneficial for trait scoring. Following their work, we adopt an independent dense layer for each trait and apply a trait attention mechanism. For the k-th trait scoring, we first input the final essay representation into the corresponding k-th dense layer to obtain the k-th trait representation a_k. We then apply a trait attention mechanism (Ridley et al. 2021) to obtain its attention vector p_k. Finally, We concatenate a_k and p_k to get the final representation of the k-th trait $g_k = [a_k; p_k]$. The k-th trait's predicted score $\hat{y}_k$ is obtained through a sigmoid activation:

$$\hat{y}_k = \text{Sigmoid}\left(g_k\right), k = 1, ..., K_t \quad (1)$$

We use mean square error as the scoring loss function.

Assuming that there are a total of N essays and K_t traits, the loss is defined as:

$$\mathcal{L}_{aes} = \frac{1}{NK_t} \sum_{i=1}^{N} \sum_{k=1}^{K_t} \left(\hat{y}_k^{(i)} - y_k^{(i)} \right)^2 \tag{2}$$

where $y_k^{(i)}$ and $\hat{y}_k^{(i)}$ are the ground truth and the predicted score of the k-th trait for the i-th essay, respectively. As some traits do not have ground truth scores, the masking mechanism (Ridley et al. 2021) is employed in the calculation.

Knowledge Finding and Aligning Module

To make our model capture the multi-aspect shared knowledge, we propose a knowledge finding and aligning strategy. First, we automatically discover this knowledge through unsupervised clustering on sampled essays, then implicitly align essay representations under the guidance of cluster signals, and finally use LLM to explicitly capture these various features based on the given texts from each cluster.

Knowledge Discovering We perform unsupervised clustering on a batch of essays sampled from all source prompts to discover various shared knowledge automatically. To enhance sample diversity, we employ stratified sampling to maintain balance across source prompts. Specifically, we select B essays per prompt and use the shared encoder $f_\theta(.)$ to encode them. Our KAES is not limited to one specific clustering method. Different clustering methods are applicable and available, such as K-Means (Hartigan and Wong 1979) and GMM (Rasmussen 1999). We cluster the sampled essays and obtain a total of K_c clusters, denoted as $\{C_1, C_2, ..., C_{K_c}\}$. The centroid of these clusters are represented as $\{\mu_1, \mu_2, ..., \mu_{K_c}\}$. During clustering, essays sharing similar knowledge across different prompts are clustered together. We treat each cluster label as a guiding signal corresponding to that knowledge, which will be used for the subsequent aligning and updating process.

Implicitly Knowledge Aligning Using the knowledge signals retrieved by unsupervised clustering, we aim to bring essays from the same cluster closer, thereby aligning their representations in the semantic space and consequently improving the model's ability to learn this shared knowledge. There are various approaches to achieving this goal, such as minimizing the distance between essay representations and centroids. Inspired by Wang et al. (2019), we employ a method that involves minimizing the divergence between two distributions, which has been shown to yield superior optimization results. We first obtain the clustering distribution Q, which measures the distance between each essay representation e_i and cluster center embedding μ_c. Based on Q, we construct an augment distribution P, which amplifies the influence of the assignments corresponding to high probabilities in Q, which indicates that the essays are in closer proximity to the cluster center. The calculations of the two distributions Q and P are as follows:

$$q_{ic} = \frac{\left(1 + \|e_i - \mu_c\|^2\right)^{-1}}{\sum_k \left(1 + \|e_i - \mu_k\|^2\right)^{-1}} \tag{3}$$

$$p_{ic} = \frac{q_{ic}^2 / \sum_j q_{jc}}{\sum_k \left(q_{ik}^2 / \sum_j q_{jk}\right)} \tag{4}$$

where q_{ic} is the probability in Q, p_{ic} is the probability in P. We then encourage the distribution P to approximate Q, which can promote closer proximity among essays that belong to the same cluster. We employ Kullback-Leibler (KL) divergence to measure the difference between the two distributions and construct a loss to update the parameters θ of the shared encoder. The loss is defined as follows:

$$\mathcal{L}_{cluster} = \mathrm{KL}(P \| Q) = \sum_i \sum_u p_{iu} \log \frac{p_{iu}}{q_{iu}} \tag{5}$$

where p_{iu}, q_{iu} represent the probability that essay e_i is assigned to cluster C_u in the two distributions, P and Q, respectively.

