Calibrating LLMs with Preference Optimization on Thought Trees for Generating Rationale in Science Question Scoring

Jiazheng Li[1], **Hainiu Xu**[1], **Zhaoyue Sun**[1,4]
Yuxiang Zhou[1], **David West**[2], **Cesare Aloisi**[2], **Yulan He**[1,3]
[1]Department of Informatics, King's College London, UK   [2]AQA, UK
[3]The Alan Turing Institute, UK [4]Department of Computer Science, University of Warwick, UK
{jiazheng.li, hainiu.xu, yuxiang.zhou, yulan.he}@kcl.ac.uk
{caloisi, dwest}@aqa.org.uk zhaoyue.sun@warwick.ac.uk

Abstract

Generating rationales that justify scoring decisions has been a promising way to facilitate explainability in automated scoring systems. However, existing methods do not match the accuracy of classifier-based methods. Plus, the generated rationales often contain hallucinated information. To address these issues, we propose a novel framework capable of generating more faithful rationales and, more importantly, matching performance with classifier-based black-box scoring systems. We first mimic the human assessment process by querying Large Language Models (LLMs) to generate a thought tree. We then summarise intermediate assessment decisions from each thought tree path for creating synthetic rationale data and rationale preference data. Finally, we utilise the generated synthetic data to calibrate LLMs through a two-step training process: supervised fine-tuning and preference optimization. Extensive experimental results demonstrate that our framework achieves a 38% assessment performance improvement in the QWK score compared to prior work while producing higher-quality rationales, as recognised by human evaluators and LLMs. Our work sheds light on the effectiveness of performing preference optimization using synthetic preference data obtained from thought tree paths.

1 Introduction

Student answer marking requires substantial human efforts. The human scoring process (e.g., Figure 1(a)) typically involves first understanding the content of the student response, then identifying valid key answer elements (e.g., "*student should answer what materials to test in their response*"), and finally assigning a score according to a marking rubric (e.g., "*2 points for accurately describe two additional pieces of information*") Ennis (1993); Scouller (1998). Numerous studies Alikaniotis et al. (2016); Dong et al. (2017) have aimed to automate such an assessment pipeline, thereby reducing labour and standardising assessment criteria. Traditional automated assessment methods generally address this problem through text classification Larkey
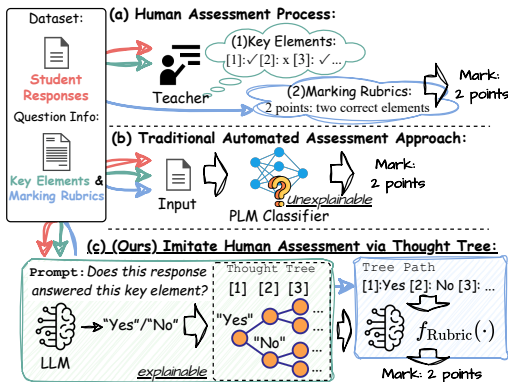

Figure 1: Illustration of (a) human assessment and (b) text classifier assessment approaches. We propose (c) imitating the human assessment procedure by prompting LLMs to generate rationales using a thought tree.
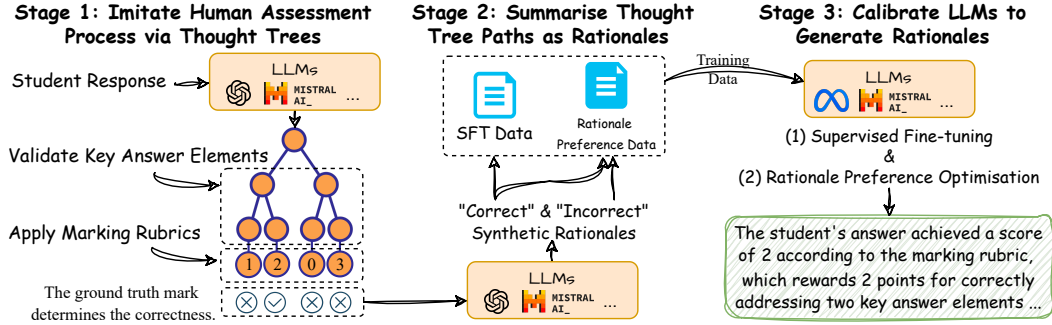
Figure 2: Overview of our three-stage Thought Tree Guided Rationale Generation Framework.

(1998) or regression Xie et al. (2022), with the goal of predicting the score for a given student response (Figure 1(b)).

Recent studies suggested that Large Language Models (LLMs) could be leveraged to explain student answer scoring. For example, Li et al. (2023a) utilised ChatGPT-generated rationales as noisy supervision signals to train a smaller model for simultaneous student answer scoring and rationale generation. However, their framework does not match the accuracy of classification baselines and suffers from hallucination problems, leading to unfaithful and inconsistent rationales. Moreover, it does not support the assessment of key elements in student responses according to the marking rubric.

To address the aforementioned issues, we propose a novel framework to imitate the human scoring process by dividing the scoring of student answers into two steps (Figure 1(c)). Initially, an LLM is tasked with comparing each key answer element against a student answer and providing a binary assessment decision ("*Yes*" or "*No*") indicating its presence or absence. These series of intermediate decisions, combined with prompts, ultimately form an assessment thought path Yao et al. (2023a,b). Subsequently, the assessment outcomes for each key element are aggregated using a marking rubric to derive the final score. This aggregation process can either be executed using explicit arithmetic calculations in Python code or directly by the LLM. Each thought tree paths can be faithfully summarised into synthetic response-level rationales, facilitating explainable automated student answer scoring.

Furthermore, the thought tree paths could contain trajectory that leads to correct or incorrect assessments. Such positive and negative rationale samples generated by the thought tree can be regarded as preference data, which allows us to conveniently construct a synthetic rationale preference dataset. Therefore, to calibrate LLMs for both accurate assessment of student answers and faithful generation of corresponding rationales, we align LLMs with our preference dataset using advanced RLHF methods Christiano et al. (2017); Ouyang et al. (2022). Specifically, our alignment pipeline contains the following two steps: (1) supervised fine-tuning (SFT) on synthetic *response-level rationale data*, followed by (2) preference optimization Rafailov et al. (2023) on synthetic *preference data*. To address data scarcity, we generate synthetic data nearly three times larger than the original dataset. Our experimental results demonstrate that our framework enhances the explainability of the assessment process by generating more faithful rationales while matching or exceeding the accuracy of traditional text classification approaches.

In summary, our contributions are: **(1)** We proposed a method for generating more faithful assessment rationales by imitating human assessment processes through thought trees; **(2)** We developed a technique for generating synthetic preference data by modelling preferences based on the correctness of thought tree paths; **(3)** We carried out extensive experiments, demonstrating that our framework achieved a 38% improvement in QWK compared to the state-of-the-art assessment rationale generation framework while generating more informative and higher-quality rationales. To the best of our knowledge, this is the first work to perform preference optimization using synthetic preference rationales that are generated via thought tree paths[1].

---

[1]Data and code are available at `https://github.com/lijiazheng99/thought_tree_assessment`.

2   Thought Tree Guided Rationale Generation Framework

We propose a *Thought Tree Guided Rationale Generation framework* to improve scoring performance and generate higher-quality assessment rationales. An overview of our proposed three-stage framework is provided in Figure 2. In this section, we will first introduce the problem setup and then explain the method used in each stage.

Problem Setup   Datasets for developing automated answer scoring systems typically contain information about questions, such as question prompts, key answer elements, and marking rubrics, along with a set of student responses and the corresponding marks. Formally, let $D = \{(x_i, y_i)\}_{i=1}^{N}$ represent N data points under the same question, where x_i denotes a student response and y_i represents the corresponding mark.

Let $K = \{k_j\}_{j=1}^{M}$ denote the set of key elements for the current question, where M is the total number of distinct key answer elements. For a points-based marking scheme, the marking rubric can be viewed as a question-dependent, piece-wise scoring function, $f_r(\cdot)$, defined as:

$$y_i = f_r(\mathbf{v}(x_i, K)), \tag{1}$$

where $\mathbf{v}(x_i, K) \in \mathbb{R}^M$ is a multi-hot vector indicating the coverage of key answer elements $k_j \in K$. Specifically, the jth index of $\mathbf{v}$ takes the form of $\mathbb{1}_{x_i}(k_j)$ that returns "*Yes*" if the key element k_j correctly answered in x_i and "*No*" otherwise.

The application of the marking rubric is purely deterministic; therefore, $\mathbb{1}_{k_j}(\cdot)$ becomes the most critical component influencing the quality of the human assessment process.

Stage 1: Imitate Human Assessment Process via Thought Trees

Prior work Li et al. (2023a) on explainable student answer scoring suggests that LLMs may struggle to perform the faithful assessment in one step, which may sometimes lead to hallucinations. To tackle this issue, we propose using LLMs to mimic human assessment decision-making by breaking the overall assessment task into intermediate assessment decisions that form deterministic thought trees.

As illustrated in Figure 3, our thought trees contain two levels of intermediate assessment. For the first level, we employ LLMs to imitate the key answer element validation in the human assessment process, denoted as $\mathbb{1}_{x_i}(k_j)$. We constrain the assessment decision for each key element k_j to a binary classification task. Each decision is sampled using Monte Carlo methods to mimic the variation of human decisions Browne et al. (2012). Once decisions are obtained for all key answer elements, the final leaf node aggregates the path's final mark $\hat{y}_i$ according to the marking rubric using explicit arithmetic calculations in Python code or directly by LLM.
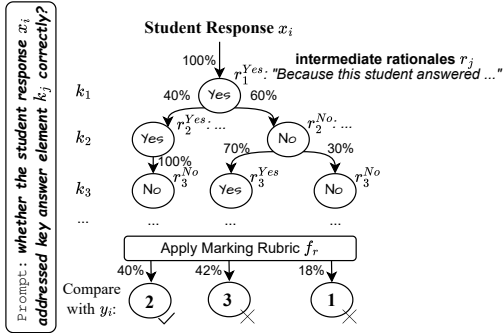

Figure 3: Illustration of the structure of a thought tree.

We formally define our thought tree as follows:
given an LLM LLM_θ (e.g., GPT-4), we prompt the LLM with the question information and student response x_i to generate an assessment thought tree. Initially, at the key element assessment level, the assessment decision z_j for a key answer element k_j is obtained by sampling LLM n times:

$$z_j^{(t)} = \text{LLM}_\theta(x_i, k_j), t = 1, 2, ..., n, \forall k_j \in K, \tag{2}$$

where we use an LLM parameterized by θ to mimic the human judgement process for identifying key answer elements from the student answer, denoted as $\mathbb{1}_{x_i}(k_j)$. Prompt messages adhere to a template of "*whether the student response x_i addressed key answer element k_j correctly*". Thus, akin to human judgement, the decision z_j can only be either "*Yes*" or "*No*". We can aggregate similar decisions to derive a sampled decision probability:

$$P(z_j^{\text{Yes}}) = |\{t : z_j^{(t)} = 1\}|/n,$$
$$P(z_j^{\text{No}}) = |\{t : z_j^{(t)} = 0\}|/n, \tag{3}$$

3

where $|\cdot|$ denotes the cardinality of the given set. These decision probabilities $P(z_j^{\text{Yes}})$ and $P(z_j^{\text{No}})$ represent the likelihood of each decision outcome, mimicking variations in real assessment decisions.

