

Autoregressive Multi-trait Essay Scoring via Reinforcement Learning with Scoring-aware Multiple Rewards

Heejin Do^{1,2}, Sangwon Ryu¹, Gary Geunbae Lee^{1,2}

¹Graduate School of Artificial Intelligence, POSTECH, South Korea

²Department of Computer Science and Engineering, POSTECH, South Korea

{heejindo, ryusangwon, gblee}@postech.ac.kr

Abstract

Recent advances in automated essay scoring (AES) have shifted towards evaluating multiple traits to provide enriched feedback. Like typical AES systems, multi-trait AES employs the quadratic weighted kappa (QWK) to measure agreement with human raters, aligning closely with the rating schema; however, its non-differentiable nature prevents its direct use in neural network training. In this paper, we propose Scoring-aware Multi-reward Reinforcement Learning (SaMRL), which integrates actual evaluation schemes into the training process by designing QWK-based rewards with a mean-squared error penalty for multi-trait AES. Existing reinforcement learning (RL) applications in AES are limited to classification models despite associated performance degradation, as RL requires probability distributions; instead, we adopt an autoregressive score generation framework to leverage token generation probabilities for robust multi-trait score predictions. Empirical analyses demonstrate that SaMRL facilitates model training, notably enhancing scoring of previously inferior prompts.

1 Introduction

An essay can be evaluated from diverse perspectives, such as *Content*, *Sentence Fluency*, and *Organization*. As providing multi-view assessment is essential for enhancing the learner’s writing skill, recent attention to automated essay scoring (AES) systems has shifted from solely relying on holistic scoring (Taghipour and Ng, 2016; Dong and Zhang, 2016; Dong et al., 2017; Wang et al., 2022) to evaluating multiple trait scores (Kumar et al., 2022; Ridley et al., 2021; Do et al., 2024). Although simultaneous assessment for multiple traits is more challenging than a holistic paradigm, it has been much less explored.

Typically, AES systems are evaluated using the Quadratic Weighted Kappa (QWK) (Cohen, 1968)

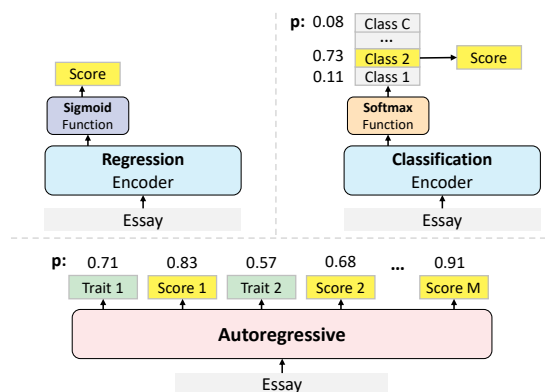


Figure 1: Overview of distinct AES frameworks. The autoregressive framework eliminates the need for multiple trait-wise layers. Classification and autoregressive AES models probabilistically predict final scores; hence, a policy gradient reinforcement algorithm is applicable.

score, which measures the agreement between human ratings and model predictions. Despite its effectiveness and close alignment with real-world rating schemes, its non-differentiable nature prevents direct use in neural-network training (Wang et al., 2018). Instead, previous AES models predominantly utilized cross-entropy or mean squared error (MSE) loss to train classification- or regression-based AES models, respectively (Figure 1).

In this paper, we propose a Scoring-aware Multi-reward Reinforcement Learning (SaMRL) method to unlock the potential of the QWK for training multi-trait AES systems. By constructing multiple rewards of bi-directional QWK and MSE penalty, SaMRL effectively incorporates nuanced measurement schemes in training phrases. Generally, the QWK score derives from a set of essays rather than a single score; thus, when applied conventionally in a batch set, it assigns the same metric to every sample. To ensure stable training, we introduce trait-wise comparison for QWK, integrating them with the batch-wise calculation to construct a bi-directional QWK reward.

Applying RL in AES is underexplored, and prior work (Wang et al., 2018) is limited to holistic scoring. Further, their method is restricted to the classification approach despite its inferior performance than the regression, as policy gradients for RL require probability distributions. Unlike prior works, we treat AES as a generation paradigm (Do et al., 2024), leveraging token generation probability distributions instead of categorical ones for policy gradient. Note that standard autoregressive AES is trained with cross-entropy and does not reflect any score-related metrics (i.e., MSE or QWK) during training; however, SaMRL enables direct parameter updates based on those scoring-aware metrics.

Extensive experiments on the representative ASAP and ASAP++ datasets demonstrate scoring enhancement over robust baselines across the traits and prompts. Comprehensive Analyses, which compare SaMRL with both single-reward applications and unidirectional use of QWK rewards, further reveal the robustness of our method. Notably, significant improvements observed on prompts with a broader score range highlight the overcoming of challenges posed by prior use of RL in AES.

2 Related works

Multi-trait essay scoring Although automated essay scoring has achieved notable success (Dong et al., 2017; Yang et al., 2020; Wang et al., 2022), research on multi-trait essay scoring is still underdeveloped and requires further exploration. Early attempts for multi-trait AES lie in constructing multiple trait-specific layers or models for different predictions (Mathias and Bhattacharyya, 2020; Ridley et al., 2021; Kumar et al., 2022; Do et al., 2023). Pointing out the inefficiency of duplicating individual encoder-only models generating a single score, Do et al. (2024) sequentially produces multi-trait scores by defining scoring as a decoder-introduced text generation. Autoregressively generating multi-trait scores significantly improved all trait-scoring performance, achieving stat-of-the-art results. Further, it reduces the burden of designing separate trait-specific layers as it consecutively predicts full trait scores for entire prompts with a single model. To take advantage of the efficiency and high performance, we introduce our RL method based on their autoregressive AES paradigm.

RL for text generation Recently, RL has been actively employed across diverse natural language generation tasks. Notably, the advent of reinforce-

ment learning from human feedback (RLHF) has improved the capabilities of general-purpose large language models (LLMs) such as GPT, showing the strength of RL (Ouyang et al., 2022). Numerous researchers have applied RL to specific downstream tasks such as text summarization (Paulus et al., 2018; Dong et al., 2018; Chen and Bansal, 2018; Narayan et al., 2018; Pasunuru and Bansal, 2018; Gunasekara et al., 2021; Parnell et al., 2022; Roit et al., 2023; Ribeiro et al., 2023; Su et al., 2023; Ryu et al., 2024; Singh et al., 2024), machine translation (Wu et al., 2018; He et al., 2024), and reasoning (Havrilla et al., 2024; Dutta et al., 2024; Xi et al., 2024; Lu et al., 2024). They aim to enhance performance by using rewards tailored to their specific objectives. For instance, in text summarization, Stiennon et al. (2020) employ human feedback as a reward model to generate summaries that align with human preferences, while Roit et al. (2023) use the entailment relationship between summary and source document as a reward to generate a factually consistent summary. In arithmetic reasoning problems, Dutta et al. (2024) utilizes a non-differentiable symbolic solver as a reward to address multi-step reasoning. We integrate the RL framework to unexplored autoregressive multi-trait AES by introducing novel multiple scoring-aware rewards.