Explicitly Knowledge Capturing Due to the excellent text understanding and generation capabilities of existing large language models, such as ChatGPT (Floridi and Chiriatti 2020), these models can easily analyze the features of multiple texts and generate coherent descriptions. To explicitly capture the shared knowledge, we propose leveraging ChatGPT to generate the feature descriptions based on the text of essays within each cluster.

Specifically, for each cluster, we select M representative essays that are closest to the centroid μ_k:

$$\{i_1, i_2, ..., i_M\} = \mathrm{argmin}_i \left\{ d\left(e_{ki}, \mu_k\right) \right\} \tag{6}$$

where $d(e, \mu)$ denotes the euclidean distance between the essay representation e and the centroid μ, e_{ki} is the representation of the i-th essay in cluster C_k and $i \in \{1, 2, \ldots, |C_k|\}$. We carefully craft an instruction to guide ChatGPT in capturing the multi-aspect shared features of each cluster. The instruction is designed based on keywords extracted and synthesized from rubrics in the experiment dataset. ChatGPT generates a description of the common characteristics possessed by the selected representative essays for each cluster C_k:

$$T_k = \text{Prompt}(text\_i_1, text\_i_2, ..., text\_i_M) \tag{7}$$

where T_k is a textual description of shared features of the given texts. We then obtain the shared knowledge representation r_k using BERT and its `[CLS]` token output:

$$r_k = \text{BERT}(T_k) \tag{8}$$

For all K_c clusters, we obtain a set of shared knowledge representations, denoted as $\{r_1, r_2, ..., r_{kc}\}$.

Knowledge Selection As the shared knowledge across prompts is multifaceted, the essays from different prompts share the knowledge of different aspects. Therefore, when scoring an essay x, we propose a selection mechanism to enable the model to understand which aspect of knowledge this essay shares. This mechanism selects the corresponding shared knowledge from all the captured knowledge representations for the given essay. Specifically, we compute the

distance between the essay representation h_x and the centroids μ of these clusters. We select the explicit knowledge representation that has the smallest distance to h_x, denoted as $h_{x\_shared} = r_{\arg\min_i \|h_x - \mu_i\|}$. Finally, we concatenate h_x with h_{x_shared} and input the concatenated vector h_{final} into the multi-trait scorer for scoring. The knowledge representations dynamically change with each clustering iteration.

Training and Inference Module

Model Training Stage Our model employs an iterative training scheme in which the model training and knowledge finding and aligning steps are conducted alternately. Specifically, we first train a scoring model over a certain number of epochs using the loss in Equation (2). Then, we conduct finding and aligning steps to refine the model parameters using the clustering loss Equation (5) and obtain the knowledge representations as shown in Equation (8). After a certain number of iterations, the model will effectively learn the shared knowledge from various aspects and finally represent and score new prompt essays appropriately.

To preserve established essay representations and maintain the model's scoring performance, it is important to avoid overtraining the knowledge finding and aligning modules. We set the epoch of the multi-trait scoring training module to τ, and we perform the knowledge finding and aligning steps one time in each iteration to allow for a moderate refinement of the model's encoding.

Model Inference Stage During model inference, we first input the essay into the trained encoder to obtain its representation. We select the shared knowledge representation corresponding to the essay from the trained knowledge representations. The selection mechanism is the same and is based on the centroids stored in our model. We then concatenate the essay representation with the knowledge representation and input the concatenated representation into the multi-trait scorer to predict scores.

Experiments and Results

Dataset and Evaluation Metrics

We use the ASAP and ASAP++ datasets (Mathias and Bhattacharyya 2018) to evaluate our method. It includes 12,978 English writings in response to eight prompts. We employ the prompt-wise validation method, which is commonly utilized in existing cross-prompt AES studies (Jin et al. 2018; Ridley et al. 2021; Chen and Li 2023). Our evaluation metric is Quadratic Weighted Kappa (QWK), which is the official metric for the Kaggle competition ASAP and was widely used in previous AES methods. QWK quantifies the level of agreement between the human rater and the AES model. A higher QWK value indicates better scoring performance.

Implementation Details

For the encoder, we use 200 CNN filters and 200 LSTM units in hierarchical encoder and employ the `bert-base-uncased` BERT model. We set the batch size B to 200, the number of clusters K_c to 10, and the number of representative essays M to 3. The initial shared representation h_{x_shared} is randomly initialized. To avoid impairing original scoring performance, τ is set to 5, as $\mathcal{L}_{aes}$ stabilizes at 5 epoch. Our experiments are conducted on an NVIDIA RTX8000 GPU. The best model is selected based on the highest average QWK on the validation set. We run it three times and report the average results on the test set. Our code is available at https://github.com/gdufsnlp/KAES.

