



An Empirical Analysis of Large Language Models for Automated Cross-Prompt Essay Trait Scoring in Brazilian Portuguese

André Barbosa   [Universidade de São Paulo | andre.barbosa@ime.usp.br]

Igor Cataneo Silveira  [Universidade de São Paulo | igorcs@ime.usp.br]

Denis Deratani Mauá  [Universidade de São Paulo | ddm@ime.usp.br]

 Institute of Mathematics and Statistics, R. do Matão, 1010 - Butantã, São Paulo - SP, 05508-090, Brazil.

Received: 01 April 2025 • **Accepted:** 28 July 2025 • **Published:** 06 October 2025

Abstract The development of automated essay grading systems with minimal human intervention has been pursued for decades. While these systems have advanced significantly in English, there is still a lack of in-depth analysis of the use of modern Large Language Models for automatic essay scoring in Portuguese. This work addresses this gap by evaluating different language model architectures (encoder-only, decoder-only, reasoning-based), fine-tuning and prompt engineering strategies. Our study focuses on scoring argumentative essays written as practice exercises for the Brazilian national entrance exam regarding five trait-specific criteria. Our results show that no architecture is always dominant, and that encoder-only models offer a good balance between accuracy and computational cost. We obtain state-of-the-art results for the dataset, obtaining trait-specific performance that ranges from .60 to .73 measured in Quadratic Weighted Kappa.

Keywords: Automatic Essay Scoring, Large Language Models, Natural Language Processing

1 Introduction

Automatic Essay Scoring (AES) systems promise to release educators from the burden of grading written assignments, scaling up the ability to provide timely, consistent and useful feedback [Page, 1966]. Many large-scale high-stakes standardized proficiency tests already incorporate automated scoring in their evaluation [Attali and Burstein, 2006; Klebanov and Madnani, 2020]. Typically, an AES system replaces a human annotator in a pool of assessments that are used to reach an average score.

While AES encompasses a variety of related tasks, such as providing inline corrections [Burstein *et al.*, 2004] and assessing the correctness of constructed responses [Chang and Ginter, 2024], we are here concerned with its more frequent form: assigning scores assessing the writing quality of essays. This type of assessment continues to be a prominent feature of higher-level educational environments [Schneer, 2014; Mei, 2006].

Early approaches to AES predominantly employed traditional machine learning techniques, leveraging carefully designed handcrafted features [Page, 1966; Attali and Burstein, 2006]. More recently, an “ImageNet moment” [Ruder, 2018] in Natural Language Processing has driven the adoption of modern deep learning techniques, significantly improving AES performance [Alikaniotis *et al.*, 2016; Dong *et al.*, 2017; Tay *et al.*, 2018]. Current systems are now able to match human inter-annotator agreement rates even at the challenging task of assigning trait-specific scores with unseen prompts [Jin *et al.*, 2018; Hussein *et al.*, 2020; Ridley *et al.*, 2020, 2021]. The recent use of Large Language Models [Radford *et al.*, 2018; Devlin *et al.*, 2019; Raffel *et al.*, 2020] aided

in mitigating the scarcity of annotated data and improved generalization [Rodriguez *et al.*, 2019; Silveira *et al.*, 2024, 2025]. In particular, their ability to perform zero-shot and in-context learning [Schick and Schütze, 2021] lifts the need for task-specific or prompt-specific fine-tuning [Mansour *et al.*, 2024; Lee *et al.*, 2024].

Following the trend of AES systems for English, early works on AES for Portuguese adopted a feature-engineering approach [Bazelato and Amorim, 2013]. Progressively, researchers turned to shallow representations based on word embeddings [Marinho *et al.*, 2022] and more sophisticated deep learning approaches using Recurrent Neural Networks [Fonseca *et al.*, 2018]. More recent work has investigated the use of pre-trained Large Language Models based on the Transformer Neural Network architecture [de Lima *et al.*, 2024; Ribeiro *et al.*, 2024; Silveira *et al.*, 2024, 2025]. To the best of our knowledge, last-generation Large Language Models with advanced reasoning abilities such as OpenAI O1 [OpenAI *et al.*, 2024a] and Deepseek R1 [DeepSeek-AI *et al.*, 2025] are yet to be evaluated for Portuguese AES.

In this work, we provide an extensive empirical analysis on the use of state-of-the-art Large Language Models for AES in Portuguese. We use the benchmark described by Silveira *et al.* [2024], which consists of publicly available essays written by high-school students in a variety of prompts as practice for standardized University entrance exams; each essay was annotated by two expert human graders with respect to five different traits, using a six-point ordinal scale. We conduct an in-depth analysis across various dimensions of language models, including differences in parameter size, architectures (such as encoder-only [Devlin *et al.*, 2019] and decoder-only types [Radford *et al.*, 2018]), and task special-

ization strategies such as full and parameter-efficient supervised fine-tuning [Hu *et al.*, 2022] and zero-shot learning [Brown *et al.*, 2020]. We release all the models and the scripts needed to reproduce our results.

Our results suggest that no single model class and learning strategy outperforms the others across different traits. Encode-only models obtain a good balance of performance and computational cost and zero-shot large language models seem to excel at some particularly difficult to evaluate traits. The overall choice does depend on amount of computational resources, minimum level of accuracy and additional concerns (such as the use of proprietary API-based models, environmental impact, inference speed, etc).

The remainder of this paper is organized as follows: Section 2 reviews related work. Section 3 presents the experimental setup, including the dataset, selection of Large Language Models and hyperparameters, and performance evaluation methodology. We then present and discuss the results in Section 4, along with a brief discussion on the environmental impact of our training. Finally, we conclude the paper in Section 5.

2 Related Work

AES traces back to the seminal work of Page [1966], which described the task and proposed a feature-engineering solution using linear regression from handcrafted textual features. That approach was later extended by the e-Rater system [Attali and Burstein, 2006], one of the most prominent real-world applications.

Eventually, the labor-intensive task of crafting these features was superseded by automatic feature extraction in the form of dense vector representations. Initial works used these representations together with different deep learning architectures [Taghipour and Ng, 2016] such as Convolutional Neural Networks [Dong and Zhang, 2016] and Long Short-Term Memory Recurrent Neural Networks [Alikaniotis *et al.*, 2016; Cummins and Rei, 2018]. Recently, Large Language Models (LLMs) based on the Transformer architecture became the standard tool for NLP after BERT became the state-of-the-art in a myriad of different tasks [Devlin *et al.*, 2019].

The AES task can actually be framed in different forms, involving different challenges. Single-prompt scoring considers grading a collection of essays under a fixed essay prompt, which implies uniform topic and grading criteria. In such a variant, traits such as topic or genre adherence and use of supporting texts can be inferred from statistical cues in the corpus, even without access to topic and additional material. This is arguably the most prevalent form found in the literature, possibly due to its commercial importance in standardized testing [Ke and Ng, 2019].

The cost and difficulty in obtaining large amounts of graded essays lead researchers to propose cross-prompt scoring [Jin *et al.*, 2018; Hussein *et al.*, 2020; Ridley *et al.*, 2020]. In such a variant, one instead receives a collection of essays from one or more source prompts/domains and needs to score essays on one or more different target-prompts/domains. One can also have access to a small subset of graded target-domain essays. This allows training and

test instances to differ in topic, genre (e.g. narrative vs. argumentative) and even grading criteria. In this setting, evaluating prompt-specific traits such as topic adherence and use of evidence requires more sophisticated techniques such as transfer learning to be effective [Ridley *et al.*, 2021]. As a consequence, performances in single-prompt scoring are typically higher than in cross-prompt scoring [Dong and Zhang, 2016; Li and Ng, 2024; Yang *et al.*, 2024].

There is also a distinction about the dimension of scores. Holistic scoring consists in assigning a single grade to each essay summarizing several different aspects of relevance. This form is mostly motivated by the larger availability of labeled data and improved system performance. Yet, feedback on multiple trait-specific dimensions is more informative and helpful to students, as well as a way to provide justification. Thus, in trait-specific scoring, also called analytic scoring, essays are graded along multiple dimensions that evaluate different competences or writing constructs, such as coherence [Higgins *et al.*, 2004], organization [Persing *et al.*, 2010] and thesis clarity [Persing and Ng, 2013].

Finally, there are different marking scales for scoring. One approach is to cast the problem as a regression task, that is, to score essays using a real-valued scale. One can interpret such a score as a form of average score of a hypothesized population of graders [Page, 1966]. Another approach is to mimic a human grader and provide scores on a finite ordinal scale. Here, a scoring model can be interpreted as simulating a particular grader or a type of median grader [Attali and Burstein, 2006; Klebanov and Madnani, 2020].

In this work, we consider the problem of cross-prompt, trait-specific scoring, where no access to the target prompt is provided. We assume the scoring model outputs scores on a finite ordinal scale on five different traits. This is the most prevalent task pursued for AES in Portuguese [Bazelato and Amorim, 2013; Amorim and Veloso, 2017; Marinho *et al.*, 2021; Mello *et al.*, 2024; Silveira *et al.*, 2024, 2025]. Compared to previous work, we here provide a more extensive analysis, including a much wider variety of models and configurations, and a more valid evaluation methodology that relies on the average performance with respect to two human annotators used separately as ground truth. This makes our results more comparable to other benchmarks, such as the Automatic Student Assessment Prize,¹ which involves student-written essays in English.