3 Preliminary

We adopt the policy gradient reinforcement learning to train the policy to optimize rewards. Policy gradient aims to increase the probability of actions that yield high rewards. To guide the model towards taking actions that result in a higher reward, the policy gradient function includes the probability of actions taken by the policy, $\pi_{\theta}(a | s)$:

$$L^{PG}(\theta) = \hat{E}_t \left[\log \pi_{\theta}(a_t | s_t) \hat{A}_t \right] \quad (1)$$

Most of the existing AES systems use encoder-only models like BERT (Devlin et al., 2019) and rely on regression- or classification-based approaches. In general, the model generates the essay embedding vector from a given essay input; however, the differences lie in the objective functions and the process of deriving the final score from the output vector. Regression models predict the score with a sigmoid function given the embedding vector and are trained with the MSE loss function between the label y and predicted score \hat{y} for n samples:

$$loss_{MSE} = \frac{1}{n} \sum_{i=1}^n (y - \hat{y})^2 \quad (2)$$

However, as they output a single scalar value, they are unsuitable for policy gradient, which requires a probability distribution for policy training. Contrarily, classification-based AES models output a probability distribution (p) given the essay vector with the softmax function, and cross-entropy loss is employed as training objectives:

$$loss_{CE} = - \sum_{i=1}^C y_i \log(p_i) \quad (3)$$

where C denotes the number of classes, and y_i is a one-hot vector having one for the ground-truth category. By treating scoring a single essay as an action, the RL mechanism can be applicable to classification models (Wang et al., 2018). However, classification-based AES typically underperforms regression approaches, particularly lagging behind in prompts with broader score ranges (Wang et al., 2018). This performance drop is attributed to the increased number of prediction candidate classes.

To eliminate a performance decline, we introduce the RL application for AES with a generation framework. In particular, we ground our method on an autoregressive multi-trait score generation model, ArTS (Do et al., 2024), instead of classification approaches. We consider generating text trait and score tokens as an action and treat token generation probability distribution as the probability of actions. In addition, to take advantage of comparing the exact error rate in regression, we leverage the MSE penalty as part of our multi-rewards.

4 SaMRL

To incorporate a rating schema in the training phase, SaMRL updates the policy model using multiple rewards obtained by a scoring-aware multi-reward function. In RL, relying solely on reward-based learning can lead the model to prioritize enhancing rewards exclusively, potentially losing its capacity to score essays in appropriate prediction form. Therefore, we employ a fixed-parameter anchor model to guide the policy to prevent significant deviations in training and maintain the trait patterns. Figure 2 describes the overall process of SaMRL.

4.1 Score generation model

In autoregressive multi-trait prediction, given the essay input with the prefix of "score the essay of

the prompt i : ", the model outputs the sequence of the trait name and the corresponding score, such as $[trait_1 score_1, trait_2 score_2, \dots, trait_m score_m]$. In this work, we denote the generated sequence as \hat{y} and the ground-truth sequence as y ; the extracted numeric trait scores from each are denoted as \hat{s} and s , respectively. The sequence-to-sequence T5 (Raffel et al., 2020) model is employed for score generation, adhering to the same trait prediction order as the baseline model (Do et al., 2024). This order progresses from traits with smaller data sizes to those with larger data sizes. Also, traits that are not evaluated in certain prompts are regarded and predicted as the "nan" value, such as $[trait_j nan]$.

4.2 Multi-rewards function

To manage both the overall consistency and agreement across ratings, as well as precision at individual score levels, we introduce multiple rewards: bidirectional QWK (r_Q) and mean trait-wise MSE reward (r_M). QWK accounts for the ordinal nature of essay scores and weighting exact and near matches; thus, it effectively captures the rating schema and is sensitive to the overall qualities (Wang et al., 2018). Meanwhile, MSE aggregates the exact difference between predicted and actual scores; hence, it provides a clear yet simple indication of individual score deviations from true labels.

QWK The QWK metric evaluates the agreement and consistency between human and system assessments across a set of essays, calculating a single QWK score for the entire set. Therefore, rewarding with the measurement within the batch set, Q_B , assigns the same reward to all samples, potentially resulting in unstable training. To establish a more precise and stable reward strategy, we introduce a trait-wise QWK, Q_T , which evaluates the agreement between the trait sets of predictions and gold labels at the sample level. Specifically, batch-wise quadratic weighted kappa score, Q_B , is defined as:

$$Q_B = 1 - \frac{\sum_{i,j} W_{i,j} C_{i,j}}{\sum_{i,j} W_{i,j} E_{i,j}} \quad (4)$$

where the $N \times N$ weight matrix $W_{i,j}$ is calculated as $(i - j)^2 / (N - 1)^2$ for the number of candidate ratings N . $C_{i,j}$ denotes the counts of the essays assigned i score by human grader, and j score by our system. $E_{i,j}$ represents the expected count of two ratings assigning the i and j scores, respectively, and calculated by the outer product of two ratings' histogram vectors (Wang et al., 2018). Akin to

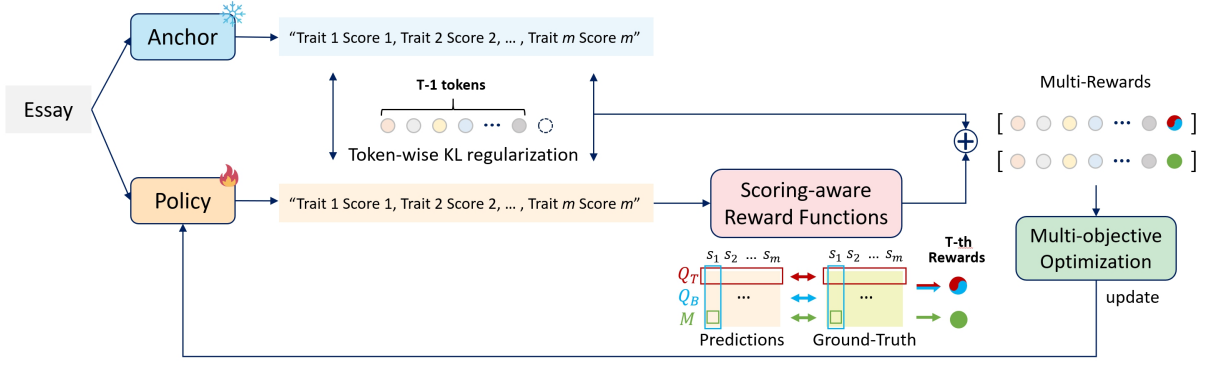


Figure 2: Overview of the entire process for the proposed autoregressive multi-trait AES with SaMRL. We maintain the structure of the score generation within the policy model through token-wise KL regularization and allow the model to align with human judgment by introducing multiple scoring-aware rewards.