Baselines

We compare the baselines as follows. We use Hier att (Dong and Zhang 2016), AES aug (Hussein, Hassan, and Nassef 2020), PAES (Ridley et al. 2020), CTS no att, CTS (Ridley et al. 2021), PMAES (Chen and Li 2023) and PLAES (Chen and Li 2024) as baselines for their modeling multi-traits scoring and having the same experimental setups with ours. We also use ChatGPT and a Finetune BERT as baselines for comparison with our BERT encoder-based models. We experiment with five model variants based on different encoders and clustering methods: KAES(hier/bert)+kmeans/gmm. We use GPT-3.5 as the LLM backbone for the five model variants.

Main Results

The main results of our method (KAES) on each prompt and each trait are shown in Table 1 and 2. The results indicate that KAES demonstrates effectiveness in cross-prompt automated scoring of essay traits. Specifically, KAES(hier)+gmm outperforms all baselines, with an average improvement of 2.2% for prompts and 2.4% for traits compared to the SOTA model PLAES, achieving maximum gains of 5.1% on prompt P5 and 5.7% on trait WC. This outstanding improvement shows that our proposed multi-aspect knowledge optimization strategy is effective for enhancing multi-trait scoring task.

The results also show that employing either GMM or Kmeans yields comparable model performance, with GMM performing slightly better. This improvement may stem from its compatibility with the Gaussian-based alignment strategy employed in KAES. Additionally, the results indicate that different essay encoders yield varying performance on the cross-prompt AES task. KAES(hier)+gmm outperforms KAES(bert)+gmm by 7.0% on prompts and 6.3% on traits.

Considering the distinct text generation capacities of LLMs, we also compare the performance impact of using GPT-4 and GPT-3.5 as the LLM backbone for KAES. GPT-4 achieves the best performance (0.606 on prompts, 0.602 on traits) compared to GPT-3.5 (0.597 on prompts, 0.594 on traits). These results suggest higher-quality text generation enhances scoring by improving descriptive features.

Ablation Studies

The ablation results in Table 3 provide insights into the contribution of two components in KAES: the cluster-based alignment optimization process (*Cluster*) and the LLM-based knowledge capture process (*LLM*). We observe the following findings: 1) When *LLM* is removed, model performance decreases by 0.9% on prompts and 1.1% on traits. 2) When both *Cluster* and *LLM* are removed, performance

Model	P1	P2	P3	P4	P5	P6	P7	P8	AVG
GPT-3.5-turbo *(0-shot)*	0.264	0.492	0.351	0.437	0.516	0.489	0.153	0.307	0.376
Finetune BERT	0.556	0.549	0.616	0.618	0.655	0.457	0.345	0.275	0.509
Hi att (Dong and Zhang 2016)	0.315	0.478	0.317	0.478	0.375	0.357	0.205	0.265	0.349
AES aug (Hussein, Hassan, and Nassef 2020)	0.330	0.518	0.299	0.477	0.341	0.399	0.162	0.200	0.341
PAES (Ridley et al. 2020)	0.605	0.522	0.575	0.606	0.634	0.545	0.356	0.447	0.536
CTS no att (Ridley et al. 2021)	0.619	0.539	0.585	0.616	0.616	0.544	0.363	0.461	0.543
CTS (Ridley et al. 2021)	0.623	0.540	0.592	0.623	0.613	0.548	0.384	0.504	0.553
PMAES (Chen and Li 2023)	0.656	0.553	0.598	0.606	0.626	0.572	0.386	0.530	0.566
PLAES (Chen and Li 2024)	0.648	0.563	0.604	0.623	0.634	0.593	0.403	0.533	0.575
Our Model									
KAES(bert)+kmeans	0.580	0.570	0.591	0.620	0.663	0.510	0.325	0.301	0.520
KAES(bert)+gmm	0.604	0.581	0.568	0.651	0.668	0.486	0.349	0.310	0.527
KAES(hier)+kmeans	**0.632**	**0.617**	**0.627**	0.630	0.656	0.573	0.429	**0.598**	0.595
KAES(hier)+gmm	0.623	0.611	0.619	**0.636**	**0.685**	**0.593**	**0.433**	0.577	**0.597**

Table 1: Main results on each prompt. The average QWK across all traits for each prompt is reported.

