Rubric-Specific Approach to Automated Essay Scoring with Augmentation Training

Anonymous ACL submission

Abstract

Neural based approaches to automatic evaluation of subjective responses have shown superior performance and efficiency compared to traditional rule-based and feature engineering oriented solutions. However, it remains unclear whether the suggested neural solutions are sufficient replacements of human raters as we find recent works do not properly account for rubric items that are essential for automated essay scoring during model training and validation. In this paper, we propose a series of data augmentation operations that train and test an automated scoring model to learn features and functions overlooked by previous works while still achieving state-of-the-art performance in the Automated Student Assessment Prize dataset.

1 Introduction

Automated Essay Scoring (AES) is defined as the task of applying automation algorithms to evaluate the quality of written essay responses without the intervention of a human grader. Recently, neural applications involving deep neural networks and representation learning quickly proved to be flexible and effective (Ramesh and Sanampudi, 2021) in AES. Consequently, series of neural based approaches to AES including recurrent neural networks (Taghipour and Ng, 2016), attention mechanism (Dong et al., 2017), and pre-trained language models (Yang et al., 2020; Jeon and Strube, 2021) have been researched and tested on the Automated Student Assessment Prize (ASAP) dataset.[1]

However, the ongoing AES performance competition overlooks several critical problems. Specifically, multiple attributes commonly found from scoring rubrics are left out from consideration during AES model training and validation. Instead of evaluating the AES model's alignment with items outlined in the rubric, previous neural approaches to ASAP focus on achieving state-of-the-art similarity scores between a human rater and an AES model. While similarity score is one important aspect of functioning AES systems, it alone does not guarantee that an AES model can replace a human rater (Bennett and Bejar, 1997; Attali, 2007; Zhang, 2013; Perelman, 2014; Madnani and Cahill, 2018). Therefore, previous neural approaches must be evaluated for additional AES functions other than similarity (Kabra et al., 2022) before being deployed for service.

A deeper investigation into previous works reveals a potential source for the aforementioned problem. One common trait shared by previous neural approaches to ASAP is the implementation of prompt-specific models (Attali and Burstein, 2005; Ridley et al., 2021). The approach is defined by how the training dataset is segmented. Given a dataset comprised of essays to n question prompts, prompt-specific approach segments the dataset into n subsets based on prompt (prompt-segmented dataset) and trains one AES model on each subset, resulting in n prompt-specific models for n question prompts even when the prompts share the same rubric. The n models train to learn the same scoring rubric, but the scoring standard learned by each model will likely diverge as the model optimizes on each data segments (Attali et al., 2010; Chen and He, 2013), resulting in models that no longer embody the original scoring rubric. Moreover, segmenting the dataset based on features eliminates the need for AES models to account for those features during training and validation. For instance, once the dataset is segmented based on question prompts, the AES model is never tested on its ability to assess relevance of essay responses in relation to varying question prompts. Similarly, once the training dataset is segmented based on features relating to a specific rubric item, the resulting AES model will not be able to account for the rubric item during training and inference (Madnani and

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

Cahill, 2018).

In this research, we propose an alternative approach to AES termed rubric-specific model. Rubric-specific models are trained and tested on datasets segmented by scoring rubrics (rubric-segmented dataset), resulting in n rubric-specific models for n scoring rubrics. Each rubric-specific model is trained to be the best and only representation of it's respective scoring rubric regardless of how many prompts are tied to the same rubric. The proposed approach is general and efficient as it is aligned with human raters who are trained to learn each scoring rubric instead of each question prompt. Most importantly, since rubric-segmented datasets include features precluded in prompt-segmented datasets, rubric-specific models must consider the following rubric items overlooked by prompt-specific models:

- Rubric-segmented dataset includes essay responses to multiple question prompts. Therefore, rubric-specific models must be able to distinguish various response-prompt combinations and assess the relevance of an essay in relation to the question prompt. Relevance assessment is not only essential in essay scoring, but is also crucial in AES service application. For instance, the inability to detect and evaluate irrelevant responses leaves the AES model unprepared against adversarial attacks and could potentially debunk the effectiveness and reliability of AES systems in their entirety (Ding et al., 2020; Kabra et al., 2022).

- Rubric-segmented dataset includes essay responses written by students from varying grade levels. Along with individual writing skills, student grade level is also a significant predictor of essay scores (Burdick et al., 2013). Therefore, the quality of writing a human rater expects from a well-written essay should be different and adjusted based on the student's grade level. Similar to a human rater, a rubric-specific model must be able to identify and incorporate grade level differences in automated scoring (Zhang, 2013).

In addition to the previously precluded factors, our research seeks to address another rubric item that is fundamental to essay scoring, yet insufficiently investigated during the performance competition at ASAP:

- Human raters are expected to detect and penalize incoherently ordered words or sentences. However, the same cannot be expected from neural AES systems (Pham et al., 2020). Consequently, rubric-specific models must be equipped with and tested on the ability to penalize permuted text and distinguish adversarial inputs (Farag et al., 2018; Ding et al., 2020; Singla et al., 2021; Kabra et al., 2022).