3 Experimental Setup

In this section, we describe the dataset and the models, along with their different usage configurations, used in our experiments. These models are grouped according to their nature: feature-based, encoder, and decoder-based language models. We also present our evaluation methodology, including the performance metrics and their use in the dataset at hand. To facilitate the interpretation of the results, we discuss some simple heuristics that exploit the labels of the test set to provide an upper bound on the optimum performance of a grader.

¹Hosted by Kaggle at <https://www.kaggle.com/c/asap-aes/>

Model Name	Version	Architecture	Context	#Params	#Train	Pretrain. Corpus
R ₀	—	Feature	—	1	1	—
Linear Regression	—	Feature	—	73	73	—
Random Forest	—	Feature	—	—	—	—
mBERT	bert-base-multilingual-cased	Encoder	512	178M	178M	Multilingual
BERTimbau Base	bert-base-portuguese-cased	Encoder	512	109M	109M	Portuguese
BERTimbau Large	bert-large-portuguese-cased	Encoder	512	334M	334M	Portuguese
BERTuguês	bertugues-base-portuguese-cased	Encoder	512	110M	110M	Portuguese
Albertina	albertina-1b5-portuguese-ptbr-encoder	Encoder	512	1.5B	1.5B	Portuguese
Tucano	tucano-2b4	Decoder	4k	2.4B	10–21M	Portuguese
Llama3	llama-3.1-8B-instruct	Decoder	128k	8B	21–42M	Multilingual
Phi3	phi3.5-mini-instruct	Decoder	4k	2B	13–25M	Multilingual
Phi4	phi-4	Decoder	16k	14.7B	28–56M	Multilingual
DeepseekR1	deepseek-reasoner	Decoder	64k	671B	0	Multilingual
GPT4o	gpt-4o-2024-11-20	Decoder	120k	?	0	Multilingual
Sabiá3	sabia-3	Decoder	32k	?	0	Portuguese

Table 1. Characteristics of automatic scoring models used in the experiments. #Params refers to the total number of parameters and #Train refers to the number of fine-tuned trained parameters for the AES task. Version refers to HuggingFace naming scheme.

3.1 Dataset

We use the dataset described in our previous work [Silveira et al., 2024], which contains 385 argumentative essays on 38 different topics (prompts), downloaded from publicly available websites. The websites present a selection of essays submitted by students who wish to practice for the essay writing part of the Brazilian National High-School Exam (*Exame Nacional do Ensino Médio*, ENEM, in Portuguese), a high-stakes large-scale standardized exam used as part of the entrance admission process in many universities in Brazil. In addition to the essay text, the dataset also contains the prompt given to students (containing the main topic and instructions) as well as some additional material such as supporting texts on which to base the argumentation. The prompt and additional material are made to simulate the ENEM essay test.

Following the official ENEM 2019 Grader’s Handbook, each essay was annotated by two independent experienced human graders on five different traits: fluency (C1), style (C2), argumentation and relevance to prompt (C3), cohesion (C4), and persuasiveness (C5). Each trait received, from each grader, a score in the scale 0, 40, 80, 120, 160, 200 — where higher is better. The website also provides annotations in the form of trait-specific scoring; however, there is no guarantee regarding the quality of these annotations or any explanation about how they were produced. Additionally, their agreement with either expert annotator was significantly lower than the agreement observed between the two expert annotators. For those reasons, and unlike previous work that has used that same corpus [Silveira et al., 2025], we do not use the annotations available on the website. That is, we assume that each essay is associated only with two (sets of) grades, each given by one of the two expert annotators.

In order to produce a fair evaluation of prompt-agnostic scoring, the dataset was shuffled and then split into training (500 instances), validation (132 instances) and test (138 instances) subsets such that essays related to a particular topic/prompt appeared exclusively in one of the data subsets.

3.2 Automatic Scoring Models

We evaluate a large variety of automatic scoring models with different inputs, architectures, learning strategies, and pre-training corpus. Table 1 summarizes the evaluated scoring models, which we detail next.

Feature-Based Models To serve as baselines, we evaluate the performance of two effective feature-based regression methods for AES: Linear Regression and Random Forest (as implemented in the Scikit-Learn Python package). To agree with our evaluation, the output of each regressor is rounded to the nearest point in the marking scale. We use 72 features computed by NILC-Metrix software [Leal et al., 2024], that include descriptors of style (e.g., verbs ratio, negation ratio, etc.), shape (number of words, sentences, paragraphs, etc.), and syntactic complexity indexes (Yngve and Frezier), to name a few. We also compare against a simple R₀ rule-based classifier that predicts the most frequent class label (or grade in our case). To account for the variance in the model selection of Random Forest Regressions, we train each regressor 100 times and report its average performance.

Encoder-Only Language Models Modern Large Language Models based on the Transformer Architectures are generally distinguished with respect to their pre-training strategy and inference use. Encoder-only models are pre-trained using masked language modeling, where the model learns to predict masked tokens from the input sequence [Devlin et al., 2019]. This strategy produces context-rich representations of input text that can be used for downstream tasks such as AES. We use the standard practice of supervised fine-tuning pre-trained encoder-only models on each trait-specific scoring task by adding an additional softmax classification layer with six outputs, then re-training the entire model (hence generating task-specific embeddings).

We test with several variants of BERT [Devlin et al., 2019]: the multilingual BERT (*mBERT*, for short) [Devlin et al., 2019], which was pre-trained on a large corpus of texts in several languages including Portuguese; *BERTimbau*, which was pre-trained on a corpus containing only texts in Por-

tuguese [Souza *et al.*, 2020]; BERTuguês [Mazza Zago and Agnoletti dos Santos Pedotti, 2024], which has some improvements over BERTimbau tokenizer and corpus filtering for training; and Albertina [Santos *et al.*, 2024], a larger variant also pre-trained in Portuguese corpora. Due to their reduced context-window size (512 tokens), encoder-only models are fed with only the text of the essay; essays longer than the context window are truncated.

The encoder-only models were trained on RTX A6000 GPU with a batch size of 16 for training, 1 step of gradient accumulation, and no gradient checkpointing [Chen *et al.*, 2016]. We used a weight decay rate of 0.01, warm-up of 0.1, and learning rate of $5 \cdot 10^{-5}$.

Small Language Models Models based on a decoder-only Transformer architecture are pre-trained using causal language modeling, where the model is trained to predict the next token based on a partial sentence [Radford *et al.*, 2018]. While such models were initially envisioned for generating language, the use of instruction tuning and other engineering feats has enabled them to generalize to unseen tasks, including high-level cognitive tasks [Brown *et al.*, 2020; OpenAI, 2022; Ouyang *et al.*, 2022]. Remarkably, current decoder-only models have much wider input context windows (4k–128k), which allow us to provide richer text input containing additional instructions and additional material. Small Language Models are decoder-only Language Models distilled from larger Language Models to achieve comparable performance with a reduced number of parameters [Gu *et al.*, 2023; Fu *et al.*, 2023; Javaheripi, 2023]. Despite their name, Small Language Models such as Llama [Touvron *et al.*, 2023] and Phi [Abdin *et al.*, 2024b,a] still require significant computational resources to be trained. Low Rank Adaptation (LoRA) is a parameter-efficient learning technique for training a Language Model by composing its parameters with a low-rank (smaller) parameter matrix [Hu *et al.*, 2022]. LoRA thus allows effectively learning Small Language Models under more affordable hardware.

The scoring guidelines for some traits such as style (C2) and argumentation (C3) specifically refer to additional material available to test takers such as supporting texts and the essay prompt. Other traits such as fluency (C1), cohesion (C4) and persuasiveness (C5) can in principle be scored with no such information, relying only on the essay text. We make use of the wider context available and evaluate fine-tuned small language models in two variants: providing only the student essay text (*essay-only*) as input and providing essay text, essay prompt and supporting texts (*full-context*). While in principle providing additional content should improve performance, it also makes learning less effective given the relatively small dataset size of fine tuning.

We use LoRA to fine-tune Tucano [Corrêa *et al.*, 2024], Llama3 and the Phi models for each trait-specific scoring task by adding an additional softmax classification layer with six class outputs. We follow common practices for the choice of hyperparameters of LoRA training, such as adopting an approximating rank $r \in \{8, 16\}$ and setting the scaling factor r by $\alpha = 2r$ [Hu *et al.*, 2022; Raschka, 2023]; the dropout rate was selected by experimentation in $[0.05, 0.1]$. All linear layers are treated as LoRA targets.

The fine-tuning of decoder-only models was performed using a H200 GPU, with a batch size of 8 for training and 4 for evaluation, with 2 gradient accumulation steps and gradient checkpointing [Chen *et al.*, 2016]. We used a weight decay rate of 0.01, warm-up of 0.1, and learning rate of $5 \cdot 10^{-5}$. Moreover, FlashAttentionV2 [Dao, 2023] was used to reduce memory footprint.