To ensure faithful intermediate decisions for aggregating response-level rationale at a later stage, we generate simpler, key element-wise rationales r_j that justify each unique decision outcome:

$$r_j = \texttt{LLM}_\theta(x_i, k_j, z_j). \tag{4}$$

Once every key answer element has been traversed, we can formulate and represent each distinct tree path as a multi-hot encoding using $\mathbf{v}(\cdot)$ from Eq. (1). Assuming we have d distinct paths, where $1 \leq d \leq 2^{(M-1)}$, each path can be represented as:

$$\text{path}_l = \hat{\mathbf{v}}(\mathcal{Z}), l = 1, 2, ..., d, \tag{5}$$

where $\hat{\mathbf{v}}$ is an estimate of $\mathbf{v}$, representing the coverage of key answer elements, and $\mathcal{Z}$ is a set of binary indicators derived from $P(z_j^{\text{Yes}})$ and $P(z_j^{\text{No}})$. The probability of each path $P(\text{path}_l)$ is the product of decision probabilities along the path:

$$P(\text{path}_l) = \prod_{j=1}^{M} P(z_j). \tag{6}$$

Subsequently, we apply the marking rubric f_r to obtain a score for each path:

$$\hat{y}_{\text{path}_l} = f_r(\text{path}_l). \tag{7}$$

Although the marking rubric is programmatic, it is still provided in text descriptions within the question format. In practical implementation, we utilize the LLM to dynamically transform the textual marking rubric into executable python code, which takes key element-level assessment decisions as a list as input and provides a score as output.

Finally, the predicted score for the current thought tree is determined by the path with the highest probability:

$$\hat{y}_{tree} = \hat{y}_{\text{path}_{l*}},$$
$$\text{where} \quad l^* = \arg\max_l P(\text{path}_l). \tag{8}$$

For detailed implementation, search algorithm, prompt setup and ablation studies on thought trees, please refer to Appendix B.1.

Stage 2: Summarise Thought Tree Paths as Rationales

In the second stage, we employ the thought trees generated in Stage 1 as reliable intermediate assessment decisions for generating synthetic student response-level rationales. To mitigate hallucination and maintain alignment with the thought tree, we leverage the extensive research on generating faithful summaries with LLMs and frame the synthetic rationale generation as a text summarization task Narayan et al. (2021); Aharoni et al. (2023); Lyu et al. (2024).

In this stage, we present the LLM with the question, student answer x_i, each key answer element k_j, key answer element-level assessment decision z_j, and rationale r_j, along with textual descriptive marking rubric and the predicted path score $\hat{y}_{\text{path}_l}$, in a prompt template, and ask the LLM summarize the response-level rationales. To ensure a diverse generation of free-form rationales, we require the LLM to output in JSON format, containing three columns of outputs:

> **Mark**: Output the assessed mark, which should be the same as $\hat{y}_{\text{path}_l}$.
> **Rationale**: Summarizes a response-level rationale using each key element-level assessment decision and rationales, and explicitly assign a mark applying the marking rubric.
> **Suggestion**: If the student didn't receive a full mark, we extract the answer improvement suggestion from the assessment rationale to provide useful improvement feedback.

Eventually, we construct the following three datasets using the above-generated resources:

Supervised Fine-tuning (SFT) Data   In this dataset, we form SFT data by choosing the path with the minimum absolute predicted score difference with the ground truth score: $l^* = \arg\min_l \left| \hat{y}_{\text{path}_l} - y_i \right|$. The **Rationale** and **Suggestion** columns for summarising the path l^* are used to construct multi-round assessment SFT data. The first round of the assessment is provided with the entire question information, and the student answer x_i to request an assessment rationale. The output is the **Rationale**. If the student didn't get a full score in the second round, we continue prompting the LLM to provide an improvement **Suggestion** based on the first round chat history.

Rationale Preference Data   As we want to generate faithful rationales without compromising assessment performance, we frame assessment correctness as human preferences, where a more accurate assessment rationale is preferred without disrupting the faithfulness and coherence of the rationale Korbak et al. (2023); Zhang et al. (2024a); Lu et al. (2024); Xie et al. (2024). We then choose the **Rationale** from correctly predicted $path_l$, where $\hat{y}_{path_l} = y_i$, as our preferred rationale, denoted as $\hat{y}_{r+}$. Other incorrectly predicted $path_l$s, where $\hat{y}_{path_l} \neq y_i$, as not preferred rationales, denoted as $\hat{y}_{r-}$. Therefore, for an input x, the preference for the rationale is denoted as $\hat{y}_{r+} \succ \hat{y}_{r-}|x$. The preferences are assumed to be generated by some latent reward model $r^*(y, x)$. Subsequently, we use the Bradley-Terry Bradley and Terry (1952) model to hypothesizes the rationale preference distribution p^* can be estimated via pairwise comparison:

$$p^*(\hat{y}_{r+} \succ \hat{y}_{r-} \mid x) = \frac{\exp(r^*(x, \hat{y}_{r+}))}{\exp(r^*(x, \hat{y}_{r+})) + \exp(r^*(x, \hat{y}_{r-}))}. \tag{9}$$

Rationale-to-Score Data   Different from Li et al. (2023a), rationales generated from our framework are in a free-text format without any template constraints. Thus, evaluating the assessment performance of rationales becomes a challenge. To automatically extract the mark from the rationale, we construct a dataset using **Rationale** as input and **Mark** as output, which is used to train a score extractor for assessing performance.

Questions sometimes involve parallel key elements, where different paths may result in the same score, and all could be considered correct. To ensure clarity and relevance, we present the question details, student responses, and all generated rationales to the LLM. Then, it is directed to resolve the conflict by selecting the most appropriate and accurate rationale that reflects the student's response, after which all other rationales are discarded.

We provide prompt templates, generated data statistics, and used instructions in Appendix B.2.

Stage 3: Calibrate LLMs to Generate Rationales

Finally, we utilize the synthetic rationale data generated from Stage 2 to calibrate LLMs for rationale generation via two steps of training:

Supervised Fine-tuning   We fine-tune an LLM on the synthetic SFT dataset with the standard cross-entropy loss to obtain a model π_{SFT}.

Preference Optimization on Rationales   We parameterize the preference in Eq. (9) into a reward model $r_\phi(x, y)$ and estimate the parameters via maximum likelihood. This problem can be framed as a binary classification task with a negative log-likelihood loss:

$$\mathcal{L}_R(r_\phi, D) = -\mathbb{E}_{(x, \hat{y}_{r+}, \hat{y}_{r-}) \sim D} \left[\log \left(\sigma(r_\phi(x, \hat{y}_{r+}) - r_\phi(x, \hat{y}_{r-})) \right) \right]. \tag{10}$$

Following Direct Preference Optimization (DPO) Rafailov et al. (2023), we can formulate a maximum likelihood objective for a parameterized policy π_θ. The policy objective becomes:

$$\mathcal{L}_{DPO}(\pi_\theta; \pi_{ref}) = -\mathbb{E}_{(x, \hat{y}_{r+}, \hat{y}_{r-}) \sim D} \left[\log \sigma \left(\beta \log \frac{\pi_\theta(\hat{y}_{r+} \mid x)}{\pi_{ref}(\hat{y}_{r+} \mid x)} - \beta \log \frac{\pi_\theta(\hat{y}_{r-} \mid x)}{\pi_{ref}(\hat{y}_{r-} \mid x)} \right) \right], \tag{11}$$

where the π_{ref} is the π_{SFT}. Empirically, training solely with $\mathcal{L}_{DPO}$ may cause model degeneration. In our implementation, we add a cross-entropy loss on the preferred rationale $\hat{y}_{r+}$ to mitigate this issue.

3   Experiments

3.1   Experimental Setup

Datasets   We use two data sources for our experiments: (1) The Hewlett Foundation Short Answer Scoring (ASAP) dataset, which focuses on short essay questions spanning topics in science and biology. (2) A private dataset comprising student responses with human-assigned scores for biology exam questions from a reputable examination service. We present our dataset statistics in Table 1.

Datasets	Train	Validation	Test	Score Range
ASAP 1	1,337	331	554	0-3
ASAP 2	1,018	252	426	0-3
ASAP 5	1,436	359	598	0-3
ASAP 6	1,437	359	599	0-3
Private 1	440	89	254	0-4
Private 2	358	72	196	0-3

Table 1: Dataset Statistics.

Baselines We compare our method with two baseline setups: Text Classifier: an assessment classifier built on DeBERTa Large He et al. (2023), designed to assess student answers and output scores Mayfield and Black (2020). AERA, an explainable student answer scoring approach Li et al. (2023a), which distills ChatGPT refined rationales into a LongT5 Guo et al. (2022) model.

Evaluation Metrics We evaluate the assessment performance using Accuracy (Acc), macro F1 score (F1), and Quadratic Weighted Kappa (QWK).

We provide detailed hyper-parameters setup, model version details, etc., in Appendix A.

Methods	Private 1 (Biology)			Private 2 (Biology)			ASAP 1 (Science)			ASAP 2 (Science)		
	Acc	F1	QWK	Acc	F1	QWK	Acc	F1	QWK	Acc	F1	QWK
Text Classifier (C)	0.6787	0.6784	0.8853	0.6224	0.6355	0.8385	0.7767	0.7805	0.8528	0.6798	0.6817	0.8187
AERA (R)	0.4055	0.3838	0.7139	0.2551	0.1020	0.0024	0.6326	0.6290	0.7506	0.4327	0.4235	0.5415
Our Framework												
Thought Tree (C)	0.5276	0.5078	0.7891	0.5459	0.5362	0.6710	0.6011	0.6145	0.7419	0.6746	0.6852	0.7856
Mixtral 8×7B SFT (R)	0.5906	0.5799	0.8326	0.5255	0.5213	0.7324	0.7310	0.7398	0.8511	0.5915	0.5837	0.6886
Mixtral 8×7B DPO (R)	0.6063	0.6016	**0.8440**	0.5867	0.5917	**0.7802**	0.7744	0.7808	**0.8749**	0.6526	0.6763	**0.7788**

Methods	ASAP 5 (Biology)			ASAP 6 (Biology)			Overall			QWK Statistics	
	Acc	F1	QWK	Acc	F1	QWK	Acc	F1	QWK	σ	Cf. w/ AERA
Text Classifier (C)	0.8625	0.6055	0.8187	0.8891	0.6118	0.8426	0.7515	0.6656	0.8428	0.0006	+41.21%
AERA (R)	0.8378	0.5329	0.7644	0.8937	0.6038	0.8081	0.5762	0.4458	0.5968	0.0933	-
Our Framework											
Thought Tree (C)	0.8043	0.5317	0.7834	0.7496	0.4429	0.5831	0.6505	0.5530	0.7257	0.0069	+21.59%
Mixtral 8×7B SFT (R)	0.8579	0.5948	0.8174	0.8564	0.5942	0.7735	0.6922	0.6023	0.7826	0.0040	+31.13%
Mixtral 8×7B DPO (R)	0.8779	0.6326	**0.8447**	0.8715	0.6011	**0.8245**	0.7282	0.6473	**0.8245**	0.0015	+38.15%

Table 2: Comparison of performance on (C)lassification and (R)ationale assessment approaches. The highest QWK has been highlighted in **bold** for the highest approaches to rationale generation , and an underline denotes the highest across all approaches. **QWK Statistics** provide insight into overall assessment performance: σ indicates the QWK score variation across all datasets, with lower values showing better generalization. Cf. w/ AERA compares overall performance with AERA, where higher values denote significant performance improvements.