Model	Overall	Content	Org	WC	SF	Conv	PA	Lang	Nar	AVG
GPT-3.5-turbo *(0-shot)*	0.406	0.411	0.286	0.330	0.282	0.285	0.426	0.462	0.439	0.370
Finetune BERT	0.578	0.490	0.361	0.542	0.527	0.298	0.548	0.611	0.591	0.505
Hi att (Dong and Zhang 2016)	0.453	0.348	0.243	0.416	0.428	0.244	0.309	0.293	0.379	0.346
AES aug (Hussein, Hassan, and Nassef 2020)	0.402	0.342	0.256	0.402	0.432	0.239	0.331	0.313	0.377	0.344
PAES (Ridley et al. 2020)	0.657	0.539	0.414	0.531	0.536	0.357	0.570	0.531	0.605	0.527
CTS no att (Ridley et al. 2021)	0.659	0.541	0.424	0.558	0.544	0.387	0.561	0.539	0.605	0.535
CTS (Ridley et al. 2021)	0.670	0.555	0.458	0.557	0.545	0.412	0.565	0.536	0.608	0.545
PMAES (Chen and Li 2023)	0.671	0.567	0.481	0.584	0.582	0.421	0.584	0.545	0.614	0.561
PLAES (Chen and Li 2024)	0.673	0.574	0.491	0.579	0.580	0.447	0.601	0.554	0.631	0.570
Our Model										
KAES(bert)+kmeans	0.551	0.501	0.407	0.527	0.508	0.369	0.565	0.634	0.627	0.521
KAES(bert)+gmm	0.551	0.515	0.402	0.583	0.571	0.366	0.565	0.614	0.611	0.531
KAES(hier)+kmeans	**0.695**	0.589	**0.547**	0.604	**0.610**	**0.487**	0.599	0.563	0.632	0.592
KAES(hier)+gmm	0.679	**0.594**	0.543	**0.636**	0.574	0.476	**0.611**	**0.595**	**0.641**	**0.594**

Table 2: Main results of each trait. The average QWK across all prompts for each trait is reported.

drops by 3.6% on prompts and 3.9% on traits. This shows that our multi-aspect shared knowledge acquisition strategy significantly enhances the overall performance of KAES.

Discussion

What is the Impact of Our Method on In-domain Performance while Enhancing Cross-domain Performance?

Previous research has primarily focused on cross-domain performance without adequately addressing in-domain stability (Yang et al. 2024). We investigate the effect of our method on in-domain performance. We adjust our experimental setup by withholding 10% of essays from all source prompts and training the model with the remaining 90% essays. We evaluate target prompts P1 and P2 for cross-domain performance and on the withheld essays for in-domain performance, focusing on the 'overall' trait score common to all prompts. Results in Table 4 indicate that while in-domain performance for each prompt varies, it remains generally stable or slightly decreases average. However, cross-domain performance on P1 and P2 both improves notably, showing that our method effectively enhances cross-domain capabilities without negatively impacting in-domain performance.

Does Our Method Need Large Increases in Model Parameters and Operational Costs?

As shown in Table 5, KAES has a parameter count of 837K, similar to CTS (855K), yet significantly enhances scoring accuracy, achieving improvements of 4.2% and 4.7% on prompts and traits, respectively. Moreover, compared to the SOTA model PLAES (1.47M), KAES operates with only half of PLAES's parameters, demonstrating its advantage in terms of parameter efficiency. In terms of the operational costs, KAES introduces additional time costs for the clustering process. Training KAES(hier) takes 180 seconds per epoch, totaling 2.5 hours over 50 epochs, while KAES(bert)

Model	Avg. QWK on prompts	Avg. QWK on traits
KAES	0.595	0.592
KAES w/o *LLM*	0.586	0.581
KAES w/o *Cluster & LLM*	0.559	0.553

Table 3: Ablation results on prompts and traits. KAES denotes our model KAES(hier)+kmeans.