the batch-wise measurement, the sample-level trait-wise QWK, Q_T , is computed using three metrics, but the focus is on the set of traits within a single sample rather than the essay set within a batch. Particularly, $C_{i,j}$ denotes the number of traits received a score of i by a human grader and a score of j by the system. Then, the bidirectional QWK reward r_Q , integrating two-way calculation, is defined as:

$$r_Q(S, \hat{S}) = \lambda \cdot Q_B + (1 - \lambda) \cdot Q_T \quad (5)$$

for in-batch prediction and actual score vectors \hat{S} and S , respectively. The prior RL for AES used the packed evaluation (Wang et al., 2018), averaging the QWK obtained from randomly selected essay packs. Conversely, our bidirectional approach eliminates the need for multiple assemblies per every target essay and removes random variability, achieving both efficient and reliable rewards.

MSE The autoregressive generation framework for scoring, which bases its predictions on token probabilities, does not involve direct numerical comparisons between predicted scores and human labels, such as error rates between ratings. We introduce MSE rewards as an auxiliary component to allow a generation-based scoring system to incorporate quantitative comparisons for ratings. As we consider multiple trait scores, we use mean trait-wise root mean squared error, which computes the average of the squared errors for each trait. The MSE reward, r_M is defined as follows:

$$r_M(S, \hat{S}) = -\frac{1}{m} \sum_{j=1}^m \sqrt{\frac{1}{n} \sum_{i=1}^n (s_{ij} - \hat{s}_{ij})^2} \quad (6)$$

for m number of traits and n number of predicted samples.

4.3 RL policy update

Our multi-trait AES model generates scores sequentially in a specific order and format. Consequently, a comprehensive evaluation is only feasible after the complete sequence has been generated, culminating with the final trait $[trait_m \ score_m]$. Thus, the model receives calculated multi-rewards just after the completion of the last trait score generation at the T -th timestep. Maintaining a structured format is crucial for scoring multiple traits in order, we adopt the token-wise KL regularization technique between the frozen anchor model, π^{AC} , and our trainable policy model π^θ . This process prevents the policy from overly adapting to the reward function while ensuring the preservation of the $[trait_j \ score_j]$ generation format. For each multi-rewards, in conjunction with the obtained reward and token-wise KL regularization until the $T - 1$ -th tokens, we update the policy via PPO (Schulman et al., 2017) with generalized advantage estimation (Schulman et al., 2016). The full reward R_k ($k \in \{M, Q\}$), is defined as follows:

$$R_k(e, y, \hat{y}) = r_k(y, \hat{y}) - \beta \log \left[\frac{\pi_\theta^{RL}(\hat{y}_t|e)}{\pi^{AC}(\hat{y}_t|e)} \right] \quad (7)$$

where e is the essay, \hat{y}_t is the t -th token in the model-generated sequence \hat{y} , and y is the reference sequence. The policy π_θ is the autoregressive multi-trait AES model, and e is the essay. In our work, the action denotes the selection of the next token by the policy. The action space V corresponds to the vocabulary of the policy π . Then, the PPO loss using R_k is defined as,

$$loss_t^k(\theta) = \hat{\mathbb{E}}_t \left[loss_t^{CLIP}(\theta) - c_1 loss_t^{VF}(\theta) + c_2 H[\pi_\theta](y_t) \right] \quad (8)$$

where $loss^{CLIP}$ represents the constrained surrogate loss through clipping, while the $loss^{VF}$ de-

Prompt	# of Essays	Average Length	Essay Type	Grade Level	Traits	Score Range	
						Overall	Trait
P1	1,783	350	Argumentative	8	Over, Cont, Org, WC, SF, Conv	2 - 12	1 - 6
P2	1,800	350	Argumentative	10	Over, Cont, Org, WC, SF, Conv	1 - 6	1 - 6
P3	1,726	150	Source-Dependent	10	Over, Cont, PA, Lan, Nar	0 - 3	0 - 3
P4	1,772	150	Source-Dependent	10	Over, Cont, PA, Lan, Nar	0 - 3	0 - 3
P5	1,805	150	Source-Dependent	8	Over, Cont, PA, Lan, Nar	0 - 4	0 - 4
P6	1,800	150	Source-Dependent	10	Over, Cont, PA, Lan, Nar	0 - 4	0 - 4
P7	1,569	300	Narrative	7	Over, Cont, Org, Conv, Style	0 - 30	0 - 6
P8	723	650	Narrative	10	Over, Cont, Org, WC, SF, Conv, Voice	0 - 60	2 - 12

Table 1: ASAP and ASAP++ combined dataset statistics. Over: *Overall*, Cont: *Content*, Org: *Organization*, WC: *Word Choice*, SF: *Sentence Fluency*, Conv: *Conventions*, PA: *Prompt Adherence*, Lang: *Language*, Nar: *Narrativity*.

notes the mean squared error update of the value function. H stands for entropy.

Multi-objective Optimization We consider two distinct multi-rewards— R_Q and R_M —for individually computing the PPO loss. Considering each loss as $loss_{R_Q}$ and $loss_{R_M}$, we treat the training process as multi-task learning with this formula:

$$loss_{total} = w_Q loss_{R_Q} + w_M loss_{R_M} \quad (9)$$

To dynamically optimize multiple loss functions along with their respective weights, we also train and update w_Q and w_M rather than relying on static interpolation. They are normalized with the softmax function.

5 Experimental setup

Datasets We use the open-sourced ASAP¹ and ASAP++² (Mathias and Bhattacharyya, 2018) datasets, as in the baseline ArTS model (Do et al., 2024). ASAP++ dataset includes enriched human-annotated multi-trait scores for English-written essays of eight separate prompts. As summarized in Table 1, different prompts are assessed with distinct traits with varied score ranges. While other traits are evaluated across several prompts, *Style* and *Voice* are only assessed in prompts 7 and 8, respectively, resulting in a very limited number of samples available for training. Our model handles all prompts and all traits with a single model. We also experiment with the publicly available Feedback Prize³ dataset, where argumentative essays are annotated with six trait scores, *Cohesion*, *Syntax*, *Vocabulary*, *Phraseology*, *Grammar*, and *Conventions*. Unlike the ASAP dataset, this data is not divided by prompts.

