Automated Essay Scoring Incorporating Annotations from Automated Feedback Systems

Christopher Ormerod
Cambium Assessment Inc.
christopher.ormerod@cambiumassessment.com

Abstract

This study illustrates how incorporating feedback-oriented annotations into the scoring pipeline can enhance the accuracy of automated essay scoring (AES). This approach is demonstrated with the Persuasive Essays for Rating, Selecting, and Understanding Argumentative and Discourse Elements (PERSUADE) corpus. We integrate two types of feedback-driven annotations: those that identify spelling and grammatical errors, and those that highlight argumentative components. To illustrate how this method could be applied in real-world scenarios, we employ two LLMs to generate annotations – a generative language model used for spell correction and an encoder-based token-classifier trained to identify and mark argumentative elements. By incorporating annotations into the scoring process, we demonstrate improvements in performance using encoder-based large language models fine-tuned as classifiers.

1 Introduction

Automated Essay Scoring (AES) uses statistical models to assign grades to essays that approximate hand-scoring (Shermis and Hamner, 2013). Automated Writing Evaluation (AWE) is the provision of automated feedback designed to help students iteratively improve their essays (Huawei and Aryadoust, 2023). Initial attempts at AES and AWE were based on Bag-of-Words (BoW) models that combine frequency-based and hand-crafted features (Attali and Burstein, 2006; Page, 2003). Well-designed features can serve two purposes: to improve scoring accuracy and provide feedback to students to improve their essays. These features tend to be global features, such as the number of words, sentence length, or readability metrics, and are not based on fine-grained semantics or the organizational structure of essays.

Many modern AES engines employ transformer-based Large Language Models (LLM)s (Rodriguez et al., 2019), which offer improved accuracy over bag-of-words models. However, this comes at the expense of reduced interpretability due to implicit feature definition. Some researchers have sought to combine LLM-derived features with traditional hand-crafted features to enhance accuracy and provide some level of interpretability (Uto and Uchida, 2020). LLMs also offer the ability to provide semantically rich feedback, such as key phrases from explainable AI (Boulanger and Kumar, 2020), annotation schemas for automated writing evaluation systems (Crossley et al., 2022; Lottridge et al., 2024), and the generation of detailed feedback through LLM prompting techniques (Lee et al., 2024). This study considers how these semantically rich features, designed primarily for providing feedback, can also enhance scoring accuracy.

We demonstrate our approach using the Persuasive Essays for Rating, Selecting, and Understanding Argumentative and Discourse Elements (PERSUADE) corpus, which is a dataset of essays in which the argumentative components of the essays were annotated (Crossley et al., 2022). This dataset also contains scores assigned against an openly available holistic rubric [1] and demographic data, which allows us to test any AES system with respect to operational standards, including the addition of potential bias (Williamson et al., 2012).

The two classes of features we consider are derived from Grammatical Error Correction (GEC) and Computational Argumentation. The goal of GEC is to provide a mapping from a sentence that may or may not contain errors in language, to a version with the same meaning with fewer language errors (Martynov et al., 2023), while computational argumentation seeks to isolate and analyze the set of argumentative components of an essay (Stab and Gurevych, 2014a). For these features to be incorporated into an AES pipeline, we leverage

[1] https://github.com/scrosseye/persuade_corpus_2.0

Proceedings of the Artificial Intelligence in Measurement and Education Conference (AIME-Con) – Volume 3: Coordinated Session Papers, pages 9–18

the ability of language models to accurately annotate argumentative components (Ormerod et al., 2023) and correct spelling and grammatical errors (Rothe et al., 2021). These spelling and grammatical corrections can then be annotated and classified to provide locally defined information on conventions (Korre and Pavlopoulos, 2020). Once these annotations have been derived, we incorporate the annotations using Extensible Markup Language (XML) for easy parsing, which facilitates natural integration into an AWE system based on essays encoded in HTML.

We must also be cognizant that increased automation can exacerbate biases, especially for English Language Learners (Ormerod, 2022a). For this reason, we also examine whether this pipeline leads to greater bias by examining the standardized mean difference for the relevant subgroups.

In §2, we describe our methods, including the data used, the models, and the approach. The performance of the annotation models and the scoring models are presented in §3. We will discuss the findings and suggest future directions in §4.

2 Method

2.1 Data

The PERSUADE corpus is an openly available dataset of 25,996 argumentative essays between grades 6 and 12 on a range of 15 different topics (Crossley et al., 2022). The essays are responses to prompts that are either dependent on source material, or independent of source material. The set has been divided into a training set and a test set by the original authors. An outline of the composition of these two sets is presented in Table 1.