Model	in domain	cross domain
	AVG (source)	P1 (target)
Hier	0.737	0.562
Hier+*Cluster*	0.729	0.634
Hier+*Cluster+LLM*	0.733	0.675
	AVG (source)	P2 (target)
Hier	0.779	0.628
Hier+*Cluster*	0.777	0.657
Hier+*Cluster+LLM*	0.776	0.662

Table 4: In-domain and cross-domain model performance.

requires 1200 seconds per epoch, reaching 6.7 hours over 20 epochs. Regarding memory usage, due to the need to load sampled essays from each prompt per clustering iteration, KAES occupies external GPU memory, with peak usage reaching 17,284 MiB when using 10% sample essays.

Can the Knowledge within Clusters be Visualized through Feature Descriptions Generated by LLM?

Knowledge within clusters is difficult to interpret and quantify manually. Our approach utilizes LLM to generate coherent, human-understandable descriptions of shared features for each cluster. These generated feature descriptions enable qualitative visualization of this knowledge. Figure 3 shows the LLM instruction and two cases of generated feature descriptions. The instruction provides potentially shared feature dimensions summarized from the scoring rubric to ensure that the descriptions reflect the quality of the essay. It is evident that in both cases, the LLM effectively describes the features of knowledge within clusters based on rubric dimensions, with distinct descriptions for each cluster. For example, Case 1 highlights knowledge about grammatical errors in the corresponding cluster, while Case 2 describes strong language expression of the essays in its cluster.

Model	Number of Parameters	Avg. QWK on prompts	Avg. QWK on traits
CTS	855 K	0.553	0.545
PMAES	174 K	0.566	0.561
PLAES	1.47 M	0.575	0.570
KAES (ours)	837 K	0.595	0.592

Table 5: Comparison of parameters and performance.

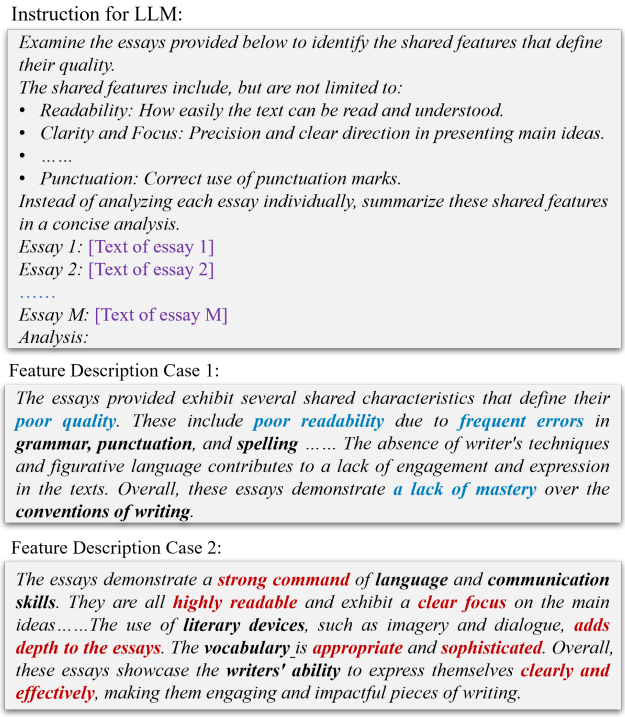

Figure 3: Prompt for LLM and feature descriptions cases. Purple text indicates representative essay input and blue and red text denote negative and positive descriptions.

How Do the Number of Clusters and the Sampling Quantity Influence the Model's Performance?

To assess how variations in clustering parameters impact the model's performance, we adjust the number of clusters and the quantity of sampled essays, observing changes in model performance. It shows that model performance subtly varies with different cluster numbers K_c, achieving optimal performance at $K_c = 5$. Additionally, while sampling only 50 essays per prompt exhibits limited effectiveness, increasing the number of sampled essays markedly enhances performance, indicating that larger sample sizes for clustering lead to improved results.

Conclusion

This paper introduces a novel multi-aspect shared knowledge finding and aligning strategy and an iterative training-based optimization mechanism (KAES) for cross-prompt multi-trait AES. We pioneer the use of clustering and LLM-based methods to identify and leverage multi-aspect shared knowledge. Our approach enhances the interpretability and utility of implicit knowledge through explicit knowledge representations. Extensive experiments on the public dataset demonstrate that our method significantly outperforms existing SOTA models in scoring performance and generalization. In addition, we believe that our approach is well-suited for other tasks, such as cross-domain sentiment classification and cross-lingual text classification.

Acknowledgments

This work is supported by the National Natural Science Foundation of China [grant number: 61976062].