Zero-Shot Learning Large Language Models can perform zero-shot learning, which is the ability to specialize in unseen tasks without the need of re-training — the name is a bit misleading as no parameter learning is involved [Brown *et al.*, 2020]. This allows us to use them for scoring without costly supervised fine-tuning. Task specialization is implemented by specific prompting strategies, that is, by carefully designing the input sequence. We use the API interfaces of three different models: Sabiá3 [Abonizio *et al.*, 2025], which is trained exclusively in Portuguese, GPT4o [OpenAI *et al.*, 2024b] and DeepSeekR1 [DeepSeek-AI *et al.*, 2025], both of which were trained on multilingual corpora.

To generate adequate prompts to the zero shot learning models, we resort to official grading guidelines made available for the ENEM essay test.² In addition to the official Grader’s Handbook containing detailed instruction on how to grade each trait, the official exam also provides test takers with a significantly shorter and less technical Student’s Handbook that explains the meaning of each trait and summarizes the respective grading procedure. While the Student’s Handbook contains about 50 pages in total, the Grader’s Handbook contains more than 50 pages per trait.

Accordingly, we evaluate three prompting strategies for trait-specific scoring based on different grading guidelines. The *Student Guideline* and *Grader Guideline* use the explanation of the trait and the respective grading procedure available in the Student’s and the Grader’s Handbook, respectively. The *Mixed Guideline* uses the description of the trait from the Student’s Handbook and the respective grading procedure from the Grader’s Handbook. The rationale for this is to combine the succinctness of the Student’s Handbook when describing a trait with the precision of the Grader’s Handbook about the marking criteria. The *Grader Guideline* also differs in that we include an instruction to analyze the sub-criteria that make up the trait grading scoring rule before providing the trait-specific overall score. For instance, for Trait C2, after describing the trait and its scoring rule this prompt also asks the model to analyze whether the essay has three well-developed parts, whether it contains an original repertoire, whether it is relevant to the topic, etc. Table 2 shows the respective size of each prompting strategy in number of tokens as generated by GPT4 tokenizer, for each trait. The Grader Guideline is from 13% to 80% larger than the Student Guideline. Notably, even the smallest prompt size exceeds the context window of the encoder-only models we evaluate.

The guidelines for some traits such as style (C2) and argumentation (C3) specifically refer to additional material available to test takers such as supporting texts and the essay

²Downloaded from <https://www.gov.br/inep/pt-br/areas-de-atuacao/avaliacao-e-exames-educacionais/enem/outros-documentos>

Guideline	C1	C2	C3	C4	C5
Student	1,269	2,193	1,559	1,555	1,641
Mixed	1,323	2,193	1,501	1,696	1,619
Grader	1,757	2,663	1,768	2,886	3,032

Table 2. Trait-specific instruction size for the different guidelines (in number of tokens generated by GPT4o tokenizer).

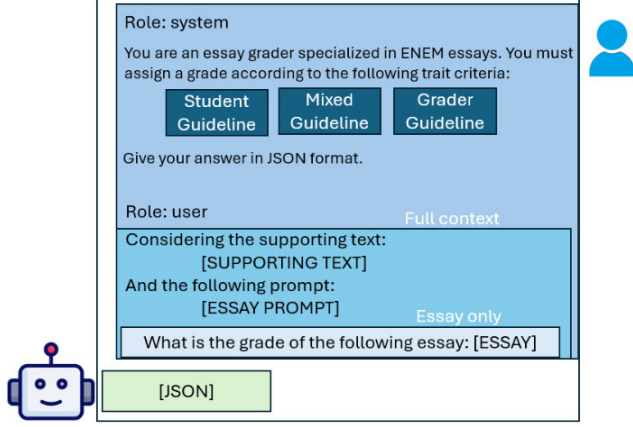


Figure 1. Prompt outline used for zero shot learning models.

prompt. To account for the effect of including or not relevant/irrelevant information, for each type of guideline and for each trait, we create two different prompting strategies by either including such extra information (*full-context*) or not including it (*essay-only*). Figure 1 outlines the prompting strategies used. To get a sense of the differences, we observe that the average essay size (in number of tokens produced by GPT4 tokenizer) is 338 compared to the average size of 1167 tokens of the full context (essay text, essay prompt and supporting texts).

We follow a Chain-of-Thought strategy [Wei *et al.*, 2023] by also asking the zero short learning models to justify their responses before presenting the final score. Since LLMs exhibit non-deterministic behavior, we repeat each inference 10 times and report the most frequent score (breaking ties arbitrarily).

3.3 Evaluation

AES models are typically evaluated using either inter-rater agreement metrics such as the Quadratic Weighted Kappa [Williamson *et al.*, 2012; Ramnarain-Seetohul *et al.*, 2022; Doewes *et al.*, 2023] or classification accuracy metrics for imbalanced data such as macro and weighted F1 score [Mello *et al.*, 2024].

The Quadratic Weighted Kappa (QWK) metric extends Cohen’s inter-rater agreement statistic to ordinal rating scales [de la Torre *et al.*, 2018]. The metric compares the agreement between two raters relative to agreement by chance, while penalizing disagreement by squared loss:

$$\kappa = 1 - \frac{\sum_{i=1}^N \sum_{j=1}^N w_{ij} O_{ij}}{\sum_{i=1}^N \sum_{j=1}^N w_{ij} E_{ij}},$$

where N is the number of rating categories (six, in our case), O_{ij} is the observed agreement matrix, indicating the number of times category i was predicted while category j was the

actual category, E_{ij} is the expected agreement matrix, representing the agreement expected by pure chance (which takes into account the relative frequency of each category), w_{ij} is the weight assigned to a disagreement between categories i and j and given by

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

QWK computes a value in $[-1, 1]$, with -1 indicating perfect disagreement, 0 indicating agreement by chance, and 1 indicating perfect agreement (hence the higher the value of a scorer, the better).

The *F1 score* computes the geometric weight of precision and recall of a binary classification prediction:

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}},$$

where

$$\text{Precision} = \frac{TP}{TP + FP} \quad \text{and} \quad \text{Recall} = \frac{TP}{TP + FN}.$$

The metric can be extended to multi-class classification by considering each class i as a binary prediction task with its corresponding $F1_i$ measure, then averaging out. The *weighted F1* score weights each $F1_i$ score by the relative class label frequency. Instead, the *macro F1* score uses a simple average:

$$F1_{\text{weighted}} = \sum_{i=1}^N \frac{n_i}{\sum_{j=1}^N n_j} F1_i, \quad F1_{\text{macro}} = \sum_{i=1}^N \frac{1}{N} F1_i,$$

where n_i is the number of true instances for class i . Both weighted F1 and macro F1 given values in $[0, 1]$, with 1 denoting perfect classification accuracy and 0 denoting complete misclassification.

The class label distribution of our dataset follows a bell curve shape, with the vast majority of instances being rated close to the middle of the range and the lower and higher classes being rare. Accordingly, weighted F1 emphasizes model performance on the middle range classes, with tail distribution classes having little to no effect on the score. On the other hand, the macro F1 tends to proportionally emphasize performance on the less frequent classes, and lead to much higher dataset variance.

In general, QWK, macro F1 and weighted F1 metrics provide different and complementary views and insights about a scoring model’s performance.

Ground-Truth Ambiguity Since each essay in the dataset is annotated with two grades, for any metric we actually have two performance indicators. We thus compute an average metric for each trait and metric by taking the mean of each metric with respect to each grader. Because each scoring model provides a single score, for any metric its maximum performance falls below the maximum value (e.g., for any essay with two disagreeing annotated grades, a predicted score necessarily makes at least one mistake). In order to provide a comparative indicator of (unrealistic) optimum performance

for each metric, we simulate predictions based on the ground-truth scores (hence, information that is unavailable to models).

We call *GradeA* (resp., *GradeB*) the mean metric for a scorer that always agrees with Grader A (resp., Grader B). This is clearly suboptimal but useful to compare if a scorer is perhaps aligning more closely with one Grader’s labels than the other one’s. The *LessFrequentGrade* selects for each essay and trait the less frequent label among the scores assigned by Grader A and B. This tends to increase QWK and macro F1, which emphasize agreement on less frequent labels. The *MostFrequentGrade* instead selects the most frequent among the labels given by Grader A and B, per instance. Finally, *MeanGrade* rounds the mean of the scores of Grader A and B, to make it a valid class label. The mean is rounded either up or down such that the less frequent grade is produced.

To illustrate the different behaviors of each upper bound metric described above, suppose we have a trait-specific scoring task with the following count distribution of rates in the evaluation set (from lowest to highest score): [10, 20, 30, 15, 10, 7]. If an instance was assigned the third lowest score by Grader A (which occurs 30 times in the set) and was assigned the highest score by Grader B (which occurs 7 times) then *GradeB* and *LessFrequentGrade* will output the same as Grader B, *GradeA* and *MostFrequentGrade* will output the same as Grader A, and *MeanGrade* computes the mean of scores of Grader A and Grader B and rounds to the less frequent, which in this case is the second highest score of the range.

4 Results and Discussion

We now present and discuss the empirical results of our analysis. We first evaluate models inside their class, in order to obtain insight into the impact of different architectures, pre-training corpus and prompt engineering. We then perform a cross-comparison among different model classes.