2.1.1 Annotations

A key characteristic of the corpus that makes it useful from the standpoint of computational argumentation is the annotations. The argumentative clauses of each essay were identified and classified into one of seven classes:

1. **Lead (L):** An introduction that begins with a statistic, a quotation, a description, or some other device to grab the reader's attention and point toward the thesis.
2. **Position (P):** An opinion or conclusion on the main question
3. **Claim (C1)**: A claim that supports the position
4. **Counterclaim (C2)**: A claim that refutes another claim or gives an opposing reason to the position
5. **Rebuttal (R)**: A claim that refutes a counterclaim
6. **Evidence (E)**: Ideas or examples that support claims, counterclaims, rebuttals, or the position
7. **Concluding Statement (C3)**: A concluding statement that restates the position and claims

Grade	Train	Test	Total	Avg. Len.
6	688	684	1372	294.6
8	5614	4015	9629	374.9
9	1831	235	2066	426.6
10	4654	3620	8274	407.6
11	1863	1220	3083	610.9
12	243	161	404	469.0
Unk.	701	467	1168	452.5
Total	15594	10402	25996	418.1

Table 1: The grade level and length statistics for the training and testing splits for the PERSUADE corpus, and the counts of essays that are responses to prompts that are dependent (Dep.) on source material and independent (Ind..) of source material.

There are an average of 11.0 annotated components in each essay. Some descriptive statistics on the distribution of applied labels can be found in Table 2. Among the labels, the most frequently applied labels are Claims and Evidence and the least frequently applied labels are Counterclaims and Rebuttals. In accordance with state standards, the development of a counter-argument in persuasive essay writing is developed at grades eight and beyond, hence, Counterclaims and Rebuttals are rarely applied at the sixth-grade level.

	6	8	9	10	11	12
L	4.3	4.5	4.7	5.9	6.0	5.0
P	9.8	9.0	9.1	9.9	7.0	8.3
C1	27.2	27.7	27.6	29.8	31.2	34.3
C2	2.1	2.8	5.2	2.5	5.3	5.0
R	1.5	2.2	3.7	1.7	4.3	2.0
E	27.3	25.7	26.8	28.7	23.4	27.5
C3	8.0	7.5	7.6	8.7	6.9	7.5

Table 2: Some descriptive statistics regarding the distribution of annotations with respect to the various grade levels.

We did not use the effectiveness scores for the discourse elements in this study. This was a conscious choice due to the fairly low agreement between the effectiveness scores assigned by human raters ($\kappa = 0.316$). Similar attempts at judging the quality of arguments have also yielded low agreement rates (Gretz et al., 2019; Toledo et al., 2019).

2.1.2 Scores

Each essay was graded against a standardized SAT holistic essay scoring rubric, which was slightly modified for the source-based essays [2]. Based on the rubric, a high-scoring essay (5-6) demonstrates mastery through effective development of a clear point of view, strong critical thinking with appropriate supporting evidence, well-organized structure with coherent progression of ideas, skillful language use with varied vocabulary and sentence structure, and minimal grammatical errors. In contrast, a lower-scoring essay (1-3) exhibits significant weaknesses: vague or limited viewpoint, weak critical thinking with insufficient evidence, poor organization resulting in disjointed presentation, limited vocabulary with incorrect word choices, frequent sentence structure problems, and numerous grammatical errors that interfere with meaning.

The score distribution is fairly regular, with the highest and lowest scores being the rarest. The full score distribution can be found in Table 3. The reported inter-rated reliability, as reported in (Crossley et al., 2022), is given by $\kappa = 0.745$ (see (6)).

[2] https://www.kaggle.com/datasets/davidspencer/persuade-rubric-holistic-essay-scoring

Score	1	2	3	4	5	6
%	4.0	21.9	32.2	25.9	12.7	3.4

Table 3: The score distribution for the PERSUADE dataset.

The key differentiators in the rubric between high and low scoring essays are the clarity of thought, quality of supporting evidence, organizational coherence, and technical proficiency in language use. The premise behind the approach is that organizational coherence and technical proficiency in language are both made clearer by highlighting the argumentative components and conventions-based errors. Provided our pipeline for annotating essays is sufficiently accurate, these annotations should help the engine align scores with the rubric.