3.2 Overall Performance

To examine the scoring performance of our proposed framework, Table 2 summarizes the scoring results of classification-based baselines, the state-of-the-art explainable scoring model, compared with results produced from our framework[2].

Text Classifier Vs. AERA: We replicated the AERA framework on our private datasets with Chat-GPT[3]. The results show that AERA is significantly affected by data scarcity (on Private 2). With only a few hundred data points, the model struggles to generalize. Moreover, a comparison on the ASAP datasets reveals a significant performance gap, particularly on ASAP 2. Although AERA offers better explainability, its performance shows a considerable trade-off, with an average 23% drop in accuracy and a 33% reduction in F1 score compared to the text classifier.

Thought Tree Vs. Baselines: Thought Trees[4] are used for generating response-level synthetic rationales. Simply deriving the scoring decisions from the Thought Tree's highest probability paths reduces the performance gap with text classifiers to 14%. However, as discussed in §2, there is a chance that paths with lower probabilities could also be correct, suggesting there is room for model calibration. Notably, Thought Trees already achieve better assessment results than AERA on Private 1 & 2 and ASAP 2 & 5 datasets.

Mixtral 8×7B SFT Vs. Baselines: By fine-tuning Mixtral 8×7B using our synthesized rationales summarized from Thought Trees, we observe a notable improvement over AERA by nearly 20% in QWK averaged over all datasets, closing the performance gap with text classifiers.

---

[2]All the scores from free-form rationales are extracted with a score extractor described in Appendix A.5.

[3]We didn't upgrade the backend for rationale generation to GPT-4 due to the incompatibility with the refinement module. An analysis of this issue is presented in Appendix A.6.

[4]Thought Tree employs GPT-4 as the backend. Results using open-source models can be found in Appendix B.1

Mixtral 8×7B DPO Vs. All: The DPO training step shows improved performance across all metrics compared to the SFT stage (nearly 5% improvement in QWK) and baselines. As shown in Figure 4, we compare the confusion matrix between the assessment results from the SFT and DPO steps. Following the DPO training step, all scores ranging from 0 to 3 were better calibrated than the SFT result, showing modelling accuracy as synthetic human preferences could effectively enhance scoring performance. The decline in the score of 4 is attributed to a

Figure 4: Normalized confusion matrix for Mixtral 8×7B SFT (left) and Mixtral 8×7B DPO (right).

shortage of data, as Private 1 is the only dataset with a maximum score of 4 with only 51 instances. Remarkably, our DPO step's scoring performance not only outperforms AERA but also matches or exceeds the Text Classifier on the ASAP 1 & 5 datasets, confirming the effectiveness of our framework in enhancing explainability without compromising scoring accuracy.

3.3 Rationale Quality

Since the dataset lacks ground-truth annotations for rationales, we perform an automated evaluation using LLMs and a human evaluation to examine the quality of generated rationales.

3.3.1 Automated Evaluation with GPT-4

To compare the informativeness and correctness of rationales generated by our framework with the AERA baseline, we conduct an automated evaluation using GPT-4. The preferred rationale for each pairwise comparison was determined based on which rationale more accurately assessed the student response. We randomly sampled 30 generated rationales from each dataset, using the same SFT and DPO results presented in Table 2. As demonstrated in Table 3, GPT-4 prefer rationales generated from the SFT model

Pairwise Comparison	Win Rate
SFT vs. AERA	99.17%
DPO vs. AERA	100%
DPO vs. SFT	100%

Table 3: Rationale preferences evaluated by GPT-4.

over AERA because they were more informative and directly quoted the student responses. The SFT rationales that were not preferred often incorrectly assessed student responses. A similar preference trend was observed in the comparison between the rationales generated by AERA and DPO. However, the 100% preference rate in the comparison between SFT and DPO does not necessarily imply that DPO is superior. GPT-4 noted that in some cases, both rationales were identical in content, structure, and explanation, stating, "*Neither is more accurate or helpful than the other, as they provide the same level of detail and critique.*"

3.3.2 Human Evaluation

To evaluate the correctness of assessment decisions based on rationales in more detail, we conducted a human evaluation focusing on the *correctness of matching key answer elements* and the *faithfulness of applying marking rubrics*. We randomly sampled 30 cases from all datasets and provided evaluation instructions to our annotators[5].

As shown in Table 4, our evaluation reveals that rationales generated by our framework more accurately match key answer elements and apply the marking rubric more faithfully. This verifies the effectiveness of the thought tree rationale generation approach, which produces higher-quality rationales. In contrast, some AERA

Evaluated Aspects	DPO	AERA
Key Element Correctness	**83.33%**	36.67%
Marking Rubric Faithfulness	**96.67%**	66.67%
Accuracy on Sampled Set	**83.33%**	50.00%

Table 4: Human Evaluation on Rationale Quality.

cases showed that the model failed to generate meaningful content, instead repeating template phrases, suggesting difficulties in generalization with their approach.

Interestingly, we found no false positive cases in our framework, meaning the rationales were always meaningful when giving a correct score. However, three such cases were identified in the rationales

---

[5]Detail setup is presented in Appendix C.4.

generated by AERA. In the sampled DPO cases, incorrect assessments were all attributable to erroneous matching of key answer elements, often caused by overanalysis by the LLM based on context. For example, the LLM incorrectly inferred that the response correctly addressed "*global warming*" based on the mention of "*higher temperature of water*". This highlights the need for more explicitly defined scopes for the key answer elements. Among the sampled DPO cases, only one was found to apply the marking rubric incorrectly. Therefore, the majority of errors stem from mismatching key elements.

3.4 Case Studies for Rationale Quality

Student Response (from ASAP 2): The pour into the take the vinegar in each containers. who is the problem (Hypothesis) How you can do this experiment A table who explain the experiment use more types of masses like glass, paper, etc.

Ground-truth mark: 0

AERA (A): 1 point; This response describes one additional piece of information that would be needed to accurately replicate the experiment: 'the pour into the water in each container.'

Mixtral 8×7B DPO (B): The student's answer scored 0 points as per the marking rubric, which requires the response to describe additional pieces of information needed to accurately replicate the experiment. The student's answer failed to specify any necessary details such as the amount of vinegar used, the type of vinegar, the materials tested, the size/surface area of materials, the rinsing duration, the drying time, the drying method, or the size/type of container. The student's response included vague and unrelated statements such as 'The pour into the take the vinegar in each containers. who is the problem (Hypothesis) How you can do this experiment A table who explain the experiment use more types of masses like glass, paper, etc.' which do not address the specifics required for replication.

GPT-4 Evaluation: Comparison: Rationale B is more accurate and helpful because it clearly explains why the student's response failed to meet the requirements of the marking rubric by not providing specific and relevant information needed to replicate the experiment, whereas Rationale A incorrectly credits the student for providing useful information. More accurate and helpful: B

Mixtral 8×7B SFT: The student's answer scored 0 points as per the marking rubric, which requires the response to describe additional pieces of information needed to accurately replicate the experiment. The student's answer failed to specify any necessary details such as the amount of vinegar used, the type of vinegar, the materials tested, the size/surface area of materials, the rinsing duration, the drying time, the drying method, or the size/type of container. The student's answer included vague and unrelated statements such as 'The pour into the take the vinegar in each containers. who is the problem (Hypothesis) How you can do this experiment A table who explain the experiment use more types of masses like glass, paper, etc.' which do not address the specifics required for replication. *(Exactly the same as DPO output)*

Student Response (from ASAP 5): mRNA will change into tRNA and will leave amino acids. Then the amino acids will make a ribosome. mRNA will send a message to tRNA. Finally ribosomes will connect and make a bond   **Ground-truth Mark**: 0

AERA (A): 1 point; The student answer matches one key element, '...mRNA is sent to the ribosomes...'. However, the other statements are incorrect or incomplete.

Mixtral 8×7B SFT (B): The student's answer scored 0 points according to the marking rubric which awards 3 points for addressing four key elements, 2 points for three key elements, 1 point for one or two key elements, and 0 points for other responses. The student's answer contains several fundamental errors and misunderstandings about the protein synthesis process. Key inaccuracies include stating 'mRNA will change into tRNA' and 'the amino acids will make a ribosome,' which demonstrate a confusion between the roles and identities of mRNA, tRNA, and ribosomes. The student also fails to describe critical steps such as mRNA exiting the nucleus, mRNA traveling to the ribosome, the reading of mRNA codons by rRNA, and the specific interactions and roles of tRNA in carrying and adding amino acids. None of the key answer elements were correctly addressed as required by the marking rubric.

GPT-4 Evaluation: Comparison: Rationale B is more detailed and accurate, providing specific feedback on the student's misunderstandings and errors, which is essential for learning, whereas Rationale A is overly generous and lacks detail on the specific errors made. More accurate and helpful: B

Mixtral 8×7B DPO: The student's answer scored 0 points according to the marking rubric, which awards 0 points if none of the key elements are correctly addressed. The student's statements such as 'mRNA will change into tRNA' and 'the amino acids will make a ribosome' demonstrate a fundamental misunderstanding of mRNA, tRNA, and ribosome roles and functions in protein synthesis. None of the required key answer elements, including mRNA exiting the nucleus, mRNA traveling to the ribosome, reading of codons, and interactions between tRNA and mRNA, were correctly described.

Table 5: Case studies on rationale quality. Critical statements from the rationales are highlighted in red.

As demonstrated in Table 5, we provide several case studies to compare rationales generated by the AERA, Mixtral 8×7B SFT, and DPO models.

The first example is from the ASAP 2 dataset, where the correct mark is 0. The AERA model incorrectly assessed the student's answer, quoting *"the pour into the water in each container"*, which was not a valid key answer element. In contrast, the DPO output correctly assigned a score of 0 and highlighted the vagueness of the student's statements, directly quoting the exact student's answer. This direct quoting enhances the faithfulness of the generated rationale. The usefulness of the DPO's rationale is further recognized by GPT-4, noting that rationale B is more informative. Interestingly, the DPO-generated rationale remains consistent with the SFT stage output; possibly, it already aligns with the model's preferred distribution.

The second case involves a student response from ASAP 5, where the correct mark is 0. Again, the AERA model incorrectly inferred the key answer element "mRNA is sent to the ribosomes" from the student answer. In contrast, the SFT model correctly identified the student's fundamental errors and misunderstandings about the process, providing useful and specific feedback, which GPT-4 also

recognized. The Mixtral 8×7B DPO output provides almost the same information while being more concise, indicating that the DPO stage effectively removes redundant information while reinforcing assessment accuracy. Further extensive experiments, case studies and error analyses are provided in Appendix C.