Our experiment demonstrates training an AES model to learn the above rubric items while maintaining significant similarity score requires more than simply training on a rubric-segmented dataset. We introduce three data augmentation methods, $Prompt\ Swap$, $Grade\ Match$, and $Response\ Distortion$, to guide the AES model to learn the intended features without suffering from robustness-accuracy trade-off (Su et al., 2018). Moreover, we introduce a neural network architecture, $Response - Prompt\ AES$, capable of processing the suggested augmentation training. Our experiment results show the proposed augmentation methods resolve the functional limitations of previous neural approaches to AES. In addition to the added functions, we also demonstrate our proposed AES model achieves state-of-the art performance in the ASAP dataset.

2 Related Work

Neural AES Neural approaches to AES adopted learned representations such as pre-trained word vectors (Taghipour and Ng, 2016; Mathias et al., 2020) and contextual embeddings from pre-trained language models (Yang et al., 2020; Jeon and Strube, 2021; Xue et al., 2021) to replace conventional handcrafted features utilized in AES. In addition to learned features, recent works experimented with training strategies such as multi-task learning (Muangkammuen and Fukumoto, 2020; Yang et al., 2020; Mathias et al., 2020) to achieve enhanced performance in the ASAP dataset.

Generic AES While neural applications in AES proved to be effective, prompt-specific AES models required large amounts of labeled training data and were limited to scoring essays from only one question prompt. To address the problems of efficiency, researches including Jin et al. (2018) and Ridley et al. (2021) proposed a prompt-independent approach to AES utilizing essay responses from multiple prompts for AES model training. However, while generic AES model training involved essay responses to multiple question prompts, the topic of irrelevant responses or adversarial inputs was never properly discussed.

Robust AES The performance race at ASAP sparked another important discussion in AES. Multiple researches raised questions regarding the robustness of neural AES systems. According to related works, state-of-the-art AES models were unable to detect essays with randomly shuffled word ordering (Farag et al., 2018; Ding et al., 2020), off-topic content (Liu et al., 2019; Kabra et al., 2022), and abnormal inputs (Perelman, 2014). While related works proposed various adversarial training methods as potential remedies, increase in AES model robustness was accompanied with loss in accuracy, architecture overhead, and added optimization tasks (Liu et al., 2019; Ding et al., 2020; Sun et al., 2022).

3 Response-Prompt AES

We first introduce the neural network architecture implemented in our experiments and describe the computation flow and reasoning behind the model structure. Our AES model includes a pre-trained language model, a response self-attention layer, and a response-prompt attention layer which are all fine-tuned on the ASAP dataset. Implementation details on each module are listed below in order of computation.

Pre-trained Language Model To generate contextual embeddings from essays and prompts, we utilize the widely successful pre-trained language model BERT (Devlin et al., 2019) and its implementation (bert-base-uncased) in the Python language.[2] Essay responses and question prompts are tokenized into list of tokens and used as inputs to BERT. To address the maximum token length restriction imposed on BERT, token lists longer than 512 are segmented and stacked into token groups of size 512. After forward passing through BERT, we collapse the segment axis of the output embedding matrix.

Response Self-Attention Layer The collapsed embedding matrix passes through a custom designed self-attention layer without predefined length restriction. The response self-attention layer implements self-attention (Vaswani et al., 2017) with relative position embeddings (Shaw et al., 2018) to simulate human reading pattern on long texts with multiple sentences and paragraphs. Response self-attention is computed as follows:

$$e_{ij} = \frac{x_i W^Q (x_j W^K)^T + x_i W^Q (R_{ij}^K)^T}{\sqrt{d}} \quad (1)$$

$$a_{ij} = \frac{\exp e_{ij}}{\sum_{k=1}^{n} \exp e_{ik}} \quad (2)$$

$$z_i = \sum_{j=1}^{n} a_{ij} (x_j W^V + R_{ij}^V) \quad (3)$$

where $\{x_1, x_2, ..., x_n\}$ is the embedding matrix output from BERT, W^Q, W^K, W^V trainable parameters from the attention layer, and R^K and R^V relative position representation matrices also trained during the fine-tuning process.

Finally, the contextualized embedding vector from equation 3 corresponding to the CLS token position index, z_1, is used as essay response representation.

Response-Prompt Attention Layer Attention mechanism (Bahdanau et al., 2015) is implemented in the response-prompt attention layer. Essay response representation matrix $\{z_1, z_2, ..., z_n\}$ attends the prompt embedding matrix P to compute response-prompt attention vector as shown below.

$$e_{ij} = \frac{z_i W^Q (P_j W^K)^T}{\sqrt{d}} \quad (4)$$

$$a_{ij} = \frac{\exp e_{ij}}{\sum_{k=1}^{n} \exp e_{ik}} \quad (5)$$

$$r_i = \sum_{j=1}^{n} a_{ij} P_j W^V \quad (6)$$

The resulting response-prompt attention vector corresponding to the CLS token position index, r_1, is used as response-prompt relevance representation.

Regression Layer Lastly, the essay response representation vector z_1 is concatenated with the response-prompt attention vector r_1 to form the final representation vector used for score prediction. The concatenated vector passes through a dense layer to compute the model output, $\hat{o}$, as shown in equation 7.

$$\hat{o} = concat(z_1, r_1)W + b \quad (7)$$

[2] https://github.com/huggingface/transformers

The Response-Prompt AES model is trained with the Mean Squared Error (MSE) loss function. Specifically, given training label score o_i, the model is trained to minimize the following loss function over the training dataset:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(o_i - \hat{o}_i)^2 \tag{8}$$

4 Experiment

In this section, we outline the details of training a rubric-specific model. Specifically, we describe our experimental procedure for implementing Prompt Swap, Grade Match, and Response Distortion on the Response-Prompt AES model. Our experiment is focused on the following investigations:

- Measure the isolated effect of each data augmentation method and establish it's functional significance in AES.
- Assess the isolated and combined effect of data augmentation methods on the ASAP dataset against benchmark performance.