¹<https://www.kaggle.com/c/asap-aes>

²<https://lwsam.github.io/ASAP++/lrec2018.html>

³<https://www.kaggle.com/competitions/feedback-prize-english-language-learning>

Evaluations Following the previous works (Taghipour and Ng, 2016; Do et al., 2024), we use five-fold cross-validation using the same split as their works. The ASAP dataset’s official metric, QWK, is used as the evaluation metric, and five-fold averaged values and standard deviations are reported. We also measure separate QWK by prompts to ensure fair comparisons and prevent overly high scores in case of testing in the entire set Taghipour and Ng (2016); Do et al. (2024). Comparison models are detailed in Appendix A.

Settings For the experiments, we use the T5-Base and T5-Large (Raffel et al., 2020) models as the policy and anchor models, which are based on the Transformer architecture. We set the generation hyperparameters the same as the ArTS, using Seq2SeqTrainer with 5000 evaluation steps, 2 early stopping patients, 4 batch sizes, and 15 total epochs. For RL hyperparameters, we mainly follow previous RL-based models (Dutta et al., 2024; Roit et al., 2023; Ryu et al., 2024), adopting batch size of 4, discount factor γ as 0.99 and a learning rate of $1.41e-6$. For r_Q calculation in Equation 4.2, we use 0.5 as λ . A100-SMX4-8 and A6000 GPUs are used.

6 Result and Discussions

Main results The main experimental results for each trait in Table 2 demonstrate the effectiveness of our SaMRL method in enhancing the scoring quality across most traits, establishing new state-of-the-art. Noticeably, applying SaMRL to both ArTS-base and ArTS-large models exhibits a consistent trend of performance enhancements across the traits. We conducted a paired t-test on trait-wise average scores, and both the SaMRL-base and -large models showed significant improvements over the baseline ($p < 0.05$). Given that the T5-base and T5-large models have 220M and 770M parameters, respectively, these findings underscore the ro-

Model	Traits (Prediction Order: \leftarrow)											AVG \uparrow (SD \downarrow)
	Overall	Content	PA	Lang	Nar	Org	Conv	WC	SF	Style	Voice	
HISK	0.718	0.679	0.697	0.605	0.659	0.610	0.527	0.579	0.553	0.609	0.489	0.611 (-)
STL-LSTM	0.750	0.707	0.731	0.640	0.699	0.649	0.605	0.621	0.612	0.659	0.544	0.656 (-)
MTL-BiLSTM	0.764	0.685	0.701	0.604	0.668	0.615	0.560	0.615	0.598	0.632	0.582	0.638 (-)
ArTS	0.754	0.730	0.751	0.698	0.725	0.672	0.668	0.679	0.678	0.721	0.570	0.695 (\pm 0.018)
ArTS-base*	0.737	0.727	0.751	0.702	0.739	0.665	0.679	0.672	0.679	0.735	0.557	0.695 (\pm 0.022)
SaMRL-base (Ours)	0.750	0.732	0.754	0.704	0.740	0.670	0.684	0.681	0.685	0.726	0.558	0.699 (\pm 0.022)
ArTS-large*	0.752	0.729	0.749	0.701	0.727	0.677	0.683	0.683	0.683	0.712	0.621	0.702 (\pm 0.013)
SaMRL-large (Ours)	0.754	0.735	0.751	0.703	0.728	0.682	0.685	0.688	0.691	0.710	0.627	0.705 (\pm 0.013)

Table 2: Evaluated QWK results averaged across the prompts for each **trait**. Traits are predicted from right to left (\leftarrow). Five-fold averaged standard deviation is reported (SD). ArTS* is our implemented version, and ArTS is the reported ones in Do et al. (2024). Higher values among the implemented baseline and ours are represented in **bold**.

Model	Prompts								AVG \uparrow (SD \downarrow)
	1	2	3	4	5	6	7	8	
HISK	0.674	0.586	0.651	0.681	0.693	0.709	0.641	0.516	0.644 (-)
STL-LSTM	0.690	0.622	0.663	0.729	0.719	0.753	0.704	0.592	0.684 (-)
MTL-BiLSTM	0.670	0.611	0.647	0.708	0.704	0.712	0.684	0.581	0.665 (-)
ArTS	0.708	0.706	0.704	0.767	0.723	0.776	0.749	0.603	0.717 (\pm 0.025)
ArTS-base*	0.712	0.680	0.713	0.771	0.730	0.775	0.747	0.595	0.715 (\pm 0.016)
SaMRL-base (Ours)	0.717	0.703	0.715	0.773	0.729	0.778	0.745	0.604	0.720 (\pm 0.015)
ArTS-large*	0.700	0.699	0.704	0.766	0.725	0.770	0.739	0.644	0.718 (\pm 0.012)
SaMRL-large (Ours)	0.702	0.711	0.708	0.766	0.722	0.773	0.743	0.649	0.722 (\pm 0.012)

Table 3: Evaluated QWK results averaged across the traits for each **prompt**.

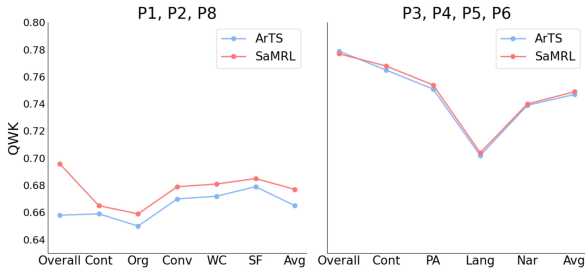


Figure 3: Comparison of performance between different prompt types with varying trait compositions. Prompts 1,2 and 8 are evaluated on the same traits, while 3-6 prompts are assessed on the other same traits.

bustness and consistency of our method regardless of model size. Unlike the *Content* or *Overall* traits, which are evaluated on every prompt’s essay sets and are benefited by our approach, the *Style* trait did not benefit from our method. SaMRL incorporates QWK scores calculated between batches as rewards during the reinforcement learning phase; thus, traits that are more prevalent within a batch are likely to be more dominantly reflected. As *Style* is evaluated only in prompt 7, and the number of corresponding essays is limited to 1,569 (Table 1), its contribution within a batch might be overshadowed by others.

Similar observations are evident in the prompt-wise results presented in Table 3, which displays

improvements across most prompts. Like the trait results, our method consistently exhibited increasing trends across different prompts, irrespective of model size. Notably, remarkable improvements are observed in prompts 1, 2, and 8, which share the same trait sets, *Cont*, *Org*, *WC*, *SF*, *Conv* as detailed in Table 1. Consequently, we further analyze those prompts by examining the performance variations for each associated trait.