2.1.3 Augmented Data

The input into the scoring model was augmented to use the annotation information using Extensible Markup Language (XML). We have an XML tag per argumentative component type. To annotate conventions errors, we used the ERRANT tool (Bryant et al., 2017). The ERRANT tool classifies errors into 25 different main types, with many of these categories appearing with three different subtypes: "R" for replace, "M" for missing, and "U" for Unnecessary. For example, one category, "PUNCT", refers to a punctuation error. We can either replace, remove, or add punctuation to a sentence to make it correct, corresponding to "R:PUNCT", "U:PUNCT", and "M:PUNCT", respectively. We refer to (Bryant et al., 2017) for a full explanation of the categories.

To simplify the categories for annotation purposes, we divide all possible ERRANT annotations into three labels: `<Spelling >`, `<PunctOrth >`, and `<Grammar >`. The `<Spelling >` label is applied to the subcategories of "SPELL", the `<PunctOrth >` is applied to the subcategories of "PUNCT" and "ORTH", while all other categories are designated as having labels of `<Grammar >`. This means that we have a total of 10 annotation labels: 7 associated with argumentative components and 3 for convention errors. An example of the input into the model is shown in Figure 1. Since we do not have human-annotated data, the augmented data relies on the output from an annotation model and a spell-correction model, both of which can contribute to annotations that are less accurate.

```
<Lead>There can be many <Spelling>
advanteges</Spelling> and <Spelling>
disadvanteges</Spelling> to having a
car but the <Spelling>advanteges
<Spelling> to not having a <Grammar>card
</Grammar> greatly outweighs having one.
</Lead>There can be many reasons why not
having a car is great but the main three
are <Claim>it reduces pollution reduces
stress</Claim> and <Claim>having less
cars reduces the <Spelling>noice
<Spelling> pollution in a city. </Claim>
Can you imagine a place with no cars?
```

Figure 1: An example of the model input using an excerpt of an annotated essay.

Demographics: The last detail of this dataset, which makes it exceptionally well-suited to the investigation of AES in an operational setting, is the accompanying demographic information. This allows us to investigate any additional potential bias introduced in modeling. We measure bias by the original operational standards defined by Williamson et al. (Williamson et al., 2012). While this standard is important for many reasons, there have been numerous alternative approaches to bias (Ormerod et al., 2022).

key	Subgroup	Train	Test
WC	White/Caucasian	7012	4559
HL	Hispanic/Latino	3869	2691
BA	Black/African American	2975	1984
AP	Asian/Pacific Islander	1072	671
Mix	Two or more	598	424
Nat	American Indian Alaskan Native	68	73
ELL	English Language Learner	1330	914
DE	Disadvantaged Economically	5391	4252
ID	Identified Disability	1516	1172

Table 4: The main subgroup populations in the train and test set.

The population of various subgroups in the train and test split, as presented in the data, have been outlined in Table 4.

2.2 Modeling details

Since the advantages of the transformer were first celebrated (Vaswani et al., 2017) and BERT was trained (Devlin et al., 2018), many of the state-of-the-art results can be attributed to transformer-based LLMs (Wang et al., 2019). For this reason, we restrict our attention to fine-tuned transformer-based LLMs, whose architectures can be described as encoder, decoder, or encoder-decoder models (Vaswani et al., 2017). As a general rule, encoder models excel in natural language inference tasks (Devlin et al., 2018), decoder models excel in generative tasks (Radford et al., 2018), and encoder-decoder models excel in translation, where the task benefits from representation learning (Raffel et al., 2020).

2.2.1 Annotation model

The task of annotations can be framed as a token-classification task, where each token is classified into one of eight possible labels, one for each possible argumentative component in addition to one extra label for unannotated regions. This task lends itself to an encoder-based model trained as a masked language model like BERT (Devlin et al., 2018). Since BERT, arguably the best performing series of models are Microsofts' DeBERTa model series (He et al., 2021). The problem with these models is that many of the essays exceed the 512 token limit after tokenization. For this reason, we turn to a newly developed long context model known as ModernBERT (Warner et al., 2024).