4 Related Work

Automated Student Answer Scoring Existing automated student answer scoring (ASAS) systems typically frame the problem as a text classification task Larkey (1998); Taghipour and Ng (2016). Most ASAS research focuses on essay scoring, evaluating essays based on language quality Yannakoudakis et al. (2011); Raheja and Alikaniotis (2020); Velentzas et al. (2024). Various approaches have aimed to enhance the trustworthiness of these systems by explaining the decision-making processes through feature analysis Dong and Zhang (2016); Vanga et al. (2023); Li et al. (2023b) or visualizing weights and self-attention mechanisms Alikaniotis et al. (2016); Yang et al. (2020). However, due to limited resources, the scoring and explanation of science questions have not been extensively studied. Li et al. (2023a) proposed a rationale generation framework for science question scoring, but their approach suffers from hallucination issues and an apparent trade-off in assessment performance. Unlike previous works, our framework managed to generate more accurate assessment rationales that are more faithful to the assessment material.

Intermediate Rationale for LLM Problem Solving Recent developments in LLMs have demonstrated advances in reasoning capabilities Wei et al. (2022a); Kaplan et al. (2020); Brown et al. (2020). However, complex reasoning in a single step remains challenging for LLMs Zhang et al. (2024b). Researchers proposed CoT Wei et al. (2022b), which aimed at breaking down complex reasoning into intermediate steps. Building on this idea, various prompting methods have been proposed to improve the success rate in solving complex tasks, such as Self-Consistency Wang et al. (2023), which utilizes the notion of bagging, and Tree-of-Thought (ToT) Yao et al. (2023a); Zhang et al. (2024b), which increases the breadth of search on top of CoT. In our framework, we propose a thought tree to imitate the human assessment process via breaking down assessment task into intermediate assessment decisions. Our thought tree differs from the ToT in two significant ways: (1) Our task requires precise planning. Instead of prompting LLMs to generate plans, prompts in our intermediate steps are determined by each key element. Consequently, our tree has a fixed maximum depth of $M + 1$ and width of $2^{(M-1)}$ for M key answer elements. (2) The goal of the ToT is to find the most promising states that solve the problem more efficiently, whereas our goal is to sample all available paths, regardless of correctness, and generate more data for training.

5 Limitation

This study has several limitations: **Complexity of Tree Paths**: Tree query presented in the first stage of our framework may take huge computational resources if the set of key answer elements is large and the backend LLM is incapable of doing assessment (e.g. the thought tree could have $2^{(M-1)}$ distinct paths in the worst case). **Generation of Data Limited by LLM Content Filter**: Many language models have been aligned to prevent harmful use. Filtering harmful content may influence the data production rate in our framework. We noticed a few cases in which student responses were not harmful, and answers about the "death of plants" or "extinction of animals" eventually were refused assessment by LLMs. This limitation is more dataset-specific and only happens to questions that contain ambiguous terms and need to rely on the context. **Noisy Synthetic Rationales**: Although we have seen promising improvement in the assessment performance and rationale quality, the synthetic rationales generated from the second stage of our framework are still noisy. However, tackling this issue would require extensive human verification.

6 Conclusion

In this work, we introduced a novel Thought Tree Guided Rationale Generation Framework on providing more accurate scoring decisions justified by more faithful rationales. We employed a thought tree to imitate human assessment decision-making to enable more faithful intermediate assessment. Each tree path was then summarized into synthetic rationales. Extensive experiments,

LLM evaluation, and human evaluation results showed that calibrating LLMs with our framework-generated synthetic response-level rationales and synthetic rationale preference data can effectively provide more informative rationales that align with student responses while achieving higher scoring performance. Our work highlights the potential for preference optimization using synthetic preference data obtained from thought tree paths.

Acknowledgements

References

Roee Aharoni, Shashi Narayan, Joshua Maynez, Jonathan Herzig, Elizabeth Clark, and Mirella Lapata. 2023. Multilingual summarization with factual consistency evaluation. In *Findings of the Association for Computational Linguistics: ACL 2023*.

AI@Meta. 2024. Llama 3 model card.

Dimitrios Alikaniotis, Helen Yannakoudakis, and Marek Rei. 2016. Automatic text scoring using neural networks. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Ralph Allan Bradley and Milton E. Terry. 1952. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, et al. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*.

Cameron B. Browne, Edward Powley, Daniel Whitehouse, Simon M. Lucas, Peter I. Cowling, Philipp Rohlfshagen, Stephen Tavener, Diego Perez, Spyridon Samothrakis, and Simon Colton. 2012. A survey of monte carlo tree search methods. *IEEE Transactions on Computational Intelligence and AI in Games*.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. In *Advances in Neural Information Processing Systems*.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms. In *Advances in Neural Information Processing Systems*.

Fei Dong and Yue Zhang. 2016. Automatic features for essay scoring – an empirical study. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*.

Fei Dong, Yue Zhang, and Jie Yang. 2017. Attention-based recurrent convolutional neural network for automatic essay scoring. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*.

Robert H. Ennis. 1993. Critical thinking assessment. *Theory Into Practice*.

Mandy Guo, Joshua Ainslie, David Uthus, Santiago Ontanon, Jianmo Ni, Yun-Hsuan Sung, and Yinfei Yang. 2022. LongT5: Efficient text-to-text transformer for long sequences. In *Findings of the Association for Computational Linguistics: NAACL 2022*.

Ben Hamner, Jaison Morgan, Mark Shermis Lynnvandev, and Tom Vander Ark. 2012. The hewlett foundation: Automated essay scoring.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2023. DeBERTav3: Improving deBERTa using ELECTRA-style pre-training with gradient-disentangled embedding sharing. In *The Eleventh International Conference on Learning Representations*.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. 2024. Mixtral of experts. *Preprint*, arXiv:2401.04088.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *Preprint*, arXiv:2001.08361.

Tomasz Korbak, Kejian Shi, Angelica Chen, Rasika Vinayak Bhalerao, Christopher Buckley, Jason Phang, Samuel R. Bowman, and Ethan Perez. 2023. Pretraining language models with human preferences. In *Proceedings of the 40th International Conference on Machine Learning*.

Leah S. Larkey. 1998. Automatic essay grading using text categorization techniques. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98.

Jiazheng Li, Lin Gui, Yuxiang Zhou, David West, Cesare Aloisi, and Yulan He. 2023a. Distilling ChatGPT for explainable automated student answer assessment. In *Findings of the Association for Computational Linguistics: EMNLP 2023*.

Jiazheng Li, Zhaoyue Sun, Bin Liang, Lin Gui, and Yulan He. 2023b. CUE: An uncertainty interpretation framework for text classifiers built on pre-trained language models. In *The 39th Conference on Uncertainty in Artificial Intelligence*.

Jiazheng Li, Runcong Zhao, Yongxin Yang, Yulan He, and Lin Gui. 2023c. Overprompt: Enhancing chatGPT through efficient in-context learning. In *R0-FoMo:Robustness of Few-shot and Zero-shot Learning in Large Foundation Models*.

Junru Lu, Jiazheng Li, Siyu An, Meng Zhao, Yulan He, Di Yin, and Xing Sun. 2024. Eliminating biased length reliance of direct preference optimization via down-sampled kl divergence. *Preprint*, arXiv:2406.10957.

Qing Lyu, Marianna Apidianaki, and Chris Callison-Burch. 2024. Towards faithful model explanation in NLP: A survey. *Computational Linguistics*.

Elijah Mayfield and Alan W Black. 2020. Should you fine-tune BERT for automated essay scoring? In *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications*.

Shashi Narayan, Yao Zhao, Joshua Maynez, Gonçalo Simões, Vitaly Nikolaev, and Ryan McDonald. 2021. Planning with learned entity prompts for abstractive summarization. *Transactions of the Association for Computational Linguistics*.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Gray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Vipul Raheja and Dimitris Alikaniotis. 2020. Adversarial Grammatical Error Correction. In *Findings of the Association for Computational Linguistics: EMNLP 2020*.

Karen Scouller. 1998. The influence of assessment method on students' learning approaches: Multiple choice question examination versus assignment essay. *Higher education*.

Kaveh Taghipour and Hwee Tou Ng. 2016. A neural approach to automated essay scoring. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, Pouya Tafti, et al. 2024. Gemma: Open models based on gemini research and technology. *Preprint*, arXiv:2403.08295.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *Preprint*, arXiv:2307.09288.

Roopchand Reddy Vanga, C. Sindhu, M. S. Bharath, T. Charandeep Reddy, and Meghana Kanneganti. 2023. Autograder: A feature-based quantitative essay grading system using bert. In *ICT Infrastructure and Computing*.

Georgios Velentzas, Andrew Caines, Rita Borgo, Erin Pacquetet, Clive Hamilton, Taylor Arnold, Diane Nicholls, Paula Buttery, Thomas Gaillat, Nicolas Ballier, and Helen Yannakoudakis. 2024. Logging keystrokes in writing by English learners. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V Le. 2022a. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le, and Denny Zhou. 2022b. Chain of thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*.

Jiayi Xie, Kaiwei Cai, Li Kong, Junsheng Zhou, and Weiguang Qu. 2022. Automated essay scoring via pairwise contrastive regression. In *Proceedings of the 29th International Conference on Computational Linguistics*.

Yuxi Xie, Anirudh Goyal, Wenyue Zheng, Min-Yen Kan, Timothy P Lillicrap, Kenji Kawaguchi, and Michael Shieh. 2024. Monte carlo tree search boosts reasoning via iterative preference learning. *arXiv preprint arXiv:2405.00451*.

Ruosong Yang, Jiannong Cao, Zhiyuan Wen, Youzheng Wu, and Xiaodong He. 2020. Enhancing automated essay scoring performance via fine-tuning pre-trained language models with combination of regression and ranking. In *Findings of the Association for Computational Linguistics: EMNLP 2020*.

Helen Yannakoudakis, Ted Briscoe, and Ben Medlock. 2011. A new dataset and method for automatically grading ESOL texts. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik R Narasimhan. 2023a. Tree of thoughts: Deliberate problem solving with large language models. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. 2023b. React: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations*.

Xuan Zhang, Chao Du, Tianyu Pang, Qian Liu, Wei Gao, and Min Lin. 2024a. Chain of preference optimization: Improving chain-of-thought reasoning in llms. *Preprint*, arXiv:2406.09136.

Yizhou Zhang, Lun Du, Defu Cao, Qiang Fu, and Yan Liu. 2024b. Prompting with divide-and-conquer program makes large language models discerning to hallucination and deception. *Preprint*, arXiv:2402.05359.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, and Yongqiang Ma. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. *arXiv preprint arXiv:2403.13372*.

A Further Experimental Setup

A.1 Further Dataset Statistics

We utilize two datasets, The Hewlett Foundation: Short Answer Scoring (ASAP-SAS) dataset Hamner et al. (2012) and our private dataset. Detailed statistics of the dataset are presented in Table 1. Following the setup from Li et al. (2023a), for the ASAP dataset, we focus on non-objective science and biology questions. Please note that due to GPT-4's filtering of harmful content, one instance from the original ASAP 5 dataset has been removed from the training set.

A.2 Hyper-parameters and Model Versions

We report all the parameters and configurations we used in Table A1.