Impacts on different essay types Specifically, we separately investigate the trait-specific performance improvements among prompts sharing common traits, comparing prompts 1,2 and 8 with prompts 3–6. Intriguingly, Figure 3 reveals that our model’s impact varies with the evaluated prompt type. Our model, SaMRL, demonstrates more substantial performance gains for the *Argumentative* or *Narrative* types, such as prompts 1, 2, and 8, compared to the *Source-dependent* prompts 3–6, which necessitate essay writings based on source texts. The marked difference in scoring range—where the latter prompts are evaluated on a [0, 4] scale and the former on a broader range up to [0, 60]—highlights that our model effectively addresses the challenges posed by traditional classification-based RL models, which typically struggle in wider score ranges. These advances might be attributed

Model	Traits (Prediction Order: \leftarrow)											AVG \uparrow (SD \downarrow)
	Overall	Content	PA	Lang	Nar	Org	Conv	WC	SF	Style	Voice	
ArTS-base*	0.737	0.727	0.751	0.702	0.739	0.665	0.679	0.672	0.679	0.735	0.557	0.695 (\pm 0.018)
SaSRL _M	0.749	0.728	0.751	0.702	0.738	0.665	0.678	0.674	0.679	0.733	0.556	0.696 (\pm 0.020)
SaSRL _Q	0.730	0.720	0.745	0.704	0.733	0.660	0.670	0.664	0.669	0.733	0.570	0.691 (\pm 0.020)
SaMRL _{uniQ_T}	0.737	0.733	0.753	0.701	0.739	0.671	0.679	0.678	0.683	0.737	0.562	0.698 (\pm 0.023)
SaMRL _{uniQ_B}	0.734	0.731	0.753	0.701	0.737	0.667	0.681	0.674	0.680	0.729	0.556	0.695 (\pm 0.022)
SaMRL_{biQ} (Ours)	0.750	0.732	0.754	0.704	0.740	0.670	0.684	0.681	0.685	0.726	0.558	0.699 (\pm 0.022)

Table 4: Ablation results comparing the use of score-wise single rewards (SaSRL_M and SaSRL_Q) and the implementation of unidirectional QWK rewards (SaMRL_{uniQ_T} and SaMRL_{uniQ_B}) instead of bidirectional ones. SaMRL_{biQ} denotes our SaMRL model, using multi-rewards with bidirectional QWK reward.

Model	Prompts								AVG \uparrow (SD \downarrow)
	1	2	3	4	5	6	7	8	
ArTS-base*	0.712	0.680	0.713	0.771	0.730	0.775	0.747	0.595	0.715 (\pm 0.025)
SaSRL _M	0.708	0.699	0.714	0.770	0.729	0.774	0.747	0.599	0.718 (\pm 0.015)
SaSRL _Q	0.705	0.677	0.712	0.765	0.725	0.767	0.744	0.586	0.710 (\pm 0.016)
SaMRL _{uniQ_T}	0.718	0.683	0.710	0.773	0.732	0.776	0.750	0.600	0.718 (\pm 0.016)
SaMRL _{uniQ_B}	0.711	0.684	0.711	0.770	0.731	0.774	0.746	0.598	0.716 (\pm 0.016)
SaMRL_Q (Ours)	0.717	0.703	0.715	0.773	0.729	0.778	0.745	0.604	0.720 (\pm 0.015)

Table 5: Evaluated QWK results averaged across the traits for each **prompt**.

Model	Traits (Prediction Order: \leftarrow)							AVG
	Coh	Syn	Voc	Phr	Gram	Conv	AVG	
MTL*	0.462	0.507	0.519	0.505	0.484	0.527	0.501	
ArTS*	0.590	0.628	0.594	0.639	0.659	0.659	0.628	
SaMRL*	0.592	0.637	0.610	0.646	0.658	0.656	0.633	

Table 6: Results on the Feedback Prize dataset. Five-fold averaged QWK score is reported; Coh: *Cohesion*, Syn: *Syntax*, Voc: *Vocabulary*, Phr: *Phraseology*, Gram: *Grammar*, Conv: *Conventions*.

to direct updates of SaMRL grounded on score-difference-reflected rewards measured by QWK and MSE, which enables the inclusion of scoring schemas (e.g., imposing significant penalties for inaccurately predicting wide-range scores) that typical generation processes fail to accommodate. Moreover, the results suggest that our approach can be particularly advantageous in less favorable conditions, given that prompts 1, 2, and 8 collectively have 4.3K essay samples, while 3–6 prompts comprise 7.1K samples.

Feedback Prize dataset To validate the broader applicability of SaMRL, we conducted additional experiments using the Feedback Prize dataset. As shown in Table 6, there is an overall enhancement in performance across the traits. Notably, although this dataset comprises only 2.3K training samples, substantially fewer than the over threefold larger ASAP dataset, the observed improvements support our findings of benefits in less favorable conditions.

6.1 Ablation studies

Are multi-rewards more effective than single rewards? We conduct comprehensive ablation studies to analyze the respective and joint effects of each reward. Experimental results in Table 4 and 5 reveal that our multi-rewards are more effective than separately applying single rewards. When adopted individually, MSE-only reward (SaSRL_M) achieves higher performance than bidirectional QWK-only reward (SaSRL_Q). The result validates the efficacy of adjusting MSE objectives from logistic regression models to fit autoregressive frameworks via RL. Meanwhile, the joint use of both rewards (SaMRL) shows superior improvements across all prompts and traits except *Style*, indicating synergistic impacts of our multi-rewarding mechanism. Our findings align with the prior works (Dann et al., 2023), suggesting the effectiveness of multi-reward RL over using a single reward.

Is bidirectional QWK more effective than unidirectional QWK? In addition, we analyze whether our bidirectional QWK strategy is indeed more advantageous than the unidirectional QWK rewards for policy training. Note that SaMRL_{uniQ_T} and SaMRL_{uniQ_B} are the models in which Q_T and Q_B are applied in combination with MSE rewards, respectively. We observe overall higher performances when solely using the trait-wise QWK Q_T as R_Q (SaMRL_{uniQ_T}) than when only applying the batch-wise QWK,

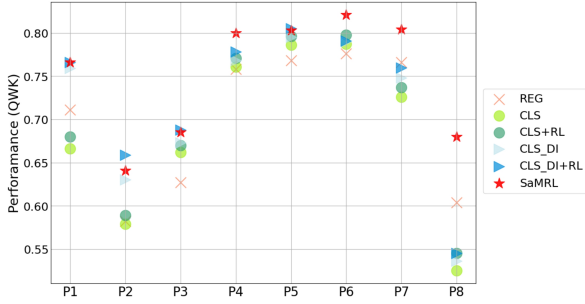


Figure 4: Comparison results of classification-based RL models and our SaMRL (★) for the *Overall* score prediction. CLS_{+RL} and CLS_{DI+RL} are models where RL is applied to CLS and CLS_{DI} , respectively.