4.1 The Impact of Encoder-Only Base Model

We start by analyzing the performance of encoder-only language models for different base modes (i.e., the pre-trained model used). Table 3 shows the relevant metrics (macro F1, weighted F1 and QWK) of trait-specific scorers obtained by fine-tuning different pre-trained encoder-only models. The models are sorted in decreasing size (in number of parameters).

The first thing we note is that the three metrics are highly correlated, yet provide somewhat different rankings of the trait-specific scorers. For example, for trait C1, weighted F1 ranks models (from best to worst) as *Albertina* > *BERTimbau Large* > *mBERT* > *BERTuguês* > *BERTimbau Base*, whereas QWK ranks models as *Albertina* = *BERTimbau Large* > *BERTuguês* > *BERTimbau Base* > *mBERT*. As another example, notice that *BERTimbau Base* has the highest QWK value for trait C2 while the highest macro F1 is obtained by *BERTimbau Large*.

Second, no single model consistently outperforms the others across traits. In fact, each model obtained the best perfor-

mance for some combination of trait and metric. Thus, there is no single model that can be discarded or recommended, and different variants of those models seem to adapt better to different traits. Overall, *BERTimbau Base* shows a good compromise of performance across the different traits.

The size of models (in number of parameters) correlates strongly with model performance for traits C1 and C4 (especially regarding weighed F1), but much less so for traits C2, C3 and C5. By inspecting the best performance across traits, we observe a significant reduction for traits C2 and C3 compared to other traits, especially in regards to QWK. This can be explained by the fact that according to the grading guidelines such traits are evaluated in comparison to the additional material (essay prompt and supporting texts) to which the models do not have access. Other traits do not have such a dependency on unavailable information. Curiously, while *BERTimbau Base* achieves a relatively high value of QWK for C5, its values for the macro and weighted F1 metrics (which are still the highest for this trait) are inferior to the respective values it obtains for C2 (where the same model is one of the top performers). This gap suggests that trait C2 (Style) can be harder than trait C5 (Persuasiveness): for C5 the models usually predict a neighboring score (high QWK) even when they miss the exact class (lower F1s), whereas for C2 the errors are often farther away, depressing all three metrics.

To investigate if differences in performance metrics are statistically significant we obtained 95% confidence intervals by bootstrapping. We deem a result statistically significant to another if the lower end of the confidence interval of the latter higher than the upper end of the latter. Note that this does not account for multiple testing and hence is likely to produce more false positives than expected. Doing so shows that no method is statistically significantly superior for any trait and metric. However, some methods do perform statistically significantly worse than the best performing method, for a fixed trait and metric. We indicate that worse performance in *italic* in Table 3. Thus for traits C1, C3 and C4 we see that differences in performance are not statistically significant. For trait C2, some methods do perform worse but only according to some metrics. The most striking difference is observed regarding trait C5, where the two variants of *BERTimbau* perform statistically significantly better than all other methods. Indeed, other methods perform rather poor in this trait. While the difference between the base and large variants is pronounced, it is still not enough to determine statistical significance.

4.2 What about Small Language Models

We now turn our attention to pre-trained Small Language Models, fine tuned with the LoRA parameter-efficient learning. Due to the larger context size window, it was possible to experiment with both essay only and full context configurations.

Table 4 presents the performance of small language models evaluated under the essay only configuration. The models are ordered by ascending parameter count, ranging from the 2.4B Tucano to the 14.7B Phi4. All models were fine-tuned using LoRA for efficiency and comparability.

	C1: Fluency			C2: Style			C3: Argument			C4: Cohesion			C5: Persuasion		
Model	M	W	Q	M	W	Q	M	W	Q	M	W	Q	M	W	Q
Albertina	.55	.71	.68	.12	.18	.24	.23	.32	.26	.36	.59	.54	.15	.17	.04
BERTimbau Large	.52	.69	.68	.33	.43	.32	.19	.26	.24	.48	.61	.60	.29	.37	.46
mBERT	.40	.59	.52	.18	.27	.22	.25	.32	.35	.35	.55	.50	.08	.09	.00
BERTuguês	.44	.56	.62	.14	.19	.33	.25	.36	.29	.35	.51	.54	.23	.28	.36
BERTimbau Base	.42	.52	.60	.26	.37	.36	.23	.28	.35	.38	.55	.55	.29	.37	.63
Best	.55	.71	.68	.33	.43	.32	.25	.36	.29	.48	.61	.60	.29	.37	.63

Table 3. Test-set performance of encoder-only models. M: macro F1, W: weighted F1, Q: Quadratic Weighted Kappa. The last row repeats the best performance for each trait and metric. Italicized values indicate statistically significantly worse performance relative to the best performing method according to a 95% bootstrapping confidence interval.

	C1: Fluency			C2: Style			C3: Argument			C4: Cohesion			C5: Persuasion		
Model	M	W	Q	M	W	Q	M	W	Q	M	W	Q	M	W	Q
Tucano	.52	.64	.66	.15	.24	.22	.25	.35	.36	.34	.57	.49	.25	.30	.58
Phi3	.33	.40	.45	.21	.26	.28	.24	.28	.22	.29	.50	.27	.19	.24	.25
Llama3	.49	.67	.65	.26	.36	.36	.25	.37	.37	.24	.53	.39	.26	.28	.29
Phi4	.50	.61	.67	.19	.25	.21	.24	.31	.33	.30	.58	.48	.44	.49	.59
Best	.52	.64	.66	.26	.36	.36	.25	.37	.37	.34	.57	.49	.44	.49	.59

Table 4. Test-set performance of small language models on essay-only prompts. M: macro F1, W: weighted F1, Q: Quadratic Weighted Kappa.

The first observation is that parameter count alone is not a reliable predictor of performance. Despite being the smallest model in the group, Tucano achieves the best or tied-best scores in three of the five traits, outperforming Phi-3 across all metrics and remaining competitive with much larger models such as Llama3 and Phi4. This suggests that monolingual pretraining, when aligned with the target evaluation language, can partially offset the limitations imposed by scale—especially in low-context settings like this one.

Across traits, no model dominates consistently. Llama3 emerges as the top performer for Style (C2) and Argumentation (C3), while Phi4 achieves the highest scores for Persuasion (C5). Interestingly, Tucano outperforms all others on Fluency (C1) and Cohesion (C4), traits that depend more on linguistic form than on content references. These results echo a pattern observed in the before: rather than a single best model, we observe trait-dependent specialization.

Scores on C2 are particularly low across all models, with macro F1 hovering around 0.15–0.26. This mirrors earlier findings that traits requiring access to external materials (such as the essay prompt) are inherently harder to judge when the input is limited to the essay itself. In contrast, traits such as C1 and C4, which focus more on grammar, flow, and internal consistency, benefit less from additional context and show stronger performance in this scenario.

Overall, the results reinforce the idea that scale is helpful but not decisive.

Table 5 reports the scores obtained when each model has access to both the essay prompt and the supporting texts. The 14-billion-parameter Phi4 dominates, achieving the highest score in every trait–metric combination. The performance gap is particularly large for Style (C2) and Persuasion (C5), where Phi4 surpasses the runner-up by 0.27 and 0.20 macro-F1 points, respectively.

These gains highlight the benefits of scale within the same decoder-only architecture: Phi4 is a direct up-sizing of Phi3 (from 3.8B to 14B parameters) with roughly an order-of-

magnitude increase in training tokens, yet no major structural changes. In contrast, the smaller Phi3 collapses on C2 (QWK −0.03), a pattern consistent with the lost-in-the-middle effect — where long-context models struggle to retrieve information located far from the input boundaries [Liu *et al.*, 2024]. The much larger Phi4 appears far less susceptible to this limitation.

Tucano models remain competitive on grammar-related traits such as Cohesion (C4), but they consistently underperform compared to both Phi4 and Llama3.

Table 6 presents the differences between the full-context and essay-only setups for each trait–metric–model triple. Positive values indicate that including the full prompt and supporting materials yields improvements, while negative values signal performance degradation.

Among the evaluated models, Phi4 stands out as the model that benefits most from additional context, particularly for traits C2 (Style) and C3 (Argument). For example, it gains as much as +0.39 in QWK on C2 and +0.24 on C3, with consistent boosts in macro and weighted F1 as well. These improvements highlight how larger models with broader context windows can leverage discourse-level features such as stylistic variation and argumentative development, which are likely distributed across the supporting texts and essay prompt.

However, the inclusion of context is not universally beneficial. Fluency (C1) consistently deteriorates across all models and metrics — macro F1, weighted F1, and QWK all register negative or neutral deltas. This suggests that exposing the model to longer or noisier input sequences may harm it.

Cohesion (C4) shows more mixed behavior. While Llama3 achieves moderate improvements in both macro F1 and QWK, Phi3 and Tucano display modest losses, particularly in weighted F1. This variability may reflect architectural differences in how well models capture long-range dependencies and discourse structure.