Aside from some differences in the choices of normalization layers (Xiong et al., 2020), the use of gated activation functions (Shazeer, 2020), and more extensive pretaining, the biggest difference in the architecture is the use of Rotational Positional Embeddings (RoPE) (Su et al., 2024). To understand how RoPE works, in the original implementation of attention, the output of attention is given as a function of the key vectors, k_i, query vectors, q_j, and value vectors, v_l, given by

$$a_{m,n} = \frac{\exp(q_m^T k_n/\sqrt{d})}{\sum_j \exp(q_m^T k_j/\sqrt{d})}. \quad (1)$$

where key, query, and value vectors are functions of the embedding vectors at the first attention layer. The standard construction is that these functions be affine linear functions (linear with a bias term), where the positional embedding is the addition of token embeddings and some learnable positional

embedding terms. Instead of adding a vector, we define the query and key vectors using

$$q_m = R^d_{\Theta,m} W_q x_m, \qquad k_m = R^d_{\Theta,m} W_k x_m \tag{2}$$

where $R^d_{\Theta,m}$ a block-diagonal matrix of 2-dimensional rotation matrices, r_ϕ, given by

$$R^d_{\Theta,m} = \text{diag}(r_{m\theta_1}, \ldots, r_{m\theta_{d/2}}), \tag{3}$$

$$r_\phi = \begin{pmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{pmatrix}, \tag{4}$$

with $\theta_i = \tilde{\vartheta}^{-2(i-1)/d}$. The term ϑ is called the RoPE Theta. The key idea is that this transformation encodes positional information by rotating token embeddings in a particular manner that allows the model to understand relative distances between tokens rather than just absolute positions.

The ModernBERT model, like BERT, is trained as a masked-language model with an encoder and a token classification head. The structure of the encoder includes multiple layers of self-attention in which the attention layers alternate between rotary embeddings with base $\vartheta = 1 \times 10^4$ and $\vartheta = 1.6 \times 10^5$. However, the training was done in two phases. By adjusting the value of ϑ, researchers developed a way to scale the context length of the RoPE embedding (Fu et al., 2024), which has been applied to many other models (AI@Meta, 2024). Using this technique, the ModernBERT model was pretrained on 1.7 trillion tokens with a context length of 1024 with a $\vartheta = 10^{-4}$, which was extended to 8196 by additional training with altered layers in which $\vartheta = 1.6 \times 10^5$. In this manner, one way of thinking of the change in ϑ values in the encoder is that the attention mechanism alternates between global and local attention.

It should be noted that many architectures have attempted to circumvent the context limitation, such as the Reformer (Kitaev et al., 2020), Longformer (Beltagy et al., 2020), Transformer-XL (Dai et al., 2019), and XLNet (Yang et al., 2019), to name a few. Many of these solutions use some sort of sliding context window and/or a recurrent adaptation of the transformer architecture. Extending the context using rotary positional embeddings is more computationally efficient and effective at scaling to large context lengths, making ModernBERT a more appropriate choice in this context.

The pretrained ModernBERT model was modified for annotation by adding a classification head to the encoder. The annotator model possesses a classification head with 9 output dimensions: 7 for argumentative component labels, 1 for unannotated text, and 1 for padded variables (which can be disregarded). The output represents log probabilities for each label. This model was trained to predict the annotations for each token in the training set using the cross-entropy loss function and the Adam optimizer with a learning rate of 1e-6 over 10 epochs. We simplified the training by not having a development set.

2.2.2 Spelling and Grammar

The most successful and accurate way to perform GEC has been to utilize representation learning, hence, we seek an encoder-decoder model, which is a sequence-to-sequence model (Sutskever et al., 2014). The premise behind this method is that we are able to use the encoder to map sentences to a vector space that encodes the semantic information, while a decoder maps from the vector space to grammatically correct text (Rothe et al., 2021). This suggests we use a T5 model, pretrained as a text-to-text transformer and fine-tuned to perform grammatical error correction (GEC) (Martynov et al., 2023). Once the correction is defined, the original sentence and the correction are used as input into the ERRANT tool to produce a classified correction (Korre and Pavlopoulos, 2020).

2.2.3 Scoring Models

Given the model input includes both the essay and any annotations, we require long-context models. While we have experimented with the use of QLoRA-trained generative Models (Ormerod and Kwako, 2024), to simplify the presentation, we use the ModernBERT model for scoring in addition to annotation. In this case, the scoring models were constructed by appending a linear classification head to the ModernBERT model with 6 targets, one for each score point.