Stage 1			
Model Versions:			
Gpt-4 API as tree backend		`gpt-4-1106-preview`	
Mixtral 8×7B API as tree backend		`open-mixtral-8x7b`	
Mixtral 8×7B as tree backend		`mistralai/Mixtral-8x7B-v0.1`	
Generation Configs:			
`temperature`	0.7		
`candidates for decision`	10	`candidates for rationale`	1
`max_tokens for decision`	4	`max_tokens for rationale`	120
Stage 2			
Model Versions:			
Gpt-4 API as generation backend		`gpt-4-turbo-2024-04-09`	
Generation Configs:			
`temperature`	1.0	`candidates`	1
Stage 3			
Base Model:			
LLaMA 3 8B		`meta-llama/Meta-Llama-3-8B`	
Mixtral 8×7B		`mistralai/Mixtral-8x7B-v0.1`	
SFT:		**DPO**:	
`finetuning_type`	lora	`finetuning_type`	lora
`lora_target`	all	`lora_target`	all
`num_train_epochs`	4.0	`num_train_epochs`	3.0
`batch_size`	8	`batch_size`	1
`gradient_accumulation`	8	`gradient_accumulation`	8
`lr_scheduler_type`	cosine	`lr_scheduler_type`	cosine
`learning_rate`	5e-5	`learning_rate`	1e-5
`quantization_bit`	4	`quantization_bit`	4
`fp16`	true	`fp16`	true
`upcast_layernorm`	true	`upcast_layernorm`	true
`warmup_steps`	0.1	`warmup_steps`	0.1
Generation Configs:			
`do_samples`	False	`max_new_tokens`	256
Baselines and Score Extractor			
Text Classifier:		**Score Extractor**:	
`num_train_epochs`	15.0	`num_train_epochs`	20.0
`batch_size`	16	`batch_size`	32
`learning_rate`	2e-5	`learning_rate`	1e-5
`warmup_steps`	100	`warmup_steps`	100
`adam_epsilon`	1e-8	`adam_epsilon`	1e-8
`early_stopping_patience`	3	`early_stopping_patience`	3

Table A1: Hyper-parameters settings.

A.3 Model Implementation Details

For our implementation, we utilized the HuggingFace Transformers package Wolf et al. (2020) for accessing pre-built language model architectures. In the third stage of our framework, the

training and inference pipeline was set up using LLaMA-factory Zheng et al. (2024), along with the implementation of DPO Rafailov et al. (2023) and QLoRA Dettmers et al. (2023). Due to limited computational resources, all of the training involved in the third stage uses 4-bit QLoRA.

For text classifiers baseline and the Score Extractor, we employed the `microsoft/deberta-v3-large` model He et al. (2023). We uses LongT5 model Guo et al. (2022) for the AERA framework implementation. The `meta-llama/Meta-Llama-3-8B` AI@Meta (2024) and `mistralai/Mixtral-8x7B-v0.1` Jiang et al. (2024) were selected as the base models during the third stage of our framework.

Additionally, for accessing the GPT-4 OpenAI et al. (2024) and Mixtral APIs, we referred to their respective official websites.

We obey the licenses of all involved models. All of our models are trained with a single NVIDIA A100 80G GPU.

A.4 Quadratic Weighted Kappa

Quadratic Weighted Kappa, a widely used metric in evaluating the agreement between two raters in student response assessment, is defined as:

$$\kappa = 1 - \frac{\sum_{i=1}^{k} \sum_{j=1}^{k} w_{ij} O_{ij}}{\sum_{i=1}^{k} \sum_{j=1}^{k} w_{ij} E_{ij}} \tag{12}$$

where k is the score set, w is the weighted matrix, calculates as: $w_{i,j} = \frac{(i-j)^2}{(k-1)^2}$. O is a $k \times k$ histogram matrix and E being the $k \times k$ expected value matrix.

A.5 Score Extractor

Since our framework generates free-form rationales, the assessed mark could be embedded anywhere within the rationale context. We therefore train a Score Extractor utilizing a PLM-based text classifier on the rationale-to-score dataset constructed in Stage 2. The classifier is trained on 11,641 training data instances and validated with 2,319 instances, achieving a final accuracy of 1.0, a macro f1 score of 1.0, and a QWK score of 1.0. This indicates that our score extractor could accurately extract assessed marks from free-text rationales.

A.6 Model Choice for AERA

We tried to align the model use and upgrade the rationale generation backend used in the AERA Li et al. (2023a) to GPT-4. However, we find that GPT-4 may sometimes fail on the label refinement step in the AERA framework. When utilising gpt-4-turbo-preview for rationale generation, an occasional occurrence arises where the model disregards the provided ground truth label hint and generates its own justification. E.g., "*0 points; 2.5 points; The student's answer begins to touch on several key elements but does not fully or clearly articulate them according to the criteria.*" where "0 points;" are provided within the prompt, and the model supposed to continue to generate a rationale that justifies this decision.

We randomly sampled 89 instances and compared the rationale generated by GPT-4 and ChatGPT. We found 5 cases from GPT-4 that suffer from such an issue. This observation verifies our concern about the difficulty of LLMs solving complex reasoning tasks in one step. Interestingly, no single case from the ChatGPT output encountered deviations from the provided hint.

B Framework Implementation Details

In this section, we provide further details on the implementation and ablation studies we carried out for our framework.

B.1 Stage 1 further details

B.1.1 Thought Tree Path Search Algorithm

As demonstrated in Algorithm 1, we adopt depth-first search in our implementation for both querying and traversing thought trees. We use Anytree[6] to implement the tree node.

Algorithm 1 Tree Path Search Algorithm

```
 1: procedure DFS(parent, key_elements)
 2:     Input: Parent node and key elements list
 3:     Output: Traversal of the thought tree
 4:
 5:     if length(key_elements)== 0 then
 6:         return
 7:     end if
 8:
 9:     decisions = Query(key_elements[0])
10:                                        ▷ Obtain intermediate decisions
11:     key_elements.pop(0)
12:                                        ▷ Remove queried key answer elements
13:
14:     for decision ∈ decisions do
15:         new_node = decision
16:         new_node.parent = parent
17:         DFS(new_node, key_elements)
18:     end for
19:                                        ▷ Assign new decision and traverse
20:
21: end procedure
```

B.1.2 Thought Tree Prompt Design

As shown in Figure B.1.2, we provide a universal prompt template we used to query LLMs.

> **Prompt Templates for Thought Trees**
>
> **Query decision for k_j:**
> [Key Answer Element]: Does this student answer specify [*fill in the current key answer element content*]? Please select Yes for matching this key answer element and No for non-matching this key answer element.
> {{demonstation}}
> [Student Answer]: "{{student_answer}}"
> [Decision]:
> **Query rationale r_j for k_j:**
> Please generate a rationale that justifies your above decision. Quote ** part of this student answer ** (e.g. just a few words) that answers this key element with "...". Do not include any other additional information that are not relevant to this decision.

B.1.3 Marking Rubric to Python Code

As shown in Figure B.1.3, we present an example Python code transformed by the GPT-4 model for ASAP 1's marking rubrics. Our framework is implemented to automatically use a Python interpreter to execute this code and dynamically used to summarize marks from a list of key answer element decisions.

---

[6]https://pypi.org/project/anytree/
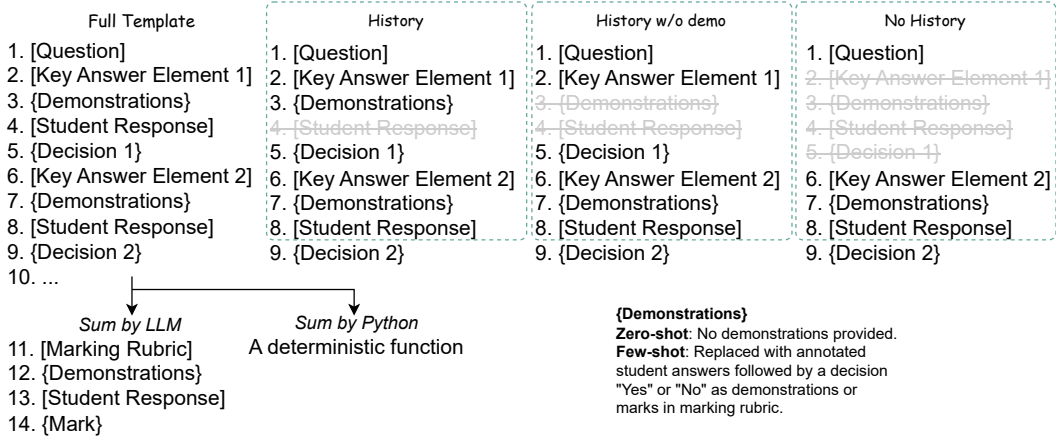
Figure A1: Explanation of our prompt structure using querying decision 2 as an example.

Marking Rubric to Python Code

Original Marking Rubric
3 points: The response describes three additional pieces of information that would be needed to accurately replicate the experiment;
2 points: The response describes two additional pieces of information that would be needed to accurately replicate the experiment;
1 point: The response describes one additional piece of information that would be needed to accurately replicate the experiment;
0 point: The response describes little or no accurate or relevant information from the acid rain investigation.
GPT-4 Transformed Python Code:

```python
def sum_score(node_names):
    num_of_correct="; ".join(node_names).count("Yes")
    if num_of_correct >= 3:
        return 3
    elif num_of_correct == 2:
        return 2
    elif num_of_correct == 1:
        return 1
    else:
        return 0
```

B.1.4 Ablation Study on Prompt Structure

We invested significant effort to identify the optimal prompt structure for querying thought trees. This section outlines our rationale behind the final template structure used in stage 1 of our framework.

TL;DR: Few-shot prompt template in "History w/o demos" mode proved most effective.

Prompt Template Setup: We sampled several hundred instances from the Private 1 dataset's training set for this trial experiment, as it only contains four key answer elements, the fewest among all our datasets. As depicted in Figure A1, our prompt template included two demonstration setups: zero-shot and few-shot. In the zero-shot mode, no demonstrations are provided. In the few-shot mode, each score level includes an annotated student answer, with the annotated example key answer element decisions labelled as "Yes" (if the answer answered the key element) or "No" (if not). At the marking rubric assessment stage, we designed two marking rubric application modes in our framework: "Sum by Python", which uses a deterministic Python function, and "Sum by LLM", which intuitively provides the text descriptive marking rubrics to the LLM and query LLMs to summarize a score. However, we didn't use "Sum by LLM" in our final thought tree for reasons detailed in the subsequent experimental analysis.