Persuasion (C5), on the other hand, suffers across the board. All models record losses, with Tucano showing a par-

	C1: Fluency			C2: Style			C3: Argument			C4: Cohesion			C5: Persuasion		
Model	M	W	Q	M	W	Q	M	W	Q	M	W	Q	M	W	Q
Tucano	.43	.56	.57	.19	.27	.24	.24	.31	.24	.37	.51	.52	.19	.24	.24
Phi3	.19	.29	.28	.14	.22	-.03	.28	.35	.44	.11	.20	.08	.18	.19	.19
Llama3	.41	.57	.60	.20	.25	.33	.25	.28	.32	.39	.55	.54	.21	.20	.18
Phi4	.48	.63	.67	.42	.52	.60	.37	.38	.57	.37	.58	.55	.29	.32	.52
Best	.48	.63	.67	.42	.52	.60	.37	.38	.57	.37	.58	.55	.29	.32	.52

Table 5. Test-set performance of small language models on full context prompts. M: macro F1, W: weighted F1, Q: Quadratic Weighted Kappa.

	C1: Fluency			C2: Style			C3: Argument			C4: Cohesion			C5: Persuasion		
Model	ΔM	ΔW	ΔQ	ΔM	ΔW	ΔQ	ΔM	ΔW	ΔQ	ΔM	ΔW	ΔQ	ΔM	ΔW	ΔQ
Tucano	-.09	-.08	-.09	.04	.03	.02	-.01	-.04	-.12	.03	-.06	.03	-.06	-.06	-.34
Phi3	-.14	-.11	-.17	-.07	-.04	-.31	.04	.07	.22	-.18	-.30	-.19	-.01	-.05	-.06
Llama3	-.08	-.10	-.05	-.06	-.11	-.03	.00	-.09	-.05	.15	.02	.15	-.05	-.08	-.11
Phi4	-.02	.02	.00	.23	.27	.39	.13	.07	.24	.07	.00	.07	-.15	-.17	-.07

Table 6. Difference between full context and essay only prompts of Small Language Models: Positive values indicate a gain from supplying the full conversational context.

ticularly steep decline in QWK (−0.34) and Phi4 facing a substantial drop in both macro and weighted F1. These declines are also indicative of the lost-in-the-middle effect.

Overall, these results reveal a nuanced trade-off: supplying additional context enhances performance on traits dependent on global stylistic and argumentative features, but it can inadvertently impair fluency and high-level persuasive reasoning.

4.3 The Impact of Prompt Engineering

We analyze the three grading–guideline configurations (Student, Grader and Mixed) within each zero learning model. Our goal here is *not* to crown an overall winner but to understand how the choice of guideline interacts with model architecture and the presence or absence of additional context (which might be required or irrelevant, depending on the trait). Table 7 shows results when only the essay text is provided in the input and Table 8 shows results when essay prompt and supporting texts are also provided.

For each model, the longer and more detailed Grader guideline leads to reduced performance for language-centric traits such as C1 and C2, while performing better for the other traits, both when providing only the essay or the full context. This is an indication that the models might be suffering from a lost-in-the-middle effect. We see that for the larger contexts that contain also the additional material, the overall differences among guidelines reduce.

We also see that for C2 and C3, performances increase significantly when providing the full context in comparison to only the essay, as expected, as those traits explicitly evaluate elements from the additional material. Notably, there is a small gain in performance also for C4 and C5. On the other hand, overall performance for C1 is reduced when additional and irrelevant content is given in the input, likely due to the increased context.

In the essay only setting the Student guideline often outperforms alternatives for discourse-level traits (C3–C5), presumably because it is more oriented to what the student should do.

When the rubric refers to additional material (notably traits C2 and C3), the Mixed guideline produces the highest QWK value, especially for DeepseekR1 and Sabiá3. This suggests that seeking a balance between level of detail and length might pay off.

Overall, the results suggest that optimal prompting is trait- and model-specific. Yet, providing the most appropriate level of detail (i.e., full context for C2 and C3 and only the essay for the others) leads to a good balance of accuracy and cost. It is also interesting to note that Sabiá, the only model trained exclusively on Portuguese, outperforms the others only for trait C1, which analyzes the presence of misspelling, grammar, etc., which are very specific to language. Monolingual training seems less important for other less language specific traits.

To better analyze the impact of including additional material, we show in Table 9 the relative differences between providing only the essay or the full context for each model and guideline.

As expected, including the additional material consistently benefits trait C2 across all models, especially under the Mixed and Student guidelines, where the gains are always statistically significant (indicated in bold). Although not always statistically significant, trait C3 also shows meaningful improvements in every scenario, matching expectations. With the exception of GPT4o, traits C4 and C5 exhibit negligible gains or slight declines (with a slight advantage in favor of the full context scenario), which is reasonable given the nature of those traits. Finally, for C1, DeepseekR1, experiences substantial drops in QWK and weighted F1, possibly because of the lost-in-the-middle effect that GPT4o and Sabiá3 appear to mitigate.

These findings reinforce the broader claim that *prompt engineering is not merely a pre-processing convenience but a critical aspect of development* in automated essay scoring, underscoring the importance of carefully aligning the prompt with both the target trait and the capabilities of the underlying language model. For example, guideline selection can swing QWK by up to +0.27 (Sabiá3, C1, full context) without changing model weights or adding training data. The

Model	Guideline	C1: Fluency			C2: Style			C3: Argument			C4: Cohesion			C5: Persuasion		
		M	W	Q	M	W	Q	M	W	Q	M	W	Q	M	W	Q
DeepseekR1	Mixed	.17	.25	.44	.05	.05	.03	.19	.19	.39	.21	.25	.48	.31	.34	.55
	Student	.11	.17	.36	.13	.26	-.02	.32	.34	.38	.27	.39	.52	.37	.43	.55
	Grader	.07	.12	.21	.01	.01	-.01	.27	.34	.41	.33	.53	.53	.35	.39	.62
GPT4o	Mixed	.21	.45	.49	.16	.22	.28	.20	.27	.28	.29	.39	.51	.32	.35	.54
	Student	.22	.43	.51	.16	.27	.20	.26	.30	.38	.30	.40	.51	.28	.31	.55
	Grader	.09	.25	.29	.13	.19	.18	.11	.19	.11	.19	.36	.45	.21	.22	.52
Sabiá3	Mixed	.23	.42	.53	.11	.12	.06	.28	.33	.34	.24	.43	.55	.33	.38	.59
	Student	.35	.66	.69	.11	.11	.01	.23	.32	.30	.31	.53	.52	.29	.35	.51
	Grader	.09	.25	.29	.13	.19	.18	.11	.19	.11	.19	.36	.45	.21	.22	.52
Best		.35	.66	.69	.16	.22	.28	.27	.34	.38	.31	.53	.52	.37	.43	.55

Table 7. Trait-specific performance of zero-shot learning scorers for different prompting strategies providing only the essay. M: macro F1, W: weighted F1, Q: Quadratic Weighted Kappa. The last row repeats the best performance for each trait and metric. Boldface values indicate statistically significant superior performance relative to other variants of the same model class (e.g. according to QWK Sabiá3 with Student Guideline prompting is statistically significantly superior to Sabiá3 with either Mixed or Grader Guideline prompting.)

Model	Guideline	C1: Fluency			C2: Style			C3: Argument			C4: Cohesion			C5: Persuasion		
		M	W	Q	M	W	Q	M	W	Q	M	W	Q	M	W	Q
DeepseekR1	Mixed	.05	.06	.20	.32	.43	.53	.30	.30	.60	.29	.46	.55	.35	.38	.60
	Student	.04	.06	.35	.26	.37	.42	.44	.47	.73	.22	.37	.49	.37	.40	.58
	Grader	.04	.08	.20	.27	.34	.51	.30	.41	.52	.37	.60	.56	.37	.42	.61
GPT4o	Mixed	.25	.52	.55	.31	.44	.56	.27	.33	.45	.26	.47	.50	.24	.26	.48
	Student	.21	.42	.48	.34	.43	.52	.27	.30	.53	.26	.37	.49	.17	.19	.35
	Grader	.25	.54	.53	.32	.43	.47	.27	.30	.57	.27	.45	.50	.14	.15	.32
Sabiá3	Mixed	.24	.50	.50	.33	.43	.50	.33	.37	.54	.31	.49	.53	.28	.32	.50
	Student	.35	.60	.58	.28	.42	.47	.21	.24	.45	.28	.43	.39	.27	.28	.54
	Grader	.10	.26	.31	.16	.24	.28	.20	.29	.21	.18	.38	.38	.22	.24	.51
Best		.35	.60	.58	.32	.43	.53	.44	.47	.73	.37	.60	.56	.35	.38	.60

Table 8. Trait-specific performance of zero-shot learning scorers for different prompting strategies providing the full context. M: macro F1, W: weighted F1, Q: Quadratic Weighted Kappa. Boldface values indicate statistically significant superior performance relative to other variants of the same model class (e.g. Sabiá3 with Student Guideline prompting is statistically significantly superior to Sabiá3 with either Mixed or Grader Guideline prompting.)