Hyper-parameter settings for all training and testing experiments are summarized in Appendix B. Our source code and experiment logs are publicly available for review and replication.[3]

4.1 Dataset

Our experiment uses the Automated Student Assessment Prize dataset from Kaggle. This dataset includes essay responses to eight different question prompts, and each essay response is labeled with an evaluation score given by a human grader according to the prompt's respective scoring rubric. Statistical summary and metadata of the dataset are provided in Tables 5 and 6 of Appendix A, respectively.

Aligned with the definition of rubric-specific models, we segment the dataset into six subsets corresponding to the number of unique scoring rubrics and train one AES model from each subset. For easier comparison, our six AES models are labeled Prompt 1, 2, 7, 8, 3-4, and 5-6 model.

4.2 Performance Evaluation

For performance assessment in the ASAP dataset, we use 5-fold cross validation to evaluate our AES model with 60% / 20% / 20% data split amongst training, develop, and test sets. Fold indices are adopted from Taghipour and Ng (2016). All of the selected performance benchmarks listed in the Results and Analysis section follow the same fold indices for accurate performance comparison. Consistent with previous works, we select our best AES model based on the performance in the develop set and adopt **Quadratic Weighted Kappa** (QWK) [4] as evaluation metric.

For performance assessment of each data augmentation method, we employ specific metrics further explained in the following subsections. Data augmentation performances are also reported in averages computed over 5 folds.

4.3 Data Augmentation Implementation

4.3.1 Prompt Swap

The fundamental fact that an essay's score is dependent on the question prompt is often overlooked. For example, an essay response to question prompt 3 that received a perfect score is no longer considered relevant when paired with question prompt 4 regardless of writing quality. The idea of relevance is also embedded in the ASAP scoring rubric which is shown in Table 7 of Appendix A. While distinguishing essays to prompt 3 from essays to prompt 4 is easy task for human raters, the same cannot be expected from AES systems. Accordingly, we apply Prompt Swap to Prompt 3-4 and 5-6 models to train relevance aware AES models.

Prompt Swap generates prompt mismatched essay samples with known labels for augmentation training. For a given training batch $b = \{t_1, t_2, ..., t_n\}$ where $t_i = \{essay, prompt, score\}$, we select k samples from the training batch, swap the prompt to mismatch the essay response, and add the generated irrelevant response-prompt sample to the training batch with known label of score zero (lowest possible score). For example, if $s = \{essay_4, prompt_4, score = 3\}$ is a relevant response-prompt sample addressing prompt 4 with a perfect score, the AES model should also be able to train from and accurately predict irrelevant response-prompt sample such as $s' = \{essay_4, prompt_3, score = 0\}$. When selecting the k samples for augmentation, Prompt Swap is only applied to essay responses with original label scores greater than the average score. Such con-

[3] Anonymized URL

[4] https://www.kaggle.com/competitions/asap-aes/overview/evaluation

dition is necessary as low score essays have low writing quality regardless of relevance, making it difficult to isolate the effect of Prompt Swap.

The contribution of Prompt Swap is reported in two folds. First, **irrelevant response detection rate** is measured with prompt swapped samples generated from the test set. The AES model should predict the lowest label score for the prompt swapped samples, which we count as detection success. All other score predictions are counted as detection failures. Irrelevant response detection rate is computed and compared against baseline models trained without Prompt Swap. Second, we investigate whether Prompt Swap improves both robustness and accuracy of AES models by comparing test set QWK against baseline models trained without Prompt Swap.

4.3.2 Grade Match

Unlike Prompts 3 and 4 which are written by students in the same grade level, Prompts 5 and 6 are written by students in different grade levels as shown in Table 6 of Appendix A. Therefore, we hypothesized that while essay responses to prompts 5 and 6 are graded with the same scoring rubric, a human rater must adjust the expectation for a well-written essay based on the student's grade level. Accordingly, we apply Grade Match in Prompt 5-6 model to not only differentiate scores, but also differentiate essays from different grade levels.

The process of recognizing differences between essays is analogous to training an AES model to learn distances between essay representations in the embedding space. Particularly, Grade Match seeks to map essays from the same grade level close together while distancing them from essays from other grade levels. Grade Match is inspired by the methodologies proposed in Supervised Contrastive Learning (Khosla et al., 2020), which leverages label information to contrast batch items from one class against batch items from another class. Following the same strategy, Prompt 5-6 model utilizes score and grade level as labels during Grade Match.