2.3 Evaluation

We have two models to evaluate: an annotation model and a classification model. There are many challenges to assessing annotations for argumentative clauses. We need to carefully define what it means for a particular clause to be identified and correctly classified, given that certain identifications may not perfectly align with the predicted components. For holistic scoring, there are many more well-defined and accepted standards presented by Williamson et al. (Williamson et al.,

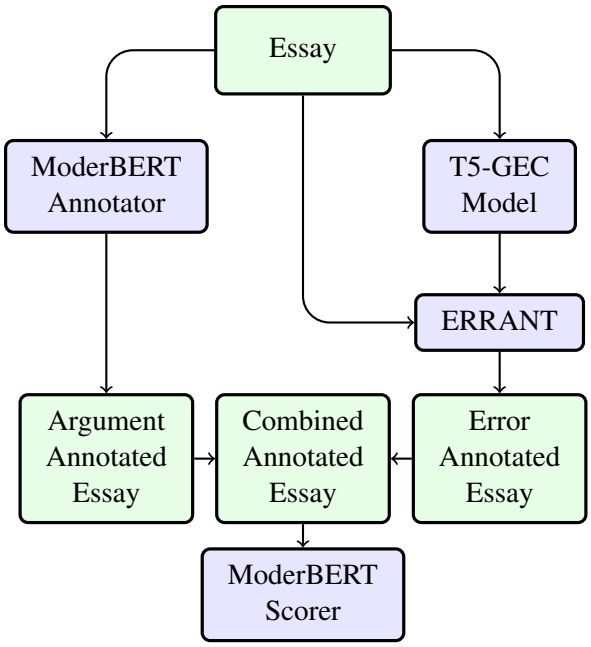

Figure 2: A diagram representing the scoring pipeline.

2012).

2.3.1 Annotator Evaluations

The standard described by the annotated argumentative essay dataset (Stab and Gurevych, 2014a,b) is reduced to a classification of IOB tags, where the governing statistic is an F1 score. Our guiding principle is the original rules for the competition[3], in which we use the ground truth and consider a match if there is over 50% overlap between the two identified components. Given a match using this rule, matching the argumentative type is considered a true positive (TP), unmatched components are considered false negatives (FN), while predicted label mismatches are considered false positives (FP). The final reported value is the F1 score, given by the familiar formula

$$F1 = \frac{2TP}{2TP + FP + FN}. \tag{5}$$

We can compare agreements for each label applied based on the ground truth. In this way, for each type of argumentative component type, we have a corresponding F1 score. In accordance with the rules of the competition, the final F1 statistic of interest is the macro average, given by the unweighted average over all the classes.

[3]https://www.kaggle.com/c/feedback-prize-2021/overview

2.3.2 Error Annotations

When it comes to annotating errors in the use of language, since the errors in the essays were not explicitly annotated by hand, we have no direct way of evaluating the accuracy of any annotations. We can only rely on the accuracy of the individual components. The T5 model we used has been evaluated in (Martynov et al., 2023) with respect to the JFLEG dataset (Napoles et al., 2017) and the BEA60k dataset (Jayanthi et al., 2020). According to those benchmarks, the accuracy of the model used is comparable to ChatGPT and GPT-4.

2.3.3 Automated Scoring Evaluations

For the scoring model, we use the standards for agreement specified by Williamson et al. (Williamson et al., 2012). The first and primary statistic used to describe agreement is Cohen's quadratic weighted kappa (QWK) (Cohen, 1960). Given scores between 1 and N, we define the weighted kappa statistic by the formula

$$\kappa = 1 - \frac{\sum w_{ij} O_{ij}}{\sum w_{ij} E_{ij}} \tag{6}$$

where O_{ij} is the number of observed instances where the first rater assigns a score of i and the second rater assigns a score of j, and E_{ij} are the expected number of instances that first rater assigns a score of i and the second rater assigns a score of j based purely on the random assignment of scores given the two rater's score distribution. This becomes the QWK when we apply the quadratic weighting:

$$w_{ij} = \frac{(i-j)^2}{(N-1)^2}. \tag{7}$$

The QWK takes values from -1 and 1, indicating perfect disagreement and agreement, respectively. It is often interpreted as the probability of agreement beyond random chance. The second statistic used is exact agreement, which is viewed as less reliable since uneven score distributions can skew it.

The last statistic used is the standardized mean difference (SMD). If y_t represents the true score and y_p represents the predicted score, then the SMD is given by

$$SMD(y_t, y_p) = \frac{\overline{y_p} - \overline{y_t}}{\sqrt{(\sigma(y_p)^2 + \sigma(y_t)^2)/2}}. \tag{8}$$

This statistic can be interpreted as a standardized relative bias. A positive or negative value indicates that the model is introducing some positive or

negative bias in the modeling process, respectively. Furthermore, when we restrict this calculation to scores for a specific demographic, assuming that demographic is sufficiently well-represented, the SMD is considered a gauge of the bias associated with the modeling for that subgroup.