17

No	Prompt Mode	w/wo history	Marking Rubric Format	Acc	F1	QWK
1	Zero-shot	no history	Python	0.4173	0.3760	0.6459
2	Zero-shot	no history	LLM	0.4252	0.3876	0.6481
3	Zero-shot	history w/o demo	Python	0.5039	0.4782	0.7355
4	Zero-shot	history w/o demo	LLM	0.5157	0.4910	0.7491
5	Few-shot	no history	Python	0.4803	0.4474	0.7370
6	Few-shot	no history	LLM	0.4843	0.4391	0.7432
7	Few-shot	history	Python	0.4528	0.4157	0.7278
8	Few-shot	history	LLM	0.4685	0.4444	0.7544
9	Few-shot	history w/o demo	Python	**0.5354**	**0.5116**	**0.7791**
10	Few-shot	history w/o demo	LLM	0.5354	0.5098	0.7690
11	Few-shot (Duo)	no history	Python	0.4567	0.4139	0.7103
12	Few-shot (Duo)	no history	LLM	0.4961	0.4608	0.7287
13	Few-shot (Duo)	history	Python	0.4685	0.4380	0.7397
14	Few-shot (Duo)	history	LLM	0.4724	0.4400	0.7528
15	Few-shot (Duo)	history w/o demo	Python	0.5197	0.4977	0.7721
16	Few-shot (Duo)	history w/o demo	LLM	0.5236	0.4991	0.7537
17	Few-shot (Shuffle)	history	Python	0.4528	0.4186	0.7182
18	Few-shot (Shuffle)	history	LLM	0.5039	0.4866	0.7551
19	Few-shot (Shuffle)	history w/o demo	Python	0.5079	0.4693	0.7533
20	Few-shot (Shuffle)	history w/o demo	LLM	0.5433	0.5219	0.7804
21	Few-shot (Duo) (Shuffle)	history	Python	0.4921	0.4678	0.7062
22	Few-shot (Duo) (Shuffle)	history	LLM	0.4921	0.4615	0.7173
23	Few-shot (Duo) (Shuffle)	history w/o demo	Python	0.5394	0.5272	0.7614
24	Few-shot (Duo) (Shuffle)	history w/o demo	LLM	0.5512	0.5328	0.7879

Table A2: Thought Tree Assessment Performance with Different Prompt Structure on GPT-4.

As shown in Figure A1, we designed three history modes: "History", which includes full history from previous decisions (e.g., when querying decision 2, we retain key answer element 1 and its demonstrations along with decision 1 query result); "History w/o demos", which omits demonstrations used in the last session but retains only the decisions (e.g., when querying decision 2, we retain key answer element 1 and decision 1 query result, but removed the demonstration from the history); and "No History", treating each key answer element as a separate, parallel query (e.g., the prompt only contains the current key element's demonstration).

Demonstration Examples Setup: "Few-shot" contains 5 examples picked from scores ranging from 0-4. Each instance has been annotated to whether contains key element k_j or not. "Few-shot (Duo)" contains 2 examples. We picked a full-score example in contrast to a zero-score example. "Shuffle": For "history" and "history w/o demo" mode, each current key answer element decision could be influenced by the previous key answer element provided in the history. Therefore, we carried out an ablation study by shuffling the order of input key answer elements to examine the influence of input order.

Experimental Results: We present a comprehensive experiment analysis in Table A2. Our analysis reveals that the scoring for different key elements is hierarchical, influenced by the sequential processing of elements and the demonstration provided. The use of few-shot examples, especially in the "History w/o demos" mode, significantly enhanced the accuracy of key element decisions, as evidenced by the improved performance in setups 3 Vs. 9 and 1 Vs. 5. History including demonstrations (7 Vs. 9) did not prove beneficial, this may due to the reason that previous key elements' demonstrations may influence the effect of demonstrations for the current key answer element and confuses the LLM to make confident decisions.

Since the key element-level decision only requires "Yes" or "No", we present (Duo) demonstrations, which means we only provide a pair of demonstrations that contain ("Yes") and not contain ("No") the key answer element. By comparing five demonstration examples with two demonstration examples (5 Vs. 11, 9 Vs. 15), we find that providing more demonstration examples can result in better results.

We also investigated the influence of order on the key answer elements, denoted as (Shuffle) in the table. We find that different order settings of key answer elements could result in fluctuation in assessment performance. However, when human assessors assess the student responses, they most likely follow a top-to-bottom order. Therefore, we keep the original key answer element orders for each question.

(a) Sum by Python	(b) Sum by LLM
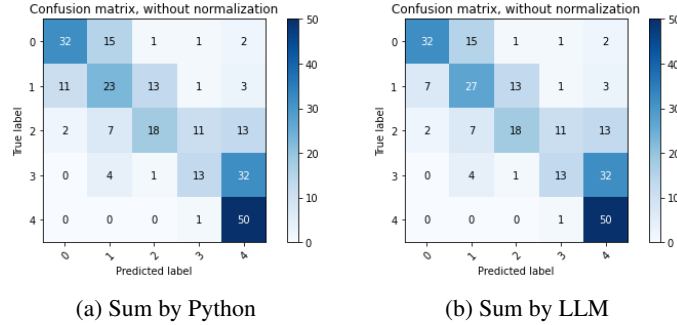
Figure A2: Comparison of confusion matrix for "Sum by Python" (left) and "Sum by LLM" (right).

In most cases, the "Sum by LLM" method improved over the "Sum by Python" approach. We illustrated this comparison in Figure A2, where the confusion matrices for experiments 9 and 10 highlight the differences between the two methods. While "Sum by LLM" appears to enhance results, it does not adhere as strictly to the original assessment tree as the "Sum by Python" method, which is purely deterministic. Interestingly, "Sum by LLM' seems to ignore previous key answer element-level assessment decisions and provide a final mark that potentially gives higher scoring performance. This unpredictable improvement led us to exclude "Sum by LLM" as the source of generating synthetic rationales. However, this approach holds potential for future investigation, particularly if combined with strategies like performing self-reflection to correct erroneous intermediate decisions thought paths.

B.1.5   Backend Model Choices

As illustrated in Table A3, we comprehensively evaluated various model options to query thought trees and imitate the real human assessment processes. We discovered that only GPT-4 and Mixtral 8×7B could accurately capture and reason based on our prompt design.

We find that the LLaMA 2 series models Touvron et al. (2023) struggled with adhering to the prompt requirements. Although the model `google/gemma-7b-it` Team et al. (2024) was able to follow the prompt, it often incorporated excessive context. In contrast, google/gemma-2b-it correctly adhered to the prompt and made appropriate judgments. However, before each tree query, we implemented a preliminary check of basic computational accuracy—querying the model with "what is $6 + 7 =$"—revealed that `google/gemma-2b-it` was the only model to incorrectly return 47 as the answer, indicating potential issues with its reliability for our purposes.

B.1.6   Assessment Performance for Thought Tree with Different Backends

As highlighted in §B.1.5, only GPT-4 and Mixtral 8×7B could effectively handle our designed prompts. In this section, we compare the assessment thought tree's performance of these two models. We consider paths with the highest predicted probability as the predicted mark.

Table A4 outlines the performance metrics on test sets for both backends. Evidently, GPT-4 significantly outperforms Mixtral on the Private 1 and ASAP datasets 1, 2, and 5. We also calculated the correct ratio:

$$\text{correct ratio} = \frac{\text{Num of tree containing correct mark}}{\text{Total num of trees}}$$

This correct ratio calculates how many tree paths contain the ground-truth labels, although it might not be the highest probability path. As shown in Table A4, counting other lower predictive probability paths could potentially increase the accuracy near 95%, with an average improvement from 14% to 27%. This insight is crucial as it implies a higher success rate for synthetic data generation in Stage 2.

B.1.7   Cost Analysis

Table A5 details the cost analysis for obtaining thought trees with GPT-4 as backend. The reported cost includes the total query cost for train, validation and test sets. The cost for querying each question highly correlates to the length of question information and the size of the key answer elements set. Our framework will take more cost in the query if the question contains more key answer elements.

Prompt:
[Question]: Microorganisms in the soil recycle carbon from the leaves so that the carbon is used for new plant growth. Explain how. [4 marks]
[Key Answer Element]: Does this answer address the decay or decomposition of leaves? Additionally, consider if the answer includes aspects of how leaves are broken down. Please select *Yes* for matching this key answer element and *No* for non-matching this key answer element.
[Student Answer]: "[*An example student answer*]" **[Decision]**: Yes
[Student Answer]: "[*An example student answer*]" **[Decision]**: Yes
[Student Answer]: "[*An example student answer*]" **[Decision]**: Yes
[Student Answer]: "[*An example student answer*]" **[Decision]**: No
[Student Answer]: "[*An example student answer*]" **[Decision]**: No
[Student Answer]: "[*A student answer needed to be assessed*]" **[Decision]**:

mistralai/Mixtral-8x7B-Instruct-v0.1:
No, this answer discusses the movement of carbon through the food chain, not the decomposition of leaves in the soil.

meta-llama/Llama-2-7b-chat-hf:
Great, it looks like you're on the right track! Here's how you can improve your answer:
1. You're partially correct when you say that microorganisms in the soil recycle carbon from leaves. However, you need to provide more detail on how this process works. For example, you could explain that microorganisms such as bacteria and fungi break down the leaves into simpler compounds, releasing carbon dioxide into the atmosphere through respiration. This carbon dioxide is then absorbed by plants during photosynthesis, which produces glucose that contributes to new plant growth.

meta-llama/Llama-2-13b-chat-hf:
Here are the decisions for each student answer:
1. [Student Answer]: "the example student answer"
Decision: Yes
This answer addresses the decay or decomposition of leaves and includes aspects of how leaves are broken down. It correctly explains that microorganisms decompose leaves, releasing carbon dioxide into the atmosphere, which is then taken in by plants during photosynthesis to produce glucose for growth.

meta-llama/Llama-2-70b-chat-hf
Yes, all of the student answers address the decay or decomposition of leaves and include aspects of how leaves are broken down.
The first student answer explains that microorganisms decompose dead leaves, releasing carbon dioxide into the atmosphere, which is then taken in by plants during photosynthesis and used for growth.

google/gemma-7b-it
Analysis of Student Answers
The question: Microorganisms in the soil recycle carbon from the leaves so that the carbon is used for new plant growth.
Key Answer Element: Does this answer address the decay or decomposition of leaves? Additionally, consider if the answer includes aspects of how leaves are broken down.
Student Answer 1:
* **Decision:** Yes

google/gemma-2b-it
No

Table A3: Comparison output with different open source LLMs.

Backend	GPT-4				Mixtral 8×7B		
Datasets	Acc	F1	**QWK**	Correct Ratio	Acc	F1	**QWK**
Private 1	0.5276	0.5078	**0.7891**	0.6614	0.3780	0.2996	0.6526
Private 2	0.5459	0.5362	**0.6710**	0.7806	0.4643	0.4454	0.6122
ASAP 1	0.6011	0.6145	**0.7419**	0.8610	0.4386	0.4351	0.6285
ASAP 2	0.6746	0.6852	**0.7856**	0.9437	0.2911	0.2793	0.3469
ASAP 5	0.8043	0.5317	**0.7834**	0.9482	0.7102	0.3999	0.6750
ASAP 6	0.7496	0.4429	0.5831	0.9299	0.8446	0.5060	**0.6716**

Table A4: Thought Tree Assessment Performance with Different Backend Options.

Datasets	Cost (in USD)
Private 1	215.96
Private 2	181.25
ASAP 1	2,116.28
ASAP 2	705.21
ASAP 5	913.23
ASAP 6	931.28

Table A5: API Cost Analysis for Querying.

B.2 Stage 2 Further Details

B.2.1 Prompt Template for Rationale Summarization

We used the prompt template provided in Figure B.2.1 to summarize response-level rationales and improvement suggestions in json format Li et al. (2023c), to obtain a free-form textual output.

Prompt Template for Response-Level Rationale Summarization for GPT-4

Here is a student answer to the following question:
"{question}"
[Student Answer]: "{student_answer}"
This question follows a points mark scheme, and the breakdown assessment by each Key Answer Element for this student's answer is as follows:
{idx}. {key_element} - {decision}: {rationale}
...

According to the: "{rubric}", the answer should get a score of {tree_predicted_score}.