Q_B (SaMRL_uni Q_B). As we hypothesized, using Q_B alone in a traditional essay-set-wise comparison can lead to unstable learning outcomes as it assigns the same reward to all in-batch samples. This phenomenon is also underscored by its degraded performance when compared to SaSRL $_M$, which exclusively uses MSE. Remarkably, introducing trait-set-wise unidirectional measurements Q_T alone brings in significant improvements over the baseline model and SaSRL $_M$, showing even comparable results to SaMRL $_{biQ}$. Nevertheless, the combined use of both directional calculations for the QWK reward stands as the most effective, implying their potential to complement each other.

6.2 Discussions

Comparison with classification-based RL Our method has shown effectiveness in essay sets evaluated in a wider score range, indicating the overcoming of limitations that exist in classification-based RL methods. To investigate our actual impacts compared to them, we compare SaMRL results with existing RL-based AES systems (Wang et al., 2018). Figure 4 illustrates two prior RL models: 1) bidirectional LSTM-based classification model, $CLS+RL$, and 2) dilated LSTM-based classification model, $CLS_{DI}+RL$. As they are holistic scoring models that predict a single *Overall* trait, the comparison is constrained to the *Overall* trait. Our SaMRL approach outperforms the prior models in most prompts, particularly showing significant improvements in prompts 7 and 8, which have a broader rating range of [0,30] and [0,60] than other prompts. Note that in these two prompts, even when RL is applied to classification, its performance is significantly inferior to regression (×). Meanwhile, our model, leveraging text generation probability while incorporating an awareness of the scoring schema,

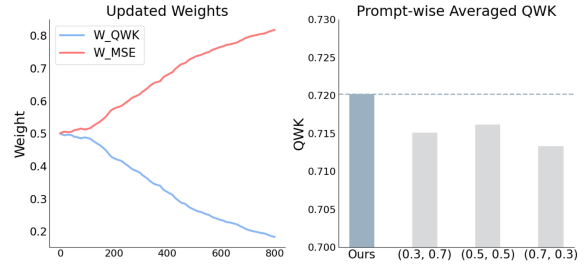


Figure 5: Variations in the updated weights of loss R_Q (W_{QWK}) and loss R_M (W_{MSE}) across training steps (left); comparison of prompt-wise averaged QWK performance between models with fixed weights and our SaMRL with trainable weights.

demonstrates significant robustness.

Impact of weight learning for multi-loss Following previous research showing that adjusting the weights of each loss adaptively in multi-task learning can improve performance (Chen et al., 2018; Kendall et al., 2018; Mao et al., 2022), we optimize multiple losses by dynamically learning the weights of each loss during the RL policy update process. By treating these weights as trainable parameters, we adaptively adjust the importance of each objective. As depicted in the left section of Figure 5, the assigned weights for each loss dynamically adjust throughout the training steps, with a growing emphasis on the MSE loss over the QWK loss as learning progresses. This shifting trend is consistent with findings from ablation studies, which indicate a greater impact of MSE reward on assisting the training of the AES model. Furthermore, our weight update mechanism has proved superior effectiveness compared to using fixed weights of (0.3, 0.7), (0.5, 0.5), and (0.7, 0.3) for loss R_Q and loss R_M , respectively, as reported in the right part of Figure 5. We anticipate that further optimization in multi-loss weight could bring in additional performance advances in future works.

7 Conclusion

In this work, we propose a Scoring-aware Multi-reward Reinforcement Learning (SaMRL) method, which incorporates the rating process for effective multi-trait scoring within the generation framework. By introducing the policy gradient reinforcement in the autoregressive score generation paradigm, we enable the direct use of the QWK metric; thus, SaMRL effectively captures the rating procedure. In addition, we jointly introduce the score-level

difference-based reward, MSE reward, which was previously limited to regression AES models, bringing in the overall enhancements on trait-wise and prompt-wise scoring qualities. Extensive experiments and analysis further support the assistance of our RL strategy over the simple generation.

Limitation

In this work, we grounded the method on autoregressive prediction, where prediction order may matter. Currently, we employ the same order as the previous work (Do et al., 2024); however, thoroughly considering the shifts in trait prediction order can lead to further improvements. Secondly, we update the policy at a time after the entire trait score prediction, motivated by existing approaches (Stiennon et al., 2020; Dutta et al., 2024; Ryu et al., 2024). However, training the policy with the instant updating per each action (i.e., token generation) might bring in more benefits in the case of the scoring task, which can be noteworthy for future work. In addition, we update the weights via training for the multiple losses. As in prior works, adaptive optimization strategies or more refined mechanisms could extend the impact of our method (Chen et al., 2018; Kendall et al., 2018; Mao et al., 2022).

Ethical Statement

Only publicly available datasets, ASAP, ASAP++, and Feedback Prize, are used in this work.

Acknowledgements

This work was partly supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.RS-2019-II191906, Artificial Intelligence Graduate School Program (POSTECH)), the MSIT (Ministry of Science and ICT) Korea, under the ITRC (Information Technology Research Center) support program (IITP-2024-2020-0-01789) supervised by the IITP, and Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2022-0-00223, Development of digital therapeutics to improve communication ability of autism spectrum disorder patients).

References

- Yen-Chun Chen and Mohit Bansal. 2018. [Fast abstractive summarization with reinforce-selected sentence rewriting](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 675–686, Melbourne, Australia. Association for Computational Linguistics.
- Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. 2018. [GradNorm: Gradient normalization for adaptive loss balancing in deep multitask networks](#). In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 794–803. PMLR.
- Jacob Cohen. 1968. Weighted kappa: nominal scale agreement provision for scaled disagreement or partial credit. *Psychological bulletin*, 70(4):213.
- Mădălina Cozma, Andrei Butnaru, and Radu Tudor Ionescu. 2018. [Automated essay scoring with string kernels and word embeddings](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 503–509, Melbourne, Australia. Association for Computational Linguistics.
- Christoph Dann, Yishay Mansour, and Mehryar Mohri. 2023. [Reinforcement learning can be more efficient with multiple rewards](#). In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 6948–6967. PMLR.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of deep bidirectional transformers for language understanding](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Heejin Do, Yunsu Kim, and Gary Lee. 2024. [Autoregressive score generation for multi-trait essay scoring](#). In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 1659–1666, St. Julian’s, Malta. Association for Computational Linguistics.
- Heejin Do, Yunsu Kim, and Gary Geunbae Lee. 2023. [Prompt- and trait relation-aware cross-prompt essay trait scoring](#). In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1538–1551, Toronto, Canada. Association for Computational Linguistics.
- Fei Dong and Yue Zhang. 2016. [Automatic features for essay scoring – an empirical study](#). In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1072–1077, Austin, Texas. Association for Computational Linguistics.