Given essay batch $E = \{e_1, e_2, ..., e_n\}$, corresponding score label $S = \{s_1, s_2, ..., s_n\}$, corresponding grade level label $G = \{g_1, g_2, ..., g_n\}$, and augmentation sample count k, the essay response representation set $Z = \{z_1, z_2, ..., z_n\}$ is calculated for each corresponding batch item in E according to Equation 3. Next, cosine similarity c_s is calculated for batch items with the same score and same grade level as follows.

$$c_s = \sum_{\substack{g_i=g_j\in G,\\ i\neq j}} \sum_{\substack{s_i=s_j\in S,\\ i\neq j}} Cos(z_i, z_j) \tag{9}$$

Similarly, cosine similarity c_d is calculated for batch items with the same score but different grade level and batch items with the same grade level but different score as follows.

$$\begin{aligned} c_d = &\sum_{g_i\neq g_j\in G} \sum_{\substack{s_i=s_j\in S,\\ i\neq j}} Cos(z_i, z_j) \\ &+ \sum_{\substack{g_i=g_j\in G,\\ i\neq j}} \sum_{s_i\neq s_j\in S} Cos(z_i, z_j) \end{aligned} \tag{10}$$

Finally, Prompt 5-6 model is trained to minimize the following loss function which incorporates both the MSE from Equation 8 and cosine similarities computed in Equations 9 and 10.

$$L = MSE - \frac{1}{k}(c_s - c_d) \tag{11}$$

The contribution of Grade Match is measured with test set QWK, and the results are compared against baseline models trained without Grade Match.

4.3.3 Response Distortion

A service ready AES model must be able to detect and penalize incoherent word ordering. Our experiment investigates whether supervised training on the ASAP dataset alone leads to such results. Moreover, we experiment with a data augmentation method, Response Distortion, for adversarial training. Since adversarial input detection is applicable to all scoring rubrics, Response Distortion is applied to all six AES models.

Response Distortion generates a partially permuted essay sample from a normal essay response. Compared to similar works in AES adversarial training, Response Distortion is unique in that it only augments essays with a particular label score. For a given training batch $b = \{t_1, t_2, ..., t_n\}$, we first filter essay samples with the lowest label score to get $b' = \{t'_1, t'_2, ..., t'_m\}$ where $t'_i = \{essay, prompt, score = 0\}$. Next, we randomly select maximum of k samples from the filtered set b' for Response Distortion. For each selected k sample, we count the number of words w in the essay and select two indices i and j such that $0 \leq i < j \leq w$. Finally, we randomly permute the ordering of all words between the ith and jth word index of the essay and add the generated distorted sample to the training batch with known label of score zero (lowest possible score). Since essays from b' are already assigned with the lowest label score, any distortions that further lowers the quality of writing will not change the assigned label score. Following the same logic, Response Distortion is also applied to prompt mismatched samples generated from Prompt Swap to introduce to the AES model various types of traits shared by low quality essay responses.

The contribution of Response Distortion is reported in two folds. First, **distorted response detection rate** is measured with distorted samples generated from the test set. Unlike the training stage in which Response Distortion is only applied to essays with the lowest label score, distortion is applied to essays without score condition during testing. Moreover, since Response Distortion only applies partial permutation to word ordering, we cannot assign the lowest label score to distorted test samples with certainty. Therefore, given normal essay t, distorted essay t', and AES model $f(essay) \rightarrow score$, distorted response detection is successful when $f(t) > f(t')$ (Kabra et al., 2022). Distorted response detection rate is computed and compared against baseline models trained without Response Distortion. Second, in response to previous work that reported a trade-off between anomaly detection rate and QWK performance (Ding et al., 2020), we test how Response Distortion affects test set QWK and compare the results against baseline models trained without Response Distortion.

5 Results and Analysis

In this section, we analyze and discuss our experiment findings. Specifically, we evaluate the performance metric of each data augmentation method against baseline performances and examine the contribution of each method in terms of AES application. Moreover, we test a Response-Prompt AES model trained with our proposed data augmentations on the ASAP dataset and investigate how the results compare against those of previous neural approaches. Statistical significance is computed using paired t-tests between augmentation and baseline results. Statistical significance is denoted by · for $p < 0.1$, * for $p < 0.05$ and ** for $p < 0.01$.

5.1 Data Augmentation Results

5.1.1 Prompt Swap

Experiment result for Prompt 3-4 and Prompt 5-6 models trained with Prompt Swap is compared against two baseline models using irrelevant response detection rate as performance metric. The first baseline implements the Response-Prompt AES model structure and includes a response-prompt attention layer. However, the first baseline model is trained without Prompt Swap. The second baseline is a replicated model with the same model structure implemented in previous research, which consists of a regression layer attached to a pre-trained language model. Consistent with previous works, the second baseline model does not use the question prompt as input and is trained without Prompt Swap. Irrelevant response detection test results are summarized in Table 1.

Baseline	Response Prompt Attention	Irrelevant Response Detection Rate 3-4	Irrelevant Response Detection Rate 5-6
w/o Prompt Swap	No	2.5%	0.0%
w/o Prompt Swap	Yes	2.5%	0.0%
w/ Prompt Swap	Yes	**100%****	**100%****

Table 1: Mean test set irrelevant response detection rate reported in averaged percentages over 5 folds.

Experiment results indicate both baseline models trained without Prompt Swap fail to detect irrelevant responses. In other words, baseline models predict non-zero points to completely irrelevant essays. In contrast, Response-Prompt AES model trained with Prompt Swap records perfect detection rate in the test set. The results also demonstrate that the response-prompt attention layer is only relevant when implemented with Prompt Swap.

Furthermore, we analyze the response-prompt attention scores computed during Prompt Swap to investigate what is being learned by the AES model. Our findings summarized in Table 8 of Appendix C show the learned attention mechanism closely resembles the decision making process of a human rater.