3 Results

3.1 Annotation Accuracy

Using the 50% overlap rule, we present the number of true positives, false positives, and false negatives, and the resulting F1 score for each of the component types, excluding unannotated. These results are presented in Table 5.

	TP	FP	FN	F1
L	4332	270	95	0.960
P	6359	832	205	0.924
C1	20764	2057	1094	0.929
C2	1839	654	164	0.818
R	1443	495	106	0.827
E	18838	1653	1299	0.927
C3	5874	321	321	0.950

Table 5: The number of true positives (matched components and labels), false positives (matched components, unmatched labels), and false negatives (unmatched components) between the true annotations and the predicted annotations by component type.

These scores are exceptionally high, with the lowest performance given by the annotator's ability to discern counterclaims and rebuttals. As an indication of the annotated errors in language, the pipeline highlighted 2795 spelling errors, 1401 grammatical errors, and 201 punctuation or orthography errors. The pipeline used does not seem to be uncovering as many errors as expected.

3.2 Scoring Accuracy

Given that the classification head is typically randomly initialized, we were also interested in whether these results were stable. We trained the scorer 10 times and reported the average, minimum, and maximum agreement levels for each agreement statistic we listed above. These statistics can be found in Table 6.

All the models performed well above the human baseline. Out of the 10 separate trials, no model trained on the full-text alone scored as accurately as any of the models trained on the component annotated data or the data with both argumentative components and language errors annotated. An interesting observation is that the SMD, as calculated by (8), is only positive for models trained on the combined annotations, and that the models trained on component annotated text showed the most controlled SMDs.

3.3 Potential bias

To investigate the possibility of potential bias, we consider the SMD defined on subgroups. These SMDs are presented in Table 7.

Key	Orig.	Comp.	Error	Comb.
Female	0.12	0.11	0.12	0.06
WC	0.09	0.11	0.08	0.15
HL	-0.25	-0.25	-0.24	-0.19
BA	-0.17	-0.16	-0.20	-0.17
AP	0.44	0.45	0.53	0.52
Nat	-0.43	-0.41	-0.44	-0.21
Mix	0.08	0.07	0.12	0.01
ELL	-0.59	-0.60	-0.62	-0.54
DE	-0.36	-0.35	-0.37	-0.32
ID	-0.51	-0.48	-0.49	-0.42

Table 7: The bias in the various subgroups as measured by SMD for the particular subgroup.

While the resulting bias was higher than expected, a cursory look seems to suggest that the use of combined annotations is mitigating some of the biases rather than exacerbating them.

This work is one of a number of works that highlight the growing need to address the bias introduced by automation, especially for ELL students (Ormerod et al., 2022). This suggests we need to apply bias mitigation, which could be of the form of a regression-based system with adjusted cut-off points (Ormerod, 2022b), or some sort of reinforcement learning mechanism.

4 Discussion

The results of this study demonstrate that incorporating feedback-oriented annotations into automated essay scoring (AES) pipelines can significantly improve scoring accuracy and provide meaningful, interpretable insights for students. The work of Uto and Uchida (2020) suggests this is also true for traditional global features. What we propose

	QWK			Exa			SMD		
	Avg	Min	Max	Avg	Min	Max	Avg	Min	Max
Full Text Only	0.860	0.859	0.862	67.2	66.9	67.5	-0.023	-0.027	-0.018
Component Annotated	0.868	0.867	0.870	68.9	68.6	69.1	-0.013	-0.017	-0.009
Error Annotated	0.858	0.856	0.859	67.1	66.9	67.5	-0.025	-0.028	-0.022
Combined Annotations	0.866	0.867	0.870	68.4	68.0	68.9	0.021	0.016	0.025
Human Baseline	0.745								

Table 6: The average QWK, Exa, and SMD results on the test set for over 10 trials of training the scoring model.

is a realignment of AES to incorporate AWE elements so that we can provide students with more than just a score.

One of the most critical findings from this study concerns the potential for introduced bias, particularly among subgroups such as English Language Learners (ELLs). Our SMD analysis revealed notable disparities across demographic groups, echoing previous research on the disproportionate impact of automated systems on linguistically diverse populations. While the current pipeline demonstrates strong overall performance, these disparities underscore the importance of ongoing bias mitigation strategies. However, we know that SMDs can be unreliable, especially for subgroups with smaller populations. Future work should explore methods such as regression-based adjustments, fairness-aware training techniques, or reinforcement learning approaches that explicitly account for subgroup characteristics.