Please summarize the above rationales and be FAITHFUL to the given assessment decisions for this student's answer briefly and precisely. Give the summarization in JSON format:
```JSON
{
    "mark": "...", # numeric
    "rationale": "...", # including mark awarded, which marking rubric applied, and detailed key elements level rationale
    "suggestion": "..." # any answer improvement suggestion
}```
- The "mark" should be the score of the student's answer.
- The "rationale" should be concise, include the assessed score and rubric applied, more importantly, justify the **marking decision-making processes** by **summarizing** the key element-level rationales, you must **quote the exact part** from the student's answer.
- If the student didn't get a full mark, you can also provide some suggestions for improvement; otherwise, leave it blank.

---

### B.2.2 Synthetic Generated Data Statistics

This section presents statistics for the synthetic SFT and preference data generated across three settings: all datasets, all datasets excluding ASAP 2, and all datasets excluding our Private data. Table A6 details the distribution of this data across training and validation subsets. A primary motivation for this study is to address data scarcity issues inherent in the student answer scoring datasets in general. As illustrated in Figure A3, our method generated more than three times the amount of synthetic data compared to the original dataset, significantly enriching the available data resources for training.

### B.2.3 Prompt Instructions for the SFT Data

In this section, we include the instruction prompt for rationale generation and generate improvement suggestions, which are randomly assigned to each instance.

| Datasets | Train | Validation |
|---|---|---|
| Rationale to score | 11,641 | 2,319 |
| SFT all datasets | 11,450 | 2,771 |
| SFT w/o ASAP 2 | 9,607 | 2,314 |
| SFT w/o Private | 10,126 | 2,520 |
| DPO all datasets | 4,960 | 705 |
| DPO w/o ASAP 2 | 4,635 | 624 |
| DPO w/o Private | 4,799 | 676 |

Table A6: Statistics for Generated Synthetic Rationale Data.



Figure A3: Data statistics of generated synthetic data.

## Rationale Instructions

- "Evaluate the [Student Answer] based on the provided [Key Answer Elements] and [Marking Rubric]. Summarize the assessment and justify the score awarded: "
- "Analyze the [Student Answer] thoroughly. Generate a detailed rationale that explains the strengths and weaknesses of the response:"
- "Conduct a comprehensive evaluation of the [Student Answer] using the provided [Key Answer Elements] and any relevant background information. Provide a detailed rationale for the assessment, including specific references to the [Marking Rubric]:"
- "Critically assess the [Student Answer] in light of the [Key Answer Elements]. Offer a detailed explanation for the score assigned, referencing specific criteria from the [Marking Rubric]:"
- "Review the [Student Answer] for accuracy and completeness compared to the [Key Answer Elements]. Provide a comprehensive rationale that includes the strengths, weaknesses, and areas for improvement:"
- "Examine the [Student Answer] thoroughly, comparing it against the [Key Answer Elements] and using the guidelines provided in the [Marking Rubric]. Clearly articulate the reasoning behind each point awarded or deducted:"
- "Systematically evaluate the [Student Answer] by aligning it with the [Key Answer Elements] and assessing according to the [Marking Rubric]. Document your findings and the logic of your evaluation comprehensively:"

## Suggestion Instructions

- "Analyze the [Student Answer] and identify areas for improvement. Suggest specific changes that could enhance the clarity, accuracy, and depth of the response:"
- "Please provide targeted suggestions to help the student refine their answer for better alignment with the expected criteria:"
- "Please offer some actionable feedback that the student can use to improve their response:"
- "Could you propose concrete improvements that could elevate the response's overall quality and effectiveness:"
- "Please recommend enhancements that can strengthen the logic, persuasiveness, and completeness of the answer:"

# C  Further Experiments Results

## C.1  Comprehensive Analysis on Framework Effectiveness

As presented in Table A8, we provide a comprehensive comparison of the effectiveness of our framework on two base models: Mixtral 8×7B and LLaMA 3 8B, with three dataset settings:

**Trained on All Data**: Models are trained on all the SFT & DPO datasets.
**Trained on All Data w/o ASAP2**: Models are trained on SFT & DPO w/o ASAP 2 datasets.
**Trained on All Data w/o Private data**: Models are rained on SFT & DPO w/o Private datasets.

Table A6 presents detailed data statistics for each dataset combination.

**Statistical Significance Test**    To determine the statistical significance of differences in the QWK scores among the AERA, Mixtral 8×7B SFT, and Mixtral 8×7B DPO methods, we employed the Kruskal-Wallis test, followed by pairwise Mann-Whitney U tests with Bonferroni correction for multiple comparisons.

**Kruskal-Wallis Test**: The test showed a statistically significant difference among the groups ($H = 9.50$, $p = 0.0087$), indicating at least one method's performance is significantly different from the others.
**Pairwise Mann-Whitney U Test**:
  AERA vs. Mixtral SFT: Not significant ($U = 9.0$, $p = 0.053$, adjusted $p$ not significant).
  AERA vs. Mixtral DPO: Significant ($U = 2.0$, $p = 0.0049$, adjusted $p$ significant).
The results suggest that the Mixtral 8×7B DPO method significantly outperforms AERA, but shows no significant difference when compared to the Mixtral SFT method.

**Experiments with LLaMA 3 8B**    As demonstrated in Table A8, we extended our experiments to LLaMA 3 8B model. When compared with AERA's results from Table 2, the SFT model trained on LLaMA exhibited an $11.17\%$ performance improvement. Consistent with observations in §3.2, our results on LLaMA further confirm that preference modelling (DPO training step) can enhance assessment performance by $2\%$ compared to SFT. Despite a minor decrease in the QWK score on the ASAP 5 dataset, both accuracy and F1 scores showed improvements. These results demonstrate that our framework works not only with a mixture of expert models, reinforcing its robustness and adaptability.

**Experiments on Mixtral 8×7B**    As demonstrated in Table A8, our experiments for Mixtral 8×7B on three dataset settings show superiority over the LLaMA 3 8B as a base model choice. This is probably due to Mixtral 8×7B has shown advanced reasoning capability compared with LLaMA 3 8B Jiang et al. (2024). However, compared with the size of the parameters and the test performance, LLaMA 3 8B is still a competitive model choice considering the training cost.

**Effectiveness on DPO**    As demonstrated in Table A8, our six experiment combinations (two base models × three datasets settings) show the effectiveness of rationale preference optimization on the generated synthetic preference data. Interestingly, experiments show that models tested on unseen datasets may bring negative effects. However, this effect is possibly influenced by the similarity of the questions. For example, the Private 1 question shares a similar format to the ASAP 5 & 6, which probably caused the performance actually to improve on the "w/o Private data" setting.

**Transferability on Unseen Datasets**    Given that ASAP 2 is recognized as the most challenging dataset in Li et al. (2023a), we conducted an ablation study by excluding ASAP 2 from the training set to evaluate the model's transferability on unseen data. Table A8 shows performance metrics for both SFT and DPO stages trained with Mixtral. Despite removing ASAP 2 from training, all other datasets improved after the DPO training step, albeit with a slight drop in QWK for ASAP 2. This is quite intuitive, as ASAP 2 hasn't been trained during any phases, so the preference for ASAP 2 hasn't been properly modelled in the dataset. By comparing with Table 2, we can still see a performance improvement than the AERA's result in both SFT ($7.63\%$) and DPO ($10.79\%$) stages. This demonstrates the assessment ability could go beyond its training data.

**Student response**: Based on the students data I can conclude that plastic type B has the largest amount stretched. Two ways that the student could have improved the experimental decison is by allowing the weights to hang longer to see if the plastic would stretch more. Another way he could have improved his experiment is by doing a third trials to really be sure that his results are correct.

**SFT w/ all data**: The student's answer scored 2 points according to the marking rubric. The student correctly drew a valid conclusion, stating that 'plastic type B has the largest amount stretched,' which aligns with the data provided. Additionally, the student proposed one valid experimental design improvement by suggesting to 'do a third trials to really be sure that his results are correct,' addressing the need for additional trials to confirm results. However, the student did not address other potential improvements such as ensuring uniform sample sizes, thickness, or the consistency in the weight added, which are crucial for experimental validity.

**SFT w/o ASAP 2**: The student's answer scored 2 points according to the marking rubric, as it 'draws a valid conclusion supported by the student's data and describes one way the student could have improved the experimental design andor the validity of the results.' The student correctly identified that 'plastic type B has the largest amount stretched,' which is supported by the data. However, the student failed to address other key elements such as the comparative stretchability of other plastics, variations in thickness, and the exact number of weights used. The student did suggest improvements like 'allowing the weights to hang longer' and 'doing a third trials,' which aligns with the rubric's requirement for describing experimental design improvements.

Table A7: Comparison of Generated Rationale between SFT trained with All Data and All Data w/o ASAP 2.

Interestingly, as detailed in the case study presented in Table A7, no significant quality differences are observed between assessment rationales generated by SFT Mixtral models trained with or without ASAP 2. Both rationales accurately capture the student's valid conclusion and proposed experimental improvements according to the marking rubric. They all agreed that another improvement suggestion was not valid, directly quoting the original phrases from the student's response.

Moreover, we can continue to observe this trend by observing the experiments trained with "w/o Private data". Despite the failure of LLaMA 3 8B on the Private 2 dataset, all the other results still maintain a comparable QWK score, as the results were trained with "all data".

**Ablation Study on Decoding Method**  We performed an ablation study on the decoding strategy as shown in the last two rows from Table A8. We set the sample as True to use beam search on both LLaMA 3 and Mixtral $8\times7$B SFT results. Interestingly, overall assessment performance dropped by 4.75% and 3.62% for LLaMA and Mixtral models, respectively. Therefore, we chose to use greedy decoding throughout our experiments.



(a) Before Training