5.1.2 Grade Match

Experiment result for Prompt 5-6 model trained with Grade Match is compared against baseline model using QWK as performance metric. The baseline model implements the Response-Prompt AES model structure but is trained without Grade Match. Grade Match test results are summarized in Table 2.

Baseline	QWK Prompt 5 (Grade 8)	QWK Prompt 6 (Grade 10)
w/o Grade Match	0.818	0.823
w/ Grade Match	**0.829·**	**0.837·**

Table 2: Mean test set QWK for Prompt 5 (Grade 8) and Prompt 6 (Grade 10) computed over 5 folds. Performance levels are computed from test set segmented by grade level for better comparison.

Baseline	Distort Rate	Response Distortion Detection Rate 1	2	3-4	5-6
w/o R.D.	25%	38.5%	42.8%	21.8%	10.3%
w/ R.D.	25%	**41.6%**	**44.8%**	**43.0%***	**24.7%***
w/o R.D.	50%	39.8%	65.2%	32.2%	17.7%
w/ R.D.	50%	**43.9%**	**74.2%**	**75.6%****	**51.6%***
w/o R.D.	100%	60.1%	100.0%	51.6%	26.8%
w/ R.D.	100%	**65.3%**	**100.0%**	**97.5%****	**83.8%****

Table 3: Mean test set distorted response detection rate reported in averaged percentages over 5 folds. Distort Rate is set during testing to control the level of permutation. For example, when Distort Rate is 50%, permutation indices i and j are sampled to cover 50% of the original response.

When an AES model trains from essay responses written by students from different grade levels, our experiment results indicate applying Grade Match leads to QWK improvements in both grade levels. Moreover, our results confirm student grade level is indeed a factor to be considered during rubric-specific model training.

5.1.3 Response Distortion

Experiment result for Prompt 1, 2, 3-4, and 5-6 models trained with Response Distortion is compared against baseline models using distorted response detection rate as performance metric (Response Distortion is not applicable in Prompt 7 and 8 models due to lack of lowest label score data in each rubric-segmented dataset). All baseline models implement the Response-Prompt AES model structure but are trained without Response Distortion. Test results summarized in Table 3 clearly indicate AES models trained with Response Distortion record higher detection rates than their baseline counterparts for both partial and whole permutations.

Test results from Prompt 1 model show the contribution of Response Distortion is relatively small in datasets with smaller portion of lowest label score samples (*See*, Table 5 of Appendix A). However, test results from Prompt 3-4 and 5-6 models suggest having more samples for augmentation does not guarantee linear increase in Response Distortion contribution. Lastly, Prompt 2 model shows even when the vanilla model performs well against full permutation, Response Distortion is still effective when processing partial permutations.

5.2 ASAP Performance

So far, experiment results indicate a Response-Prompt AES model trained with our proposed augmentation is equipped with functions necessary to handle rubric items overlooked by previous neural approaches to ASAP. Nonetheless, since QWK is still a key component of AES assessment, we investigate the relationship between the added functions and the AES model's performance on the ASAP dataset.

We evaluate six AES models corresponding to six unique rubrics provided in ASAP and compare the results against benchmark performances in Table 4. Evaluation result reveals the following: a Response-Prompt AES model trained with Prompt Swap, Grade Match, and Response Distortion record the highest average QWK on the ASAP dataset when compared to previous neural based and state-of-the-art approaches ($0.797 > 0.794$). Performance comparison analysis at the model level exhibits the following strengths and areas for improvements of our proposed method.

Prompt 1 & Prompt 2 Models Rows 7 and 8 of Table 4 indicate adding Response Distortion results in QWK performance gain in both prompts 1 and 2. Such finding goes against previous studies reporting a performance trade-off (Ding et al., 2020) between distorted response detection rate and QWK. As described in the experiment procedures, Response Distortion is different from related works in that it only augments essays with the lowest label score during training. The score condition is essential as it resolves the ambiguous task of assigning a score label to the generated sample without compromising overall label consistency. In line with related works, we confirm removing the score condition and extending Response Distortion to all score labels results in QWK performance loss.

Prompt 7 & Prompt 8 Models Response Distortion cannot be applied to datasets with insufficient number of low-score samples. Therefore, Prompt 7 and 8 model training is conducted without augmentations. Without augmentations, QWK performance in prompts 7 and 8 can be attributed

Row	AES Model	ASAP Prompt ID								Average
		1	2	3	4	5	6	7	8	
1	Taghipour and Ng (2016)	0.821	0.688	0.694	0.805	0.807	0.819	0.808	0.644	0.761
2	Dong et al. (2017)	0.822	0.682	0.672	0.814	0.803	0.811	0.801	0.705	0.764
3	Yang et al. (2020)	0.817	**0.719**	0.698	**0.845**	**0.841**	**0.847**	0.839	0.744	0.794
4	Muangkammuen and Fukumoto (2020)	**0.833**	0.685	0.690	0.795	0.812	0.816	0.798	0.673	0.763
5	Mathias et al. (2020)	**0.833**	0.681	0.698	0.818	0.815	0.821	0.806	0.699	0.771
6	Jeon and Strube (2021)	0.828	0.706	0.694	0.827	0.806	0.820	0.838	**0.769**	0.786
7	Response-Prompt AES	0.823	0.707	0.695	0.816	0.818	0.823	**0.842**	**0.763**	0.786
8	Response Distortion	**0.830**·	**0.719***	0.699	0.821·	0.823	0.827	-	-	-
9	Prompt Swap	-	-	0.702	0.830*	0.823	0.829	-	-	-
10	Prompt Swap + Response Distortion	-	-	**0.716****	**0.832****	0.824	0.834·	-	-	-
11	Grade Match	-	-	-	-	0.829·	0.837·	-	-	-
12	Prompt Swap + Response Distortion + Grade Match	-	-	-	-	**0.833***	**0.839**·	-	-	-
13	Response-Prompt AES + Best Augmentations	0.830	**0.719**	**0.716**	0.832	0.833	0.839	**0.842**	0.763	**0.797**