References

AI@Meta. 2024. Llama 3 Model Card.

Yigal Attali and Jill Burstein. 2006. Automated Essay Scoring With e-rater® V.2. *The Journal of Technology, Learning and Assessment*, 4(3). Number: 3.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The Long-Document Transformer. *arXiv preprint*. ArXiv:2004.05150 [cs].

David Boulanger and Vivekanandan Kumar. 2020. SHAPed Automated Essay Scoring: Explaining Writing Features' Contributions to English Writing Organization. In *Intelligent Tutoring Systems*, pages 68–78, Cham. Springer International Publishing.

Christopher Bryant, Mariano Felice, and Ted Briscoe. 2017. Automatic Annotation and Evaluation of Error Types for Grammatical Error Correction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 793–805, Vancouver, Canada. Association for Computational Linguistics.

Jacob Cohen. 1960. A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20(1):37–46. Publisher: SAGE Publications Inc.

Scott A. Crossley, Perpetual Baffour, Yu Tian, Aigner Picou, Meg Benner, and Ulrich Boser. 2022. The persuasive essays for rating, selecting, and understanding argumentative and discourse elements (PERSUADE) corpus 1.0. *Assessing Writing*, 54:100667.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V. Le, and Ruslan Salakhutdinov. 2019. Transformer-XL: Attentive Language Models Beyond a Fixed-Length Context. *arXiv preprint*. ArXiv:1901.02860 [cs, stat].

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Technical Report arXiv:1810.04805, arXiv. ArXiv:1810.04805 [cs] type: article.

Yao Fu, Rameswar Panda, Xinyao Niu, Xiang Yue, Hannaneh Hajishirzi, Yoon Kim, and Hao Peng. 2024. Data Engineering for Scaling Language Models to 128K Context. *arXiv preprint*. ArXiv:2402.10171 [cs].

Shai Gretz, Roni Friedman, Edo Cohen-Karlik, Assaf Toledo, Dan Lahav, Ranit Aharonov, and Noam Slonim. 2019. A Large-scale Dataset for Argument Quality Ranking: Construction and Analysis. *arXiv preprint*. ArXiv:1911.11408 [cs].

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. DeBERTa: Decoding-enhanced BERT with Disentangled Attention. *arXiv preprint*. Number: arXiv:2006.03654 arXiv:2006.03654 [cs].

Shi Huawei and Vahid Aryadoust. 2023. A systematic review of automated writing evaluation systems. *Education and Information Technologies*, 28(1):771–795.

Sai Muralidhar Jayanthi, Danish Pruthi, and Graham Neubig. 2020. NeuSpell: A Neural Spelling Correction Toolkit. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 158–164, Online. Association for Computational Linguistics.

Nikita Kitaev, Łukasz Kaiser, and Anselm Levskaya. 2020. Reformer: The Efficient Transformer. Technical Report arXiv:2001.04451, arXiv. ArXiv:2001.04451 [cs, stat] type: article.

Katerina Korre and John Pavlopoulos. 2020. ERRANT: Assessing and Improving Grammatical Error Type Classification. In *Proceedings of the 4th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, pages 85–89, Online. International Committee on Computational Linguistics.

Gyeong-Geon Lee, Ehsan Latif, Xuansheng Wu, Ninghao Liu, and Xiaoming Zhai. 2024. Applying large language models and chain-of-thought for automatic scoring. *Computers and Education: Artificial Intelligence*, 6:100213.

Susan Lottridge, Amy Burkhardt, Christopher Ormerod, Sherri Woolf, Mackenzie Young, Milan Patel, Harry Wang, Julius Frost, Kevin McBeth, and Julie Benson. 2024. Write On with Cambi: The development of an argumentative writing feedback tool.

Nikita Martynov, Mark Baushenko, Anastasia Kozlova, Katerina Kolomeytseva, Aleksandr Abramov, and Alena Fenogenova. 2023. A Methodology for Generative Spelling Correction via Natural Spelling Errors Emulation across Multiple Domains and Languages. *arXiv preprint*. ArXiv:2308.09435 [cs].

Courtney Napoles, Keisuke Sakaguchi, and Joel Tetreault. 2017. JFLEG: A Fluency Corpus and Benchmark for Grammatical Error Correction. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 229–234, Valencia, Spain. Association for Computational Linguistics.