Model	Guideline	C1			C2			C3			C4			C5		
		Δ M	Δ W	Δ QWK	Δ M	Δ W	Δ QWK	Δ M	Δ W	Δ QWK	Δ M	Δ W	Δ QWK	Δ M	Δ W	Δ QWK
DeepseekR1	Mixed	-0.12	-0.19	-0.24	+0.27	+0.38	+0.50	+0.11	+0.11	+0.21	+0.08	+0.21	+0.07	+0.04	+0.04	+0.05
	Student	-0.07	-0.11	-0.01	+0.13	+0.11	+0.44	+0.12	+0.13	+0.35	-0.05	-0.02	-0.03	+0.00	-0.03	+0.03
	Grader	-0.03	-0.04	-0.01	+0.26	+0.33	+0.52	+0.03	+0.07	+0.11	+0.04	+0.07	+0.03	+0.02	+0.03	-0.01
GPT4o	Mixed	+0.04	+0.07	+0.06	+0.15	+0.22	+0.28	+0.07	+0.06	+0.17	-0.03	+0.08	-0.01	-0.08	-0.09	-0.06
	Student	-0.01	-0.01	-0.03	+0.18	+0.16	+0.32	+0.01	+0.00	+0.15	-0.04	-0.03	-0.02	-0.11	-0.12	-0.20
	Grader	+0.16	+0.29	+0.24	+0.19	+0.24	+0.29	+0.16	+0.11	+0.46	+0.08	+0.09	+0.05	-0.07	-0.07	-0.20
Sabiá3	Mixed	+0.01	+0.08	-0.03	+0.22	+0.31	+0.44	+0.05	+0.04	+0.20	+0.07	+0.06	-0.02	-0.05	-0.06	-0.09
	Student	+0.00	-0.06	-0.11	+0.17	+0.31	+0.46	-0.02	-0.08	+0.15	-0.03	-0.10	-0.13	-0.02	-0.07	+0.03
	Grader	+0.01	+0.01	+0.02	+0.03	+0.05	+0.10	+0.09	+0.10	+0.10	-0.01	+0.02	-0.07	+0.01	+0.02	-0.01

Table 9. Relative change when moving from essay-only to full-context prompting. Boldface values indicate statistically significant superior performance when comparing the same Model-Guideline pair (e.g. DeepseekR1 C2 has a statistical significant difference for Mixed Guideline)

use of additional material can boost QWK by 0.50 points (DeepseekR1, C2).

4.4 The Impact of Architecture and Learning Strategy

Finally, we compare the differences across different classifier architectures, learning strategies, and available information. The goal here is to compare best-performing strategies, having feature-based classifiers serve as baselines. The results appear in Table 10.

The first thing to notice is that, overall, feature-based methods perform significantly worse than the best performing scorer based on language models. The performance of encoder-only, fine-tuned small language models and zero-shot learners for C1 are comparable, with slight advantage to either encoder-only or zero-shot learners depending on the metric. The situation is somehow similar for C2, except for the higher value of QWK of the best small language model (although the difference is not statistically significant). Considering trait C3, zero shot learners exhibit higher values for all metrics, possibly due to the emergent reasoning abilities of the larger language models. For C4 the situation is much less clear, as even Random Forest classifiers obtain good results for the weighted F1 (but perform quite worse in regards to QWK). Feature-based models are particularly poor at trait C5, where again the situation is less clear among the different language models.

All in all, one sees that zero shot learners obtain good performance without the hassle of parameter learning tuning (but with careful prompt-engineering), although at much higher costs and concerns due to their proprietary nature. Encode-only models do obtain best or near-best performance for traits C1, C4 and C5, with significantly less computational cost. Their main drawback seems to be the reduced context window, which prevents important information to be included in the input and seems to be particularly important for traits C2 and C3. That can be mitigated by more recent BERT-like models with larger context windows, such as ModernBERT [Warner *et al.*, 2024]. However, at the time of this writing, no such models existed for Portuguese. Fine-tuned small language models do appear as a compromise, being particularly good at trait C2.

The comparison with the oracle-like upper bound metrics described in Section 3 allow us to assess these results in more absolute terms. Remember that such values are obtained by sampling from the two grades that annotate each essay, hence provide a measure of best performing value for each metric. We see that the difference between the best obtained results and the upper bounds is relative small for most of traits and metrics, suggesting that there is small room left to further improvement in those cases. The difference is more pronounced for traits C1, C2 and C5, suggesting also that future work should focus on improving performance for such traits. In short, Fluency (C1) and Cohesion (C4) are approaching human inter-rater agreement level, while Style (C2), Argumentation (C3), and Persuasiveness (C5) still have substantial headroom.

The comparison with upper bounds also suggest that relative performances might in part be credited to the particular

strategy considered by each model in selecting a label in face of the ambiguity in training data.

In sum, modern large language models seem to excel at AES, although the particular architecture, learning strategy and input needs to be selected specifically for the type of trait being evaluated.

4.5 CO2 Emission Related to Experiments

As a final aspect of our analysis, we evaluated the environmental impact of our experimental setup. As mentioned in 3.2, we used an RTX A6000 GPU to train the encoder models and an H200 to train all decoder models. Machines were rented through vast.ai,³ and we conducted a detailed analysis of our experiments using the codecarbon library [Courtney *et al.*, 2024].

In total, we used approximately 44 hours of computation on RTX A6000 (TDP 300W) and H200 GPUs (TDP 700W) to train all encoder- and decoder-based models, respectively. This setup resulted in a total of 65 experiments.

The total estimated emissions amount to 3.42 kg CO₂eq, with 0% of this footprint directly offset. This corresponds to roughly 32 kWh of energy consumption.

Surprisingly, fine-tuning even large models in our case had a low environmental impact: the total CO₂ emissions are approximately equivalent to driving a gasoline car for 24 km.⁴

5 Conclusion

Automatic Essay Scoring in Portuguese is still in its early days. So far, most of the research has revolved around creating datasets along with defining the baseline performances. These baseline performances were usually defined through feature-based methods or deep neural networks.

In this work, we conducted an in-depth evaluation of twelve models that differ in architecture (encoder-only, small language model, zero-shot), fine-tuning strategy (full fine-tuning, LoRA), and pretraining data (Portuguese vs. multilingual). These models were assessed on their ability to grade essays written in preparation for the Brazilian national entrance exam, which are scored across five distinct traits.

Our results show that an encoder-only model trained in Portuguese and fine-tuned achieved the best overall performance on the first traits, with macro F1 of 0.55, outperforming the second and third best models (0.52 and 0.35, respectively), weighted F1 of 0.71 (vs. 0.64 and 0.66), and the second-highest Quadratic Weighted Kappa (QWK). For the second trait, a decoder-only model fine-tuned with LoRA and enhanced with essay prompt and supporting text achieved the highest QWK score of 0.60. In the third trait, a zero-shot model stood out with a QWK of 0.73, surpassing the second-best model by a substantial margin of 0.16. For the fourth trait, the BERTimbau Large encoder-only model yielded the best QWK (0.60) along with a high weighted F1 score (0.61). Finally, for the fifth trait, the top-performing model was again a encoder model pretrained in Portuguese, achieving a QWK of 0.63.

³<https://vast.ai>

⁴Estimated via: <https://www.openco2.net/en/co2-converter>

	C1: Fluency			C2: Style			C3: Argument			C4: Cohesion			C5: Persuasion		
Model	M	W	Q	M	W	Q	M	W	Q	M	W	Q	M	W	Q
R ₀	.12	.20	.00	.11	.20	.00	.09	.17	.00	.14	.39	.00	.05	.05	.00
Linear Regressor	.41	.53	.36	.13	.23	.32	.17	.27	.26	.30	.57	.45	.15	.17	.03
Random Forest	.32	.56	.41	.14	.22	.22	.21	.29	.35	.35	.64	.48	.11	.15	.12
Best Encoder	.55	.71	.68	.33	.43	.32	.25	.36	.29	.48	.61	.60	.29	.37	.63
Best SLM	.52	.64	.67	.42	.52	.60	.37	.38	.57	.37	.58	.55	.44	.49	.59
Best ZSL	.35	.66	.69	.32	.43	.53	.44	.47	.73	.37	.60	.56	.37	.43	.60
Best	.55	.71	.69	.42	.52	.60	.44	.47	.73	.48	.64	.60	.44	.49	.63
GradeA	.72	.84	.80	.64	.73	.83	.66	.71	.84	.67	.84	.72	.72	.77	.85
GradeB	.72	.83	.80	.64	.75	.83	.66	.70	.84	.67	.85	.72	.72	.75	.85
LessFrequent	.75	.84	.83	.63	.78	.85	.66	.69	.85	.70	.86	.77	.75	.77	.85
MeanGrade	.71	.82	.85	.55	.71	.89	.62	.65	.88	.66	.83	.81	.68	.68	.90
MostFrequent	.64	.81	.75	.56	.71	.79	.61	.69	.81	.55	.83	.64	.62	.75	.84

Table 10. Test-set performance of best LLM for each type, trait and metric against feature-based methods. M: macro F1, W: weighted F1, Q: Quadratic Weighted Kappa. SLM: Small Language Models. ZSL: Zero-Shot Learners. Boldface indicate advantage is statistically significant according to 95% Bootstrapping Confidence Intervals.