References

Attali, Y.; and Burstein, J. 2004. AUTOMATED ESSAY SCORING WITH E-RATER® V.2.0. *ETS Research Report Series*, 2004(2): i–21.

Cao, Y.; Jin, H.; Wan, X.; and Yu, Z. 2020. Domain-Adaptive Neural Automated Essay Scoring. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, 1011–1020. New York, NY, USA: Association for Computing Machinery. ISBN 9781450380164.

Chen, H.; and He, B. 2013. Automated Essay Scoring by Maximizing Human-Machine Agreement. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, 1741–1752. Seattle, Washington, USA: Association for Computational Linguistics.

Chen, Y.; and Li, X. 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)*, 1489–1503. Toronto, Canada: Association for Computational Linguistics.

Chen, Y.; and Li, X. 2024. PLAES: Prompt-generalized and Level-aware Learning Framework for Cross-prompt Automated Essay Scoring. In Calzolari, N.; Kan, M.-Y.; Hoste, V.; Lenci, A.; Sakti, S.; and Xue, N., eds., *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, 12775–12786. Torino, Italia: ELRA and ICCL.

Cummins, R.; Zhang, M.; and Briscoe, T. 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)*, 789–799. Berlin, Germany: Association for Computational Linguistics.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 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)*, 4171–4186. Minneapolis, Minnesota: Association for Computational Linguistics.

Do, H.; Kim, Y.; and Lee, G. G. 2023. Prompt- and Trait Relation-aware Cross-prompt Essay Trait Scoring. In *Findings of the Association for Computational Linguistics: ACL 2023*, 1538–1551. Toronto, Canada: Association for Computational Linguistics.

Dong, F.; and Zhang, Y. 2016. Automatic Features for Essay Scoring – An Empirical Study. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 1072–1077. Austin, Texas: Association for Computational Linguistics.

Dong, F.; Zhang, Y.; and Yang, J. 2017. Attention-based Recurrent Convolutional Neural Network for Automatic Essay Scoring. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, 153–162. Vancouver, Canada: Association for Computational Linguistics.

Floridi, L.; and Chiriatti, M. 2020. GPT-3: Its Nature, Scope, Limits, and Consequences. *Minds and Machines*, 30: 681–694.

Hartigan, J. A.; and Wong, M. A. 1979. A K-Means Clustering Algorithm. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 28(1): 100–108.

He, Y.; Jiang, F.; Chu, X.; and Li, P. 2022. Automated Chinese Essay Scoring from Multiple Traits. In *Proceedings of the 29th International Conference on Computational Linguistics*, 3007–3016. Gyeongju, Republic of Korea: International Committee on Computational Linguistics.

Hussein, M. A.; Hassan, H. A.; and Nassef, M. 2020. A Trait-based Deep Learning Automated Essay Scoring System with Adaptive Feedback. *International Journal of Advanced Computer Science and Applications*, 11(5).

Jiang, Z.; Gao, T.; Yin, Y.; Liu, M.; Yu, H.; Cheng, Z.; and Gu, Q. 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)*, 12456–12470. Toronto, Canada: Association for Computational Linguistics.

Jin, C.; He, B.; Hui, K.; and Sun, L. 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)*, 1088–1097. Melbourne, Australia: Association for Computational Linguistics.

Ke, Z.; Carlile, W.; Gurrapadi, N.; and Ng, V. 2018. Learning to Give Feedback: Modeling Attributes Affecting Argument Persuasiveness in Student Essays. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, IJCAI'18, 4130–4136. AAAI Press. ISBN 9780999241127.

Larkey, L. S. 1998. Automatic Essay Grading Using Text Categorization Techniques. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, 90–95. New York, NY, USA: Association for Computing Machinery. ISBN 1581130155.

Li, X.; Chen, M.; and Nie, J.-Y. 2020. SEDNN: Shared and enhanced deep neural network model for cross-prompt automated essay scoring. *Knowledge-Based Systems*, 210: 106491.

Mathias, S.; and Bhattacharyya, P. 2018. ASAP++: Enriching the ASAP Automated Essay Grading Dataset with Essay Attribute Scores. In Calzolari, N.; Choukri, K.; Cieri, C.; Declerck, T.; Goggi, S.; Hasida, K.; Isahara, H.; Maegaard, B.; Mariani, J.; Mazo, H.; Moreno, A.; Odijk, J.; Piperidis, S.; and Tokunaga, T., eds., *Proceedings of the Eleventh International Conference on Language Resources and Evalu-*

ation (LREC 2018). Miyazaki, Japan: European Language Resources Association (ELRA).