Table 4: Test set QWK performance for Response-Prompt AES model trained without augmentation (row 7), Response-Prompt AES model trained with various combinations of augmentations (rows 8-13), and AES models proposed in related works (rows 1-6). Augmented samples are only utilized during training and not included in test set QWK computation.

to the Response-Prompt AES model structure as shown in Row 7 of Table 4. Scoring rubric for prompt 7 deducts points based on the essay's focus on the topic.[5] Compared to benchmark models that only utilize the essay as input, Response-Prompt AES model utilizes both the essay and prompt as inputs and measures the essay's congruence with the prompt via response-prompt attention. Prompt 8 includes long essays that cannot be processed by previous approaches that inherit the 512 token length restriction from BERT. Response-Prompt AES model trains a self-attention layer without input length restriction to process longer essays and achieve better performance. However, Jeon and Strube (2021) suggests adopting a pre-trained language model without length restriction can also be an alternative to training a custom layer from scratch.

Prompt 3-4 Model While 18% of essays written in response to prompt 4 have zero label scores, only 2% of essays in prompt 3 have zero label scores, which makes low-score predictions particularly difficult in prompt 3. However, Prompt 3-4 model (i.e., rubric-specific model) is resilient to the label imbalance problem as it has access to zero label data from both prompts 3 and 4. Therefore, consistent with our findings from distorted response detection, we expect having access to sufficient number of zero label data will be an advantage for Prompt 3-4 model during augmentation training. Rows 8, 9, and 10 of Table 4 not only demonstrate the individual effect of each augmentation, but also show Prompt Swap complements Response Distortion by generating additional low score samples for distortion, resulting in QWK improvement especially in prompt 3.

Prompt 5-6 Model Rows 7 and 11 of Table 4 confirm our hypothesis regarding rater expectation of writing quality and student grade level. Despite the QWK performance gain from Grade Matching, Prompt 5-6 model is outperformed by Yang et al. (2020) in both prompts 5 and 6. The results are aligned with the idea that prompt-specific models, when compared to generic models, are optimized to be the better performing model for a given prompt (Chen and He, 2013). However, our results also indicate in exchange for prompt-specific performance, rubric-specific models benefit from efficiency (Attali and Burstein, 2006) as summarized in Table 9 of Appendix C.

6 Conclusion

In this paper, we seek to resolve the limitations of prompt-specific models while maintaining notable performance in the ASAP dataset. As a solution, we propose rubric-specific model training, which consists of a custom designed AES model trained from rubric-segmented datasets with series of data augmentations called Prompt Swap, Grade Match, and Response Distortion. Finally, we show the resulting AES model is capable of irrelevant response detection, student grade level adjustment, and distorted response detection while achieving state-of-the art performance in the ASAP dataset.

7 Limitation

Throughout this research, we identified several limitations relating to the performance metric and dataset utilized in the experiment.

[5] https://www.kaggle.com/competitions/asap-aes/data

First, while our research evaluates statistical significance among internal experiment results, statistical significance test was not applicable against external benchmark performances due to unavailability in released source code and limitations in replication.

Second, the ability to detect irrelevant response is a fundamental and expected feature of AES systems. Nevertheless, our experiments have demonstrated that the ASAP dataset and QWK do not test such fundamental attributes of AES models. Moreover, QWK provides limited information regarding the AES model's performance in other expected features such as fact checking or negation detection. Accordingly, while our experiment attains notable QWK performance in the ASAP dataset, we have insufficient understanding of our AES model's expected behavior against various data augmentation methods likely to be observed during real-world application.

Lastly, the QWK performance metric may not be aligned with the purpose of real-world application of autograding systems. Given that student performance in any academic field is mostly populated around the average, accurate evaluation is essential to identify the relatively smaller population of students who are falling behind or displaying talent. QWK is not an ideal performance metric for this purpose as capturing the majority around the average is a better strategy than capturing the small groups at both ends of the performance grid.

The ASAP dataset is one of many problems needed to be solved before real-world application of autograding systems. The limitations described above will be further discussed in our future work on AES focusing on service applications.

References

Yigal Attali. 2007. Construct validity of e-rater® in scoring toefl® essays. *ETS Research Report Series*, 2007(1):i–22.

Yigal Attali, Brent Bridgeman, and Catherine Trapani. 2010. Performance of a generic approach in automated essay scoring. *The Journal of Technology, Learning and Assessment*, 10(3).

Yigal Attali and Jill Burstein. 2005. Automated essay scoring with e-rater® v. 2.0. research report. ets rr-04-45. *ETS Research Report Series*.