To conclude, we conducted a straightforward analysis to estimate the environmental impact of fine-tuning these models. We found that the total emissions from all training experiments (approximately 3.5 kg CO₂eq) are roughly equivalent to the carbon footprint of a single person driving 25 kilometers by car.

Declarations

Authors' Contributions

AB was the main contributor and writer of this manuscript, as well as the responsible for the majority of the coding including: training of the LLM models and the code for querying the API-based models. ICS contributed to the writing, experiment design, reviewing, and experiments involving feature-based models and some zero-shot learners. DDM contributed to the experiment design, reviewing and editing.

Competing interests

The authors declare that they have no competing interests.

Acknowledgements

We would like to thank Guilherme Yambanis for his contributions in the early stages of development of this research.

Funding

This research was funded by the Center for Artificial Intelligence (C4AI-USP), the São Paulo Research Foundation (FAPESP grants #2019/07665-4 and 2022/02937-9) and by IBM Corporation. This work was also supported by CNPq grant no. 305136/2022-4 and CAPES Finance Code 001.

Availability of data and materials

The dataset and models generated during the current study are available in <https://huggingface.co/collections/kamel-usp/jbcs2025-67d5e73a4b89c1f0c878159c>. Experiments and

code used are available in <https://github.com/kamel-usp/jbcs2025>.

References

- Abdin, M., Aneja, J., Behl, H., Bubeck, S., Eldan, R., Gunasekar, S., Harrison, M., Hewett, R. J., Javaheripi, M., Kauffmann, P., *et al.* (2024a). Phi-4 technical report. *arXiv preprint arXiv:2412.08905*. DOI: 10.48550/arXiv.2412.08905.
- Abdin, M., Jacobs, S. A., Awan, A. A., Aneja, J., Awadallah, A., *et al.* (2024b). Phi-3 technical report: A highly capable language model locally on your phone. DOI: 10.48550/arXiv.2404.14219.
- Abonizio, H., Almeida, T. S., Laitz, T., Junior, R. M., Bonás, G. K., Nogueira, R., and Pires, R. (2025). Sabiá-3 technical report. Available at: <https://arxiv.org/abs/2410.12049>.
- Alikaniotis, D., Yannakoudakis, H., and Rei, M. (2016). Automatic text scoring using neural networks. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 715–725. DOI: 10.18653/v1/p16-1068.
- Amorim, E. and Veloso, A. (2017). A multi-aspect analysis of automatic essay scoring for Brazilian Portuguese. In *Proceedings of the Student Research Workshop at the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 94–102. DOI: 10.18653/v1/e17-4010.
- Attali, Y. and Burstein, J. (2006). Automated essay scoring with e-rater v.2. *Journal of Technology, Learning, and Assessment*, 4(3). Available at: <https://ejournals.bc.edu/index.php/jtla/article/view/1650>.
- Bazelato, B. S. and Amorim, E. (2013). A bayesian classifier to automatic correction of Portuguese essays. In *Conferência Internacional sobre Informática na Educação (TISE)*, volume 18, pages 779–782. Available at: <https://www.tise.cl/volumen9/TISE2013/779-782.pdf>.
- Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., *et al.* (2020). Language models are few-shot learners. In

- Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.. DOI: 10.48550/arXiv.2005.14165.
- Burstein, J., Chodorow, M., and Leacock, C. (2004). Automated essay evaluation: the criterion on-line writing service. *AI Mag.*, 25(3):27–36. DOI: 10.1609/aimag.v25i3.1774.
- Chang, L.-H. and Ginter, F. (2024). Automatic short answer grading for finnish with chatgpt. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 23173–23181. DOI: 10.1609/aaai.v38i21.30363.
- Chen, T., Xu, B., Zhang, C., and Guestrin, C. (2016). Training deep nets with sublinear memory cost. *CoRR*, abs/1604.06174. Available at: <http://arxiv.org/abs/1604.06174>.
- Corrêa, N. K., Sen, A., Falk, S., and Fatimah, S. (2024). Tucano: Advancing neural text generation for portuguese. DOI: 10.1016/j.patter.2025.101325.
- Courty, B., Schmidt, V., Luccioni, S., Goyal-Kamal, MarionCoutarel, Feld, B., Lecourt, J., LiamConnell, Saboni, A., Inimaz, supatomic, Léval, M., Blanche, L., Cruveiller, A., ouminasara, Zhao, F., Joshi, A., Bogroff, A., de Lavoreille, H., Laskaris, N., Abati, E., Blank, D., Wang, Z., Catovic, A., Alencon, M., Michał Stechły, Bauer, C., de Araújo, L. O. N., JPW, and MinervaBooks (2024). mlco2/codecarbon: v2.4.1. DOI: 10.5281/zenodo.11171501.
- Cummins, R. and Rei, M. (2018). Neural multi-task learning in automated assessment. *arXiv preprint arXiv:1801.06830*. DOI: 10.48550/arXiv.1801.06830.
- Dao, T. (2023). Flashattention-2: Faster attention with better parallelism and work partitioning. Available at: <https://arxiv.org/abs/2307.08691>.
- de la Torre, J., Puig, D., and Valls, A. (2018). Weighted kappa loss function for multi-class classification of ordinal data in deep learning. *Pattern Recognition Letters*, pages 144–154. Machine Learning and Applications in Artificial Intelligence. DOI: 10.1016/j.patrec.2017.05.018.
- de Lima, T. B., Freitas, E., and Macario, V. (2024). Aesvoting: Automatic essay scoring with bert and voting classifiers. In *Proceedings of the 16th International Conference on Computational Processing of Portuguese-Vol. 2*, pages 6–9. Available at: <https://aclanthology.org/2024.propor-2.2/>.
- DeepSeek-AI, Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., and et al. (2025). Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. DOI: 10.48550/arXiv.2501.12948.
- Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. In *North American Chapter of the Association for Computational Linguistics*. DOI: 10.48550/arXiv.1810.04805.
- Doewes, A., Kurdhi, N. A., and Saxena, A. (2023). Evaluating quadratic weighted kappa as the standard performance metric for automated essay scoring. In *Proceedings of the 16th International Conference on Educational Data Mining*, pages 103–113. International Educational Data Mining Society. Available at: <https://educationaldatamining.org/edm2023/proceedings/2023.EDM-long-papers.9/index.html>.
- Dong, F. and Zhang, Y. (2016). Automatic features for essay scoring - an empirical study. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1072–1077. Available at: <https://aclanthology.org/D16-1115/>.
- Dong, F., Zhang, Y., and Yang, J. (2017). Attention-based recurrent convolutional neural network for automatic essay scoring. In *Proceedings of the 21st Conference on Computational Natural Language Learning*, pages 153–162. DOI: 10.18653/v1/k17-1017.
- Fonseca, E. R., Medeiros, I., Kamikawachi, D., and Bokan, A. (2018). Automatically grading brazilian student essays. In *Proceedings of International Conference on Computational Processing of the Portuguese Language*, pages 170–179. DOI: 10.1007/978-3-319-99722-3_18.
- Fu, Y., Peng, H., Ou, L., Sabharwal, A., and Khot, T. (2023). Specializing smaller language models towards multi-step reasoning. In *International Conference on Machine Learning*, pages 10421–10430. PMLR. DOI: 10.48550/arXiv.2301.12726.
- Gu, Y., Dong, L., Wei, F., and Huang, M. (2023). Minillm: Knowledge distillation of large language models. *arXiv preprint arXiv:2306.08543*. DOI: 10.48550/arXiv.2306.08543.
- Higgins, D., Burstein, J., Marcu, D., and Gentile, C. (2004). Evaluating multiple aspects of coherence in student essays. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, pages 185–192, Boston, Massachusetts, USA. Association for Computational Linguistics. Available at: <https://aclanthology.org/N04-1024/>.
- Hu, E. J., yelong shen, Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., and Chen, W. (2022). Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*. Available at: <https://openreview.net/forum?id=nZeVKeeFYf9>.
- Hussein, M. A., Hassan, H. A., and Nassef, M. (2020). A trait-based deep learning automated essay scoring system with adaptive feedback. *International Journal of Advanced Computer Science and Applications*, 11(5):287–293. DOI: 10.14569/ijacsa.2020.0110538.
- Javaheripi, M. (2023). The surprising power of small language models. NeurIPS 2023 slides. Available at: https://nips.cc/media/neurips-2023/Slides/83968_5GxuY2z.pdf.
- Jin, C., He, B., Hui, K., and Sun, L. (2018). Tdnn: A two-stage deep neural network for prompt-independent automated essay scoring. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1088–1097. Association for Computational Linguistics. DOI: 10.18653/v1/P18-1100.
- Ke, Z. and Ng, V. (2019). Automated essay scoring: A survey of the state of the art. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intel-*