Christopher Ormerod. 2022a. Short-answer scoring with ensembles of pretrained language models. *arXiv preprint*. ArXiv:2202.11558 [cs].

Christopher Ormerod, Amy Burkhardt, Mackenzie Young, and Sue Lottridge. 2023. Argumentation Element Annotation Modeling using XLNet. *arXiv preprint*. ArXiv:2311.06239 [cs].

Christopher Ormerod, Susan Lottridge, Amy E. Harris, Milan Patel, Paul van Wamelen, Balaji Kodeswaran, Sharon Woolf, and Mackenzie Young. 2022. Automated Short Answer Scoring Using an Ensemble of Neural Networks and Latent Semantic Analysis Classifiers. *International Journal of Artificial Intelligence in Education*.

Christopher Michael Ormerod. 2022b. Mapping Between Hidden States and Features to Validate Automated Essay Scoring Using DeBERTa Models. *Psychological Test and Assessment Modeling*, 64(4):495–526.

Christopher Michael Ormerod and Alexander Kwako. 2024. Automated Text Scoring in the Age of Generative AI for the GPU-poor. *arXiv preprint*. ArXiv:2407.01873 [cs].

Ellis Batten Page. 2003. Project Essay Grade: PEG. In *Automated essay scoring: A cross-disciplinary perspective*, pages 43–54. Lawrence Erlbaum Associates Publishers, Mahwah, NJ, US.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving Language Understanding by Generative Pre-training.

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. Technical Report arXiv:1910.10683, arXiv. ArXiv:1910.10683 [cs, stat] type: article.

Pedro Uria Rodriguez, Amir Jafari, and Christopher M. Ormerod. 2019. Language models and Automated Essay Scoring. *arXiv preprint*. Number: arXiv:1909.09482 arXiv:1909.09482 [cs, stat].

Sascha Rothe, Jonathan Mallinson, Eric Malmi, Sebastian Krause, and Aliaksei Severyn. 2021. A Simple Recipe for Multilingual Grammatical Error Correction. *arXiv preprint*. Number: arXiv:2106.03830 arXiv:2106.03830 [cs].

Noam Shazeer. 2020. GLU Variants Improve Transformer. *arXiv preprint*. ArXiv:2002.05202 [cs].

Mark D. Shermis and Ben Hamner. 2013. Contrasting State-of-the-Art Automated Scoring of Essays. pages 335–368. Publisher: Routledge Handbooks Online.

Christian Stab and Iryna Gurevych. 2014a. Annotating Argument Components and Relations in Persuasive Essays. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1501–1510, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Christian Stab and Iryna Gurevych. 2014b. Identifying Argumentative Discourse Structures in Persuasive Essays. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 46–56, Doha, Qatar. Association for Computational Linguistics.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. 2024. RoFormer: Enhanced transformer with Rotary Position Embedding. *Neurocomputing*, 568:127063.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to Sequence Learning with Neural Networks. In *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc.

Assaf Toledo, Shai Gretz, Edo Cohen-Karlik, Roni Friedman, Elad Venezian, Dan Lahav, Michal Jacovi, Ranit Aharonov, and Noam Slonim. 2019. Automatic Argument Quality Assessment - New Datasets

and Methods. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5625–5635, Hong Kong, China. Association for Computational Linguistics.

Masaki Uto and Yuto Uchida. 2020. Automated Short-Answer Grading Using Deep Neural Networks and Item Response Theory. *AIED*.

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

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. Technical Report arXiv:1804.07461, arXiv. ArXiv:1804.07461 [cs] type: article.

Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, Nathan Cooper, Griffin Adams, Jeremy Howard, and Iacopo Poli. 2024. Smarter, Better, Faster, Longer: A Modern Bidirectional Encoder for Fast, Memory Efficient, and Long Context Finetuning and Inference. *arXiv preprint*. ArXiv:2412.13663 [cs].

David M. Williamson, Xiaoming Xi, and F. Jay Breyer. 2012. A Framework for Evaluation and Use of Automated Scoring. *Educational Measurement: Issues and Practice*, 31(1):2–13. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1745-3992.2011.00223.x.

Ruibin Xiong, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, and Tieyan Liu. 2020. On Layer Normalization in the Transformer Architecture. In *Proceedings of the 37th International Conference on Machine Learning*, pages 10524–10533. PMLR. ISSN: 2640-3498.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. XLNet: Generalized Autoregressive Pretraining for Language Understanding. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.