- Fei Dong, Yue Zhang, and Jie Yang. 2017. [Attention-based recurrent convolutional neural network for automatic essay scoring](#). In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 153–162, Vancouver, Canada. Association for Computational Linguistics.
- Yue Dong, Yikang Shen, Eric Crawford, Herke van Hoof, and Jackie Chi Kit Cheung. 2018. [Bandit-Sum: Extractive summarization as a contextual bandit](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3739–3748, Brussels, Belgium. Association for Computational Linguistics.
- Subhabrata Dutta, Ishan Pandey, Joykirat Singh, Sunny Manchanda, Soumen Chakrabarti, and Tanmoy Chakraborty. 2024. [Frugal lms trained to invoke symbolic solvers achieve parameter-efficient arithmetic reasoning](#). In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 17951–17959.
- Chulaka Gunasekara, Guy Feigenblat, Benjamin Szajder, Ranit Aharonov, and Sachindra Joshi. 2021. [Using question answering rewards to improve abstractive summarization](#). In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 518–526, Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Alex Havrilla, Yuqing Du, Sharath Chandra Rapparthi, Christoforos Nalmpantis, Jane Dwivedi-Yu, Maksym Zhuravinskyi, Eric Hambro, Sainbayar Sukhbaatar, and Roberta Raileanu. 2024. [Teaching large language models to reason with reinforcement learning](#). *Preprint*, arXiv:2403.04642.
- Zhiwei He, Xing Wang, Wenxiang Jiao, Zhuosheng Zhang, Rui Wang, Shuming Shi, and Zhaopeng Tu. 2024. [Improving machine translation with human feedback: An exploration of quality estimation as a reward model](#). In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8164–8180, Mexico City, Mexico. Association for Computational Linguistics.
- Alex Kendall, Yarin Gal, and Roberto Cipolla. 2018. [Multi-task learning using uncertainty to weigh losses for scene geometry and semantics](#). In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7482–7491.
- Rahul Kumar, Sandeep Mathias, Sriparna Saha, and Pushpak Bhattacharyya. 2022. [Many hands make light work: Using essay traits to automatically score essays](#). In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1485–1495.
- Zimu Lu, Aojun Zhou, Ke Wang, Houxing Ren, Weikang Shi, Junting Pan, Mingjie Zhan, and Hongsheng Li. 2024. [Step-controlled dpo: Leveraging stepwise error for enhanced mathematical reasoning](#). *Preprint*, arXiv:2407.00782.
- Yuren Mao, Zekai Wang, Weiwei Liu, Xuemin Lin, and Pengtao Xie. 2022. [MetaWeighting: Learning to weight tasks in multi-task learning](#). In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 3436–3448, Dublin, Ireland. Association for Computational Linguistics.
- 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)*.
- Sandeep Mathias and Pushpak Bhattacharyya. 2020. [Can neural networks automatically score essay traits?](#) In *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 85–91.
- Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. [Ranking sentences for extractive summarization with reinforcement learning](#). In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1747–1759, New Orleans, Louisiana. Association for Computational Linguistics.
- Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. [Training language models to follow instructions with human feedback](#). *Preprint*, arXiv:2203.02155.
- Jacob Parnell, Inigo Jauregi Unanue, and Massimo Piccardi. 2022. [A multi-document coverage reward for RELAXed multi-document summarization](#). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5112–5128, Dublin, Ireland. Association for Computational Linguistics.
- Ramakanth Pasunuru and Mohit Bansal. 2018. [Multi-reward reinforced summarization with saliency and entailment](#). In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 646–653, New Orleans, Louisiana. Association for Computational Linguistics.
- Romain Paulus, Caiming Xiong, and Richard Socher. 2018. [A deep reinforced model for abstractive summarization](#). In *Proceedings of the International Conference on Learning Representations*.
- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. [Exploring the limits of transfer learning with a unified text-to-text](#)

- transformer. *Journal of Machine Learning Research*, 21(140):1–67.
- Leonardo F. R. Ribeiro, Mohit Bansal, and Markus Dreyer. 2023. [Generating summaries with controllable readability levels](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 11669–11687, Singapore. Association for Computational Linguistics.
- Robert Ridley, Liang He, Xin-yu Dai, Shujian Huang, and Jiajun Chen. 2021. Automated cross-prompt scoring of essay traits. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 13745–13753.
- Paul Roit, Johan Ferret, Lior Shani, Roei Aharoni, Geoffrey Cideron, Robert Dadashi, Matthieu Geist, Serkan Girgin, Leonard Hussenot, Orgad Keller, Nikola Momchev, Sabela Ramos Garea, Piotr Stanczyk, Nino Vieillard, Olivier Bachem, Gal Elidan, Avinatan Hassidim, Olivier Pietquin, and Idan Szpektor. 2023. [Factually consistent summarization via reinforcement learning with textual entailment feedback](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6252–6272, Toronto, Canada. Association for Computational Linguistics.
- Sangwon Ryu, Heejin Do, Yunsu Kim, Gary Lee, and Jungseul Ok. 2024. [Multi-dimensional optimization for text summarization via reinforcement learning](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5858–5871, Bangkok, Thailand. Association for Computational Linguistics.
- John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. 2016. High-dimensional continuous control using generalized advantage estimation. In *Proceedings of the International Conference on Learning Representations*.
- John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. [Proximal policy optimization algorithms](#). *Preprint*, arXiv:1707.06347.
- Joykirat Singh, Sehban Fazili, Rohan Jain, and Md. Shad Akhtar. 2024. [EROS:entity-driven controlled policy document summarization](#). In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 6236–6246, Torino, Italia. ELRA and ICCL.
- Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. 2020. [Learning to summarize with human feedback](#). In *Advances in Neural Information Processing Systems*, volume 33, pages 3008–3021. Curran Associates, Inc.
- DiJia Su, Difei Su, John M. Mulvey, and H. Vincent Poor. 2023. [Optimizing multidocument summarization by blending reinforcement learning policies](#). *IEEE Transactions on Artificial Intelligence*, 4(3):416–427.
- Kaveh Taghipour and Hwee Tou Ng. 2016. A neural approach to automated essay scoring. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 1882–1891.
- 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. *arXiv preprint arXiv:2205.03835*.
- Yucheng Wang, Zhongyu Wei, Yaqian Zhou, and Xuanjing Huang. 2018. [Automatic essay scoring incorporating rating schema via reinforcement learning](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 791–797, Brussels, Belgium. Association for Computational Linguistics.
- Lijun Wu, Fei Tian, Tao Qin, Jianhuang Lai, and Tie-Yan Liu. 2018. [A study of reinforcement learning for neural machine translation](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3612–3621, Brussels, Belgium. Association for Computational Linguistics.
- Zhiheng Xi, Wenxiang Chen, Boyang Hong, Senjie Jin, Rui Zheng, Wei He, Yiwen Ding, Shichun Liu, Xin Guo, Junzhe Wang, Honglin Guo, Wei Shen, Xiaoran Fan, Yuhao Zhou, Shihan Dou, Xiao Wang, Xinbo Zhang, Peng Sun, Tao Gui, Qi Zhang, and Xuanjing Huang. 2024. [Training large language models for reasoning through reverse curriculum reinforcement learning](#). *Preprint*, arXiv:2402.05808.
- 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.