Mathias, S.; and Bhattacharyya, P. 2020. Can Neural Networks Automatically Score Essay Traits? In *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications*, 85–91. Seattle, WA, USA → Online: Association for Computational Linguistics.

McNamara, D. S.; Crossley, S. A.; Roscoe, R. D.; Allen, L. K.; and Dai, J. 2015. A hierarchical classification approach to automated essay scoring. *Assessing Writing*, 23: 35–59.

Mesgar, M.; and Strube, M. 2018. A Neural Local Coherence Model for Text Quality Assessment. In Riloff, E.; Chiang, D.; Hockenmaier, J.; and Tsujii, J., eds., *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 4328–4339. Brussels, Belgium: Association for Computational Linguistics.

Persing, I.; Davis, A.; and Ng, V. 2010. Modeling Organization in Student Essays. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, 229–239. Cambridge, MA: Association for Computational Linguistics.

Persing, I.; and Ng, V. 2014. Modeling Prompt Adherence in Student Essays. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1534–1543. Baltimore, Maryland: Association for Computational Linguistics.

Phandi, P.; Chai, K. M. A.; and Ng, H. T. 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*, 431–439. Lisbon, Portugal: Association for Computational Linguistics.

Rasmussen, C. 1999. The Infinite Gaussian Mixture Model. In Solla, S.; Leen, T.; and Müller, K., eds., *Advances in Neural Information Processing Systems*, volume 12. MIT Press.

Ridley, R.; He, L.; Dai, X.; Huang, S.; and Chen, J. 2020. Prompt Agnostic Essay Scorer: A Domain Generalization Approach to Cross-prompt Automated Essay Scoring. *CoRR*, abs/2008.01441.

Ridley, R.; He, L.; Dai, X.-y.; Huang, S.; and Chen, J. 2021. Automated Cross-prompt Scoring of Essay Traits. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(15): 13745–13753.

Rudner, L. M.; and Liang, T. 2002. Automated Essay Scoring Using Bayes' Theorem. *The Journal of Technology, Learning and Assessment*, 1(2).

Shibata, T.; and Uto, M. 2022. Analytic Automated Essay Scoring Based on Deep Neural Networks Integrating Multidimensional Item Response Theory. In *Proceedings of the 29th International Conference on Computational Linguistics*, 2917–2926. Gyeongju, Republic of Korea: International Committee on Computational Linguistics.

Song, W.; Song, Z.; Liu, L.; and Fu, R. 2021. Hierarchical Multi-Task Learning for Organization Evaluation of Argumentative Student Essays. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, IJCAI'20. ISBN 9780999241165.

Taghipour, K.; and Ng, H. T. 2016. A Neural Approach to Automated Essay Scoring. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 1882–1891. Austin, Texas: Association for Computational Linguistics.

Tay, Y.; Phan, M.; Tuan, L. A.; and Hui, S. C. 2018. SkipFlow: Incorporating Neural Coherence Features for End-to-End Automatic Text Scoring. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1).

Uto, M.; Aomi, I.; Tsutsumi, E.; and Ueno, M. 2023. Integration of Prediction Scores From Various Automated Essay Scoring Models Using Item Response Theory. *IEEE Transactions on Learning Technologies*, 1–18.

Wang, C.; Pan, S.; Hu, R.; Long, G.; Jiang, J.; and Zhang, C. 2019. Attributed Graph Clustering: A Deep Attentional Embedding Approach. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, 3670–3676. International Joint Conferences on Artificial Intelligence Organization.

Wang, Y.; Wang, C.; Li, R.; and Lin, H. 2022. On the Use of Bert for Automated Essay Scoring: Joint Learning of Multi-Scale Essay Representation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 3416–3425. Seattle, United States: Association for Computational Linguistics.

Yang, K.; Raković, M.; Li, Y.; Guan, Q.; Gašević, D.; and Chen, G. 2024. Unveiling the Tapestry of Automated Essay Scoring: A Comprehensive Investigation of Accuracy, Fairness, and Generalizability. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(20): 22466–22474.

Yannakoudakis, H.; Briscoe, T.; and Medlock, B. 2011. A New Dataset and Method for Automatically Grading ESOL Texts. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, 180–189. Portland, Oregon, USA: Association for Computational Linguistics.