(b) After Training

Figure A4: Comparison between Thought Tree Before (left) and After (right) Training with Our Framework.

## C.2   Case Study: Correction on Thought Tree

As illustrated in Figure A4, although our model was not trained on thought trees, we observe that the highest probability path (5% shown on the right-hand side) accurately assesses the score. The path "*Yes; Yes; No; No*" indicates that the student's response covered the first two key answer elements but missed the last two. The rationale generated for this student's response also reflects this decision by validating the first two key answer elements. However, due to finetuning, we can no longer generate a thought tree for some responses. Nonetheless, for those that can still be generated, we have identified patterns that suggest our trained model could potentially calibrate the LLM to generate thought trees that are coherent with its rationale.

| Test Data | | Private 1 | | | Private 2 | | | ASAP 1 | | | ASAP 2 | | | ASAP 5 | | | ASAP 6 | | | Overall | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Eval Metrics | | Acc | F1 | QWK | Acc | F1 | QWK | Acc | F1 | QWK | Acc | F1 | QWK | Acc | F1 | QWK | Acc | F1 | QWK | Acc | F1 | QWK |
| Model | Stage | | | | | | | | | | | | | | | | | | | | | |
| *Trained on All Data* | | | | | | | | | | | | | | | | | | | | | | |
| LLaMA 3 8B | SFT | 0.5394 | 0.5180 | 0.8118 | 0.5306 | 0.5313 | 0.7548 | 0.7058 | 0.7142 | 0.8360 | 0.6643 | 0.6769 | 0.7925 | 0.8595 | 0.5921 | 0.7936 | 0.8381 | 0.6079 | 0.7628 | 0.6896 | 0.6067 | 0.7919 |
| | DPO | 0.5315 | 0.5226 | 0.8439 | 0.5459 | 0.5570 | 0.7624 | 0.7527 | 0.7616 | 0.8579 | 0.6408 | 0.6525 | **0.8038** | 0.8612 | 0.6364 | 0.7735 | 0.8715 | 0.5809 | 0.8042 | 0.7006 | 0.6185 | 0.8076 |
| Mixtral 8×7B | SFT | 0.5906 | 0.5799 | 0.8326 | 0.5255 | 0.5213 | 0.7324 | 0.7310 | 0.7398 | 0.8511 | 0.5915 | 0.5837 | 0.6886 | 0.8579 | 0.5948 | 0.8174 | 0.8564 | 0.5942 | 0.7735 | 0.6922 | 0.6023 | 0.7826 |
| | DPO | 0.6063 | 0.6016 | **0.8440** | 0.5867 | 0.5917 | **0.7802** | 0.7744 | 0.7808 | **0.8749** | 0.6526 | 0.6763 | 0.7788 | 0.8779 | 0.6326 | **0.8447** | 0.8715 | 0.6011 | **0.8245** | 0.7282 | 0.6473 | **0.8245** |
| *Trained on All Data w/o ASAP2* | | | | | | | | | | | | | | | | | | | | | | |
| LLaMA 3 8B | SFT | 0.5197 | 0.5079 | 0.7876 | 0.5255 | 0.5203 | 0.7581 | 0.6733 | 0.6834 | 0.8096 | 0.3615 | 0.3397 | 0.4396 | 0.8512 | 0.6235 | 0.7389 | 0.8214 | 0.5647 | 0.7394 | 0.6254 | 0.5399 | 0.7122 |
| | DPO | 0.4803 | 0.4459 | 0.8256 | 0.5561 | 0.5685 | **0.7658** | 0.7310 | 0.7417 | 0.8536 | 0.4131 | 0.3854 | 0.5117 | 0.8495 | 0.5562 | 0.7413 | 0.8798 | 0.5881 | 0.8087 | 0.6517 | 0.5476 | 0.7511 |
| Mixtral 8×7B | SFT | 0.5866 | 0.5756 | 0.8311 | 0.5102 | 0.4991 | 0.7156 | 0.7040 | 0.7143 | 0.8386 | 0.5164 | 0.5217 | **0.6744** | 0.8545 | 0.6036 | 0.8055 | 0.8414 | 0.5623 | 0.7648 | 0.6689 | 0.5794 | 0.7717 |
| | DPO | 0.6378 | 0.6388 | **0.8535** | 0.5357 | 0.5446 | 0.7587 | 0.7726 | 0.7802 | **0.8727** | 0.5141 | 0.5250 | 0.6589 | 0.8696 | 0.6054 | **0.8295** | 0.8648 | 0.5703 | **0.8129** | 0.6991 | 0.6107 | **0.7977** |
| *Trained on All Data w/o Private data* | | | | | | | | | | | | | | | | | | | | | | |
| LLaMA 3 8B | SFT | 0.3701 | 0.3617 | 0.6839 | 0.3367 | 0.2894 | 0.3982 | 0.6426 | 0.6589 | 0.8058 | 0.6338 | 0.6495 | 0.7577 | 0.8495 | 0.6090 | 0.7764 | 0.8331 | 0.5776 | 0.7434 | 0.6110 | 0.5243 | 0.6942 |
| | DPO | 0.4331 | 0.3944 | **0.7457** | 0.2908 | 0.2096 | 0.3503 | 0.7112 | 0.7210 | 0.8343 | 0.6526 | 0.6659 | **0.7953** | 0.8545 | 0.5617 | 0.7601 | 0.8648 | 0.5950 | 0.8071 | 0.6345 | 0.5246 | 0.7155 |
| Mixtral 8×7B | SFT | 0.3937 | 0.3832 | 0.6594 | 0.5051 | 0.5020 | **0.6819** | 0.7130 | 0.7237 | 0.8439 | 0.5822 | 0.5755 | 0.6920 | 0.8428 | 0.5460 | 0.7942 | 0.8497 | 0.5834 | 0.7841 | 0.6478 | 0.5523 | **0.7426** |
| | DPO | 0.4606 | 0.4603 | 0.7060 | 0.4592 | 0.4395 | 0.6120 | 0.7563 | 0.7654 | **0.8643** | 0.5258 | 0.5562 | 0.6329 | 0.8679 | 0.6151 | **0.8272** | 0.8748 | 0.6220 | **0.8090** | 0.6574 | 0.5764 | 0.7419 |
| *Ablation Study on Decoding Method (Trained on All Data)* | | | | | | | | | | | | | | | | | | | | | | |
| LLaMA 3 8B | SFT | 0.5433 | 0.5247 | 0.7944 | 0.5357 | 0.5342 | 0.7421 | 0.6733 | 0.6870 | 0.8247 | 0.6526 | 0.6647 | 0.7757 | 0.8462 | 0.6043 | 0.7741 | 0.8397 | 0.6106 | 0.7434 | 0.6522 | 0.5505 | 0.7543 |
| Mixtral 8×7B | SFT | 0.5827 | 0.5590 | 0.8206 | 0.5000 | 0.4916 | 0.7151 | 0.6426 | 0.6559 | 0.7999 | 0.4953 | 0.4797 | 0.6303 | 0.8411 | 0.5157 | 0.7929 | 0.8514 | 0.6011 | 0.7667 | 0.7076 | 0.5631 | 0.7475 |

Table A8: Overall Comparison of Assessment Performance under Three Different Combinations of Training Data Settings. The highest QWK under each training dataset setting is highlighted in **Bold**, and the highest QWK under the same base model has been Underlined. Colored cells indicates the testing results on unseen datasets.

### C.3 LLM Automated Evaluation

#### C.3.1 Prompt Template

Following the DPO Rafailov et al. (2023), we designed the following prompt template to evaluate the rationale generated from our framework and calculate the win rate.

---

**Prompt Template for Response Level Rationale Summarization for GPT-4**

For the following assessment rationale for the student response, which one is more accurate and helpful?
[Question]: {question}
[Key Answer Elements]: {key_answer_elements}
[Marking Rubric]: {marking_rubric}
[Student Response]: "{student_response}"
[Teacher Assessed Mark]: {mark}

Rationale A:
"{baseline}"
Rationale B:
"{response}"
FIRST provide a one-sentence comparison of the two rationales and explain which you feel is more accurate and helpful. SECOND, on a new line, state only "A" or "B" to indicate which response is more accurate and helpful. Your response should use the format:
Comparison: <one-sentence comparison and explanation>
More accurate and helpful: <"A" or "B">

---

#### C.3.2 Sampled Win Rate Rationales

As shown in Table A9, we sampled some cases of GPT-4's win rate assessments, complete with rationales and decisions.

### C.4 Human Evaluation

#### C.4.1 Evaluation Setup

We conducted our human evaluation using the Prolific[7] annotation platform. To ensure efficient evaluation, we divided the sampled data into batches, each containing sample outputs from the same question. This allowed for a consistent evaluation standard across all rationale assessments. All annotators were screened to ensure they had higher education from an English-speaking country. We compensated our annotators at a rate of $12.80 per hour.

#### C.4.2 Evaluation on Rationale Quality

**Rationale Evaluation Instruction**    Our annotation instruction is presented to our annotator is presented in Figure A5.

#### C.4.3 Evaluation on Stage 2 Summarization Faithfulness

In Stage 2 of our framework, we model the response-level synthetic rationale generation as a summarization task. In this section, we present a human evaluation of the faithfulness of the generated rationales. We evaluated the faithfulness across four aspects with binary annotation from a total of 30 sampled cases.

The first aspect is "Scoring alignment with tree path," which assesses whether the assigned mark matches the tree path's score. We found that in 93.33% of the sampled cases, the mark column aligns with the tree path, indicating reliable summarization. Similarly, since we used the mark and rationale columns to train a score extractor, we evaluated whether the mark corresponds to the assessment decision presented in the rationale. We found that all marks are correctly aligned with the rationale's result.

---

[7] https://app.prolific.com/

**Student Response** (from ASAP 6): Ground-truth mark: 0

mRNA will change into tRNA and will leave amino acids. Then the amino acids will make a ribosome. mRNA will send a message to tRNA. Finally ribosomes will connect and make a bond

**A: AERA Generated Rationale**:

1 point; The student answer matches one key element, '...mRNA is sent to the ribosomes...'. However, the other statements are incorrect or incomplete.

**B: Mixtral 8×7B SFT Generated Rationale**:

The student's answer scored 0 points according to the marking rubric which awards 3 points for addressing four key elements, 2 points for three key elements, 1 point for one or two key elements, and 0 points for other responses. The student's answer contains several fundamental errors and misunderstandings about the protein synthesis process. Key inaccuracies include stating 'mRNA will change into tRNA' and 'the amino acids will make a ribosome,' which demonstrate a confusion between the roles and identities of mRNA, tRNA, and ribosomes. The student also fails to describe critical steps such as mRNA exiting the nucleus, mRNA traveling to the ribosome, the reading of mRNA codons by rRNA, and the specific interactions and roles of tRNA in carrying and adding amino acids. None of the key answer elements were correctly addressed as required by the marking rubric.

**GPT-4 Evaluation**:

Comparison: Rationale B is more detailed and accurate, providing specific feedback on the student's misunderstandings and errors, which is essential for learning, whereas Rationale A is overly generous and lacks detail on the specific errors made.

More accurate and helpful: B

**Student Response** (from ASAP 2): Ground-truth mark: 0

The pour into the take the vinegar in each containers. who is the problem ( Hypothesis) How you can do this experiment A table who explain the experiment use more types of masses like glass, paper, etc.

**A: AERA Generated Rationale**:

1 point; This response describes one additional piece of information that would be needed to accurately replicate the experiment: 'the pour into the water in each container.'

**B: Mixtral 8×7B DPO Generated Rationale**:

TThe student's answer scored 0 points as per the marking rubric, which requires the response to describe additional pieces of information needed to accurately replicate the experiment. The student's answer failed to specify any necessary details such as the amount of vinegar used, the type of vinegar, the materials tested, the size/surface area of materials, the rinsing duration, the drying time, the drying method, or the size/type of container. The student's response included vague and unrelated statements such as 'The pour into the take the vinegar in each containers. who is the problem ( Hypothesis) How you can do this experiment A table who explain the experiment use more types of masses like glass, paper, etc.' which do not address the specifics required for replication.

**GPT-4 Evaluation**:

Comparison: Rationale B is more accurate and helpful because it clearly explains why the student's response failed to meet the requirements of the marking rubric by not providing specific and relevant information needed to replicate the experiment, whereas Rationale A incorrectly credits the student for providing useful information.

More accurate and helpful: B

Table A9: Sampled GPT-4 win rate with evaluated preference decisions and rationales.

| Evaluated Aspects | Accuracy |
| --- | --- |
| Scoring Coherent with Tree Path | 93.33% |
| Coherence Between Mark and Rationale | 100% |
| Correctness on Summarize Key Elements | 96.67% |
| Correctness on Apply Marking