- ligence (IJCAI-19), pages 6300–6308. DOI: 10.24963/ijcai.2019/879.
- Klebanov, B. B. and Madnani, N. (2020). Automated evaluation of writing – 50 years and counting. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7796–7810. Available at: <https://aclanthology.org/2020.acl-main.697/>.
- Leal, S. E., Duran, M. S., Scarton, C. E., Hartmann, N. S., and Aluísio, S. M. (2024). Nilc-metrix: assessing the complexity of written and spoken language in brazilian portuguese. *Language Resources and Evaluation*, 58(1):73–110. DOI: 10.1007/s10579-023-09693-w.
- Lee, S., Cai, Y., Meng, D., Wang, Z., and Wu, Y. (2024). Unleashing large language models’ proficiency in zero-shot essay scoring. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 181–198. DOI: 10.18653/v1/2024.findings-emnlp.10.
- Li, S. and Ng, V. (2024). Conundrums in cross-prompt automated essay scoring: Making sense of the state of the art. In Ku, L.-W., Martins, A., and Srikumar, V., editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7661–7681, Bangkok, Thailand. Association for Computational Linguistics. DOI: 10.18653/v1/2024.acl-long.414.
- Liu, N. F., Lin, K., Hewitt, J., Paranjape, A., Bevilacqua, M., Petroni, F., and Liang, P. (2024). Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173. DOI: 10.1162/tacl.0.0638.
- Mansour, W. A., Albatarni, S., Eltanbouly, S., and Elsayed, T. (2024). Can large language models automatically score proficiency of written essays? In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 2777–2786. DOI: 10.48550/arXiv.2403.06149.
- Marinho, J., Anchiêta, R., and Moura, R. (2021). Essay-br: a brazilian corpus of essays. In *Anais do III Dataset Showcase Workshop*, pages 53–64. DOI: 10.5753/dsw.2021.17414.
- Marinho, J., Cordeiro, F., Anchiêta, R., and Moura, R. (2022). Automated essay scoring: An approach based on enem competencies. In *Anais do XIX Encontro Nacional de Inteligência Artificial e Computacional*, pages 49–60. DOI: 10.5753/eniac.2022.227202.
- Mazza Zago, R. and Agnoletti dos Santos Pedotti, L. (2024). Bertugues: A novel bert transformer model pre-trained for brazilian portuguese. *Semina: Ciências Exatas e Tecnológicas*, 45:e50630. DOI: 10.5433/1679-0375.2024.v45.50630.
- Mei, W. S. (2006). Creating a contrastive rhetorical stance: Investigating the strategy of problematization in students’ argumentation. *RELIC journal*, 37(3):329–353. DOI: 10.1177/0033688206071316.
- Mello, R. F., Oliveira, H., Wenceslau, M., Batista, H., Cordeiro, T., Bittencourt, I. I., and Isotani, S. (2024). Propor’24 competition on automatic essay scoring of portuguese narrative essays. In *Proceedings of the 16th International Conference on Computational Processing of Portuguese-Vol. 2*, pages 1–5. Available at: https://repositorio.usp.br/single.php?id=003200619&locale=en_US.
- OpenAI, :, Jaech, A., Kalai, A., Lerer, A., Richardson, A., El-Kishky, A., Low, A., Helyar, A., Madry, A., and et al. (2024a). Openai o1 system card. DOI: 10.48550/arxiv.2412.16720.
- OpenAI (2022). Introducing chatgpt. *ChatGPT*. Available at: <https://openai.com/index/chatgpt/>.
- OpenAI, Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., and et al. (2024b). Gpt-4 technical report. DOI: 10.48550/arxiv.2303.08774.
- Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al. (2022). Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744. DOI: 10.48550/arXiv.2203.02155.
- Page, E. B. (1966). The imminence of... grading essays by computer. *The Phi Delta Kappan*, pages 238–243. Book.
- Persing, I., Davis, A., and Ng, V. (2010). Modeling organization in student essays. In Li, H. and Márquez, L., editors, *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 229–239, Cambridge, MA. Association for Computational Linguistics. Available at: <https://aclanthology.org/D10-1023/>.
- Persing, I. and Ng, V. (2013). Modeling thesis clarity in student essays. In Schuetze, H., Fung, P., and Poesio, M., editors, *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 260–269, Sofia, Bulgaria. Association for Computational Linguistics. Available at: <https://aclanthology.org/P13-1026/>.
- Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., et al. (2018). Improving language understanding by generative pre-training. Available at: https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf.
- Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551. DOI: 10.48550/arxiv.1910.10683.
- Ramnarain-Seetohul, V., Bassoo, V., and Rosunally, Y. (2022). Similarity measures in automated essay scoring systems: A ten-year review. *Education and Information Technologies*, 27(4):5573–5604. DOI: 10.1007/s10639-021-10838-z.
- Raschka, S. (2023). Practical tips for finetuning llms using lora (low-rank adaptation). Available at: <https://magazine.sebastianraschka.com/p/practical-tips-for-finetuning-llms> Substack article, accessed 7 July 2025.
- Ribeiro, E., Mamede, N., and Baptista, J. (2024). Exploring the automated scoring of narrative essays in brazilian portuguese using transformer models. In *Proceedings*

- of the 16th International Conference on Computational Processing of Portuguese-Vol. 2, pages 14–17. Available at:<https://aclanthology.org/2024.propor-2.4/>.
- Ridley, R., He, L., Dai, X., Huang, S., and Chen, J. (2020). Prompt agnostic essay scorer: A domain generalization approach to cross-prompt automated essay scoring. Available at:<https://arxiv.org/abs/2008.01441>.
- Ridley, R., He, L., Dai, X.-y., Huang, S., and Chen, J. (2021). Automated cross-prompt scoring of essay traits. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(15):13745–13753. DOI: 10.1609/aaai.v35i15.17620.
- Rodriguez, P. U., Jafari, A., and Ormerod, C. M. (2019). Language models and automated essay scoring.
- Ruder, S. (2018). Nlp’s imagenet moment has arrived. Available at:<https://thegradient.pub/nlp-imagenet/>.
- Santos, R., Rodrigues, J., Gomes, L., Silva, J., Branco, A., Cardoso, H. L., Osório, T. F., and Leite, B. (2024). Fostering the ecosystem of open neural encoders for portuguese with albertina pt-* family. DOI: 10.48550/arXiv.2403.01897.
- Schick, T. and Schütze, H. (2021). It’s not just size that matters: Small language models are also few-shot learners. In Toutanova, K., Rumshisky, A., Zettlemoyer, L., Hakkani-Tur, D., Beltagy, I., Bethard, S., Cotterell, R., Chakraborty, T., and Zhou, Y., editors, *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2339–2352, Online. Association for Computational Linguistics. DOI: 10.18653/v1/2021.naacl-main.185.
- Schneer, D. (2014). Rethinking the argumentative essay. *Tesol Journal*, 5(4):619–653. DOI: 10.1002/tesj.123.
- Silveira, I. C., Barbosa, A., da Costa, D. S. L., and Mauá, D. D. (2025). Investigating universal adversarial attacks against transformers-based automatic essay scoring systems. In Paes, A. and Verri, F. A. N., editors, *Intelligent Systems*, pages 169–183, Cham. Springer Nature Switzerland. DOI: 10.1007/978-3-031-79032-4_12.
- Silveira, I. C., Barbosa, A., and Mauá, D. D. (2024). A new benchmark for automatic essay scoring in portuguese. In *Proceedings of the 16th International Conference on Computational Processing of Portuguese - Vol. 1*, pages 228–237. Available at:<https://aclanthology.org/2024.propor-1.23/>.
- Souza, F., Nogueira, R., and Lotufo, R. (2020). Bertimbau: pretrained bert models for brazilian portuguese. In *9th Brazilian Conference on Intelligent Systems, BRACIS, Rio Grande do Sul, Brazil, October 20-23 (to appear)*. DOI: 10.1007/978-3-030-61377-8_28.
- Taghipour, K. and Ng, H. T. (2016). A neural approach to automated essay scoring. In Su, J., Duh, K., and Carreras, X., editors, *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1882–1891, Austin, Texas. Association for Computational Linguistics. DOI: 10.18653/v1/D16-1193.
- Tay, Y., Phan, M., Tuan, L. A., and Hui, S. C. (2018). Skipflow: Incorporating neural coherence features for end-to-end automatic text scoring. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32. DOI: 10.1609/aaai.v32i1.12045.
- Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., Rodriguez, A., Joulin, A., Grave, E., and Lample, G. (2023). Llama: Open and efficient foundation language models. DOI: 10.48550/arXiv.2302.13971.
- Warner, B., Chaffin, A., Clavié, B., Weller, O., Hallström, O., Taghadouini, S., Gallagher, A., Biswas, R., Ladhak, F., Aarsen, T., Cooper, N., Adams, G., Howard, J., and Poli, I. (2024). Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. DOI: 10.18653/v1/2025.acl-long.127.
- Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q., and Zhou, D. (2023). Chain-of-thought prompting elicits reasoning in large language models. Available at:<https://arxiv.org/abs/2201.11903>.
- Williamson, D. M., Xi, X., and Breyer, F. J. (2012). A framework for evaluation and use of automated scoring. *Educational Measurement: Issues and Practice*, 31(1):2–13. DOI: 10.1111/j.1745-3992.2011.00223.x.
- Yang, K., Raković, M., Li, Y., Guan, Q., Gašević, D., and Chen, G. (2024). Unveiling the tapestry of automated essay scoring: A comprehensive investigation of accuracy, fairness, and generalizability. In *Proceedings of the aaai conference on artificial intelligence*, volume 38, pages 22466–22474. DOI: 10.1609/aaai.v38i20.30254.