A Comparison models

We primarily compared our method with the baseline ArTS (Do et al., 2024) model, which is the previous state-of-the-art model for multi-trait AES. As we aim to examine the effects of applying RL on the autoregressive model, the comparison mainly focuses on the model with and without applying our method. In addition, we also report the results of other multi-trait scoring models (Kumar et al., 2022) and the holistic scoring models (Cozma et al., 2018; Dong et al., 2017) individually applied for each trait prediction. In particular, the multi-trait scoring MTL model (Kumar et al., 2022) constructed each trait-specific layer and used all other trait layers auxiliary for a target trait training and prediction. The holistic scoring

Model	Traits (Prediction Order: \leftarrow)											AVG
	Overall	Content	PA	Lang	Nar	Org	Conv	WC	SF	Style	Voice	
ArTS-base*	± 0.025	± 0.015	± 0.020	± 0.027	± 0.020	± 0.021	± 0.022	± 0.033	± 0.026	± 0.008	± 0.098	± 0.029
SaMRL-base (Ours)	± 0.014	± 0.015	± 0.019	± 0.024	± 0.018	± 0.019	± 0.017	± 0.027	± 0.021	± 0.007	± 0.114	± 0.027
ArTS-large*	± 0.011	± 0.014	± 0.032	± 0.026	± 0.015	± 0.011	± 0.016	± 0.005	± 0.009	± 0.033	± 0.080	± 0.023
SaMRL-large (Ours)	± 0.010	± 0.013	± 0.030	± 0.023	± 0.017	± 0.013	± 0.019	± 0.009	± 0.008	± 0.023	± 0.081	± 0.022

Table 7: The standard deviation of QWK results over five runs (main results in Table 2) averaged across the prompts for each **trait**. The *AVG* here denotes the mean of the trait-specific standard deviations, while *SD* in Table 2 refers to the standard deviation of the five-fold average.

Model	Prompts								AVG
	1	2	3	4	5	6	7	8	
ArTS-base*	± 0.010	± 0.034	± 0.044	± 0.020	± 0.018	± 0.011	± 0.007	± 0.089	± 0.029
SaMRL-base (Ours)	± 0.015	± 0.023	± 0.041	± 0.019	± 0.021	± 0.012	± 0.012	± 0.095	± 0.030
ArTS-large*	± 0.016	± 0.031	± 0.037	± 0.024	± 0.029	± 0.025	± 0.025	± 0.050	± 0.030
SaMRL-large (Ours)	± 0.020	± 0.029	± 0.037	± 0.025	± 0.030	± 0.021	± 0.022	± 0.057	± 0.030

Table 8: The standard deviation of QWK results over five runs (Table 3) averaged across the traits for each **prompt**.

Model	Traits (Prediction Order: \leftarrow)											AVG
	Overall	Content	PA	Lang	Nar	Org	Conv	WC	SF	Style	Voice	
ArTS-base*	± 0.025	± 0.015	± 0.020	± 0.027	± 0.020	± 0.021	± 0.022	± 0.033	± 0.026	± 0.008	± 0.098	± 0.029
SaSRL _M	± 0.008	± 0.015	± 0.021	± 0.026	± 0.021	± 0.020	± 0.017	± 0.034	± 0.021	± 0.010	± 0.098	± 0.026
SaSRL _Q	± 0.024	± 0.016	± 0.022	± 0.029	± 0.028	± 0.019	± 0.016	± 0.030	± 0.024	± 0.015	± 0.077	± 0.027
SaMRL _{uniQ_T}	± 0.027	± 0.015	± 0.017	± 0.025	± 0.020	± 0.022	± 0.020	± 0.032	± 0.026	± 0.010	± 0.098	± 0.028
SaMRL _{uniQ_B}	± 0.026	± 0.014	± 0.022	± 0.030	± 0.027	± 0.016	± 0.016	± 0.032	± 0.020	± 0.010	± 0.109	± 0.029
SaMRL_Q (Ours)	± 0.014	± 0.015	± 0.019	± 0.024	± 0.018	± 0.019	± 0.017	± 0.027	± 0.021	± 0.007	± 0.114	± 0.027

Table 9: The standard deviation of QWK results over five runs (ablation study results in Table 4) averaged across the prompts for each **trait**. The *AVG* here denotes the mean of the trait-specific standard deviations, while *SD* in Table 4 refers to the standard deviation of the five-fold average.

Model	Prompts								AVG
	1	2	3	4	5	6	7	8	
ArTS-base*	± 0.010	± 0.034	± 0.044	± 0.020	± 0.018	± 0.011	± 0.007	± 0.089	± 0.029
SaSRL _M	± 0.018	± 0.027	± 0.044	± 0.019	± 0.016	± 0.012	± 0.012	± 0.087	± 0.029
SaSRL _Q	± 0.017	± 0.032	± 0.050	± 0.022	± 0.013	± 0.016	± 0.012	± 0.078	± 0.030
SaMRL _{uniQ_T}	± 0.010	± 0.033	± 0.040	± 0.022	± 0.020	± 0.010	± 0.008	± 0.090	± 0.029
SaMRL _{uniQ_B}	± 0.019	± 0.035	± 0.049	± 0.021	± 0.021	± 0.014	± 0.012	± 0.089	± 0.033
SaMRL_Q (Ours)	± 0.015	± 0.023	± 0.041	± 0.019	± 0.021	± 0.012	± 0.012	± 0.095	± 0.030

Table 10: The standard deviation of QWK results over five runs (Table 5) averaged across the traits for each **prompt**.

model, HISK, utilizes a support vector regressor paired with a histogram intersection string kernel, whereas STL-LSTM models use an LSTM-CNN-based structure; each model is iteratively deployed for independent trait scoring task. Except for the implemented ArTS*, results of other models are reported from the previous works (Kumar et al., 2022; Do et al., 2024).

standard deviation for the main results (Table 7, 8) and the results of ablation studies (Table 9, 10).

B Standard deviations for five-fold validations

In this work, we implemented five-fold validation and presented scores averaged over the five folds in all experiments. In this session, we report the