Yigal Attali and Jill Burstein. 2006. Automated essay scoring with e-rater® v.2. *The Journal of Technology, Learning and Assessment*, 4(3).

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Randy Elliot Bennett and Isaac I Bejar. 1997. Validity and automated scoring: It's not only the scoring. *ETS Research Report Series*, 1997(1):i–30.

Hal Burdick, Carl W Swartz, A Jackson Stenner, Jill Fitzgerald, Don Burdick, and Sean T Hanlon. 2013. Measuring students' writing ability on a computer-analytic developmental scale: An exploratory validity study. *Literacy Research and Instruction*, 52(4):255–280.

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

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*, Stroudsburg, PA, USA. Association for Computational Linguistics.

Yuning Ding, Brian Riordan, Andrea Horbach, Aoife Cahill, and Torsten Zesch. 2020. Don't take "nswvtnvakgxpm" for an answer –the surprising vulnerability of automatic content scoring systems to adversarial input. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 882–892, Barcelona, Spain (Online). International Committee on 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.

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

Sungho Jeon and Michael Strube. 2021. Countering the influence of essay length in neural essay scoring. In *Proceedings of the Second Workshop on Simple and Efficient Natural Language Processing*, pages 32–38, Virtual. Association for Computational Linguistics.

Cancan Jin, Ben He, Kai Hui, and Le Sun. 2018. TDNN: A two-stage deep neural network for prompt-independent automated essay scoring. In *Proceedings of the 56th Annual Meeting of the Association for*

Computational Linguistics (Volume 1: Long Papers), pages 1088–1097, Melbourne, Australia. Association for Computational Linguistics.

Anubha Kabra, Mehar Bhatia, Yaman Kumar Singla, Junyi Jessy Li, and Rajiv Ratn Shah. 2022. Evaluation toolkit for robustness testing of automatic essay scoring systems. In *5th Joint International Conference on Data Science & Management of Data (9th ACM IKDD CODS and 27th COMAD)*, pages 90–99.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. In *Advances in Neural Information Processing Systems*, volume 33, pages 18661–18673. Curran Associates, Inc.

Jiawei Liu, Yang Xu, and Yaguang Zhu. 2019. Automated essay scoring based on two-stage learning. *arXiv preprint arXiv:1901.07744*.

Nitin Madnani and Aoife Cahill. 2018. Automated scoring: Beyond natural language processing. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1099–1109.

Sandeep Mathias, Rudra Murthy, Diptesh Kanojia, Abhijit Mishra, and Pushpak Bhattacharyya. 2020. Happy are those who grade without seeing: A multi-task learning approach to grade essays using gaze behaviour. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 858–872, Suzhou, China. Association for Computational Linguistics.

Panitan Muangkammuen and Fumiyo Fukumoto. 2020. Multi-task learning for automated essay scoring with sentiment analysis. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing: Student Research Workshop*, pages 116–123, Suzhou, China. Association for Computational Linguistics.

Les Perelman. 2014. When “the state of the art” is counting words. *Assessing Writing*, 21:104–111.

Thang M Pham, Trung Bui, Long Mai, and Anh Nguyen. 2020. Out of order: How important is the sequential order of words in a sentence in natural language understanding tasks? *arXiv preprint arXiv:2012.15180*.

Dadi Ramesh and Suresh Kumar Sanampudi. 2021. An automated essay scoring systems: a systematic literature review. *Artificial Intelligence Review*, pages 1–33.

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.

Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. 2018. Self-attention with relative position representations. 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 464–468, New Orleans, Louisiana. Association for Computational Linguistics.

Yaman Kumar Singla, Swapnil Parekh, Somesh Singh, Junyi Jessy Li, Rajiv Ratn Shah, and Changyou Chen. 2021. Aes systems are both overstable and oversensitive: Explaining why and proposing defenses. *arXiv preprint arXiv:2109.11728*.

Dong Su, Huan Zhang, Hongge Chen, Jinfeng Yi, Pin-Yu Chen, and Yupeng Gao. 2018. Is robustness the cost of accuracy?–a comprehensive study on the robustness of 18 deep image classification models. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 631–648.

Jingbo Sun, Tianbao Song, Jihua Song, and Weiming Peng. 2022. Improving automated essay scoring by prompt prediction and matching. *Entropy*, 24(9):1206.

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

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Jin Xue, Xiaoyi Tang, and Liyan Zheng. 2021. A hierarchical bert-based transfer learning approach for multi-dimensional essay scoring. *IEEE Access*, 9:125403–125415.

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, Online. Association for Computational Linguistics.

Mo Zhang. 2013. Contrasting automated and human scoring of essays. *R & D Connections*, 21(2):1–11.

A Data Tables

Prompt	No. Essays	Score Range	Avg. Length	Lowest Label Score %
1	1,783	2-12	350	0.56%
2	1,800	1-6	350	1.33%
3	1,726	0-3	150	2.26%
4	1,772	0-3	150	17.61%
5	1,805	0-4	150	1.33%
6	1,800	0-4	150	2.44%
7	1,569	0-30	250	0.0%
8	723	0-60	650	0.0%

Table 5: Summary statistics of the ASAP dataset. Score Range column indicates integer range of score labels. Lowest Label Score Percentage measures the portion of essays assigned with the lowest label score for each prompt. For example, in prompt 1, 0.56% of 1,783 essays are assigned with the lowest label score of 2.

Prompt	Genre	Level	Rubric
1	ARG	8	
2	ARG	10	
3	RES	10	$\times$
4	RES	10	$\times$
5	RES	8	$\triangle$
6	RES	10	$\triangle$
7	NAR	7	
8	NAR	10	

Table 6: Metadata of the ASAP dataset. Genre column indicates the type of essays including argumentative essays, response essays (source-dependent), and narrative essays. Level column indicates the grade level of the essay writers. Rubric column indicates prompts sharing the same scoring rubric. Scoring rubrics are identical for prompts 3 and 4 and prompts 5 and 6.

Prompt	Scoring Guide for Irrelevant Essay
3	assign lowest score
4	assign lowest score
5	assign lowest score
6	assign lowest score

Table 7: Scoring rubric for source dependant essays require evaluation of relevance between essay response and question prompt.

B Hyper-parameters

Our experiments are conducted with 4 NVIDIA GeForce RTX 3090 GPUs, and training batch size for each AES model is set to match the maximum GPU memory limit.

Prompt 3-4 Model We train Prompt 3-4 model on the rubric-segmented dataset for 20 epochs. We apply learning rate of 4×10^{-5} for the pre-trained BERT and 8×10^{-5} for the custom attention layers which are trained from scratch. For accurate performance comparison, develop and test set performances are recorded and reported separately for each prompt. Training batch of size 40 is applied with 5% Prompt Swap rate, resulting in total of 42 training data samples for each batch.

Prompt 5-6 Model Prompt 5-6 model is trained for 40 epochs in total, and cosine similarity is optimized for the first 20 epochs only. After 20 epochs, Prompt 5-6 model training only optimizes the MSE loss. To make sure a given training batch is sufficiently diverse, each batch item is paired with a positive and negative sample randomly selected from outside the training batch, resulting in training batch of size 16. We apply learning rate of 8×10^{-5} for the pre-trained BERT, 1×10^{-4} for the custom attention layers, and 3×10^{-4} for cosine similarity optimization. Training batch of size 16 is applied with 2 Prompt Swap samples per batch, but prompt swapped samples are excluded from cosine similarity computation.

Prompt 1, 2, 7, and 8 Models Prompts 1, 2, 7, and 8 have distinct scoring rubrics and therefore are trained separately with distinct hyper-parameter settings. Response Distortion is applied to essay responses that have the lowest label scores when applicable. Learning rates ranging from 1×10^{-5} to 4×10^{-5} are applied for the pre-trained BERT and 8×10^{-5} to 1×10^{-4} for the custom attention layers. Batch size ranges from 10 to 16 with a response distortion rate of 1 augmented sample per batch over 10 to 20 training epochs.

Irrelevant Response Detection During irrelevant response detection testing for Prompt 3-4 and Prompt 5-6 models, we apply Prompt Swap rate of 10% to generate 72 prompt mismatched test samples for each prompt and each fold. To better capture the contribution of Prompt Swap, Prompt Swap is only applied to essays with scores greater than the average score during testing. Prompt mismatched samples are only used to compute irrelevant response detection rate and are not included in test set QWK calculation.

Distorted Response Detection During distorted response detection testing for Prompt 1, 2, 3-4 and 5-6 models, we apply Response Distortion to all samples in each rubric-segmented test set for all folds. Response Distortion is applied to essay samples without score conditions during testing. Moreover, the same test is conducted with different Distort Rate values. Distort Rates are only applied during testing to control the magnitude of permutation applied on each test sample. Distorted samples are only used to compute distorted response detection rate and are not included in test set QWK calculation.

C Additional Results

Attention Score	Sentences from Question Prompt
0.0368	"Winter Hibiscus by Minfong Ho Saeng, a teenage girl, and her family have moved to the United States from Vietnam."
0.0325	"A wave of loss so deep and strong that it stung Saeng's eyes now swept over her."
0.0280	"I'd read once that sucking on stones helps take your mind off thirst by allowing what spit you have left to circulate."
0.0019	"Write a response that explains why the author concludes the story with this paragraph."
0.0029	"How did it go? Did you-?"
0.0029	"Goodness, it's past five. What took you so long?"

Table 8: Response-prompt attention scores computed during Prompt 3-4 model training with Prompt Swap. Table includes three largest and three smallest attention score values with their corresponding question prompt sentence. Sentences corresponding to the first two largest attention scores can be easily associated with prompt 4, which is a story describing the struggles of immigration. Similarly, the third largest attention score can be associated with prompt 3, which is an essay describing a cyclist's battle against thirst and dehydration. On the contrary, prompt sentences with the lowest attention scores cannot be directly associated with either prompt 3 or prompt 4.

AES Model	QWK		No. of Trained Models
	3-4	5-6	
Yang et al. (2020)	0.772	0.844	4
Jeon and Strube (2021)	0.761	0.813	4
Rubric-Specific Model w/ Augmentation Training	0.774	0.836	**2**

Table 9: Test set QWK performance (prompt averages) and number of trained AES models for benchmark models utilizing pre-trained language models. Prompt-specific approach requires training n models for n question prompts. On the other hand, rubric-specific approach only requires training 1 model for n question prompts as long as the prompts share the same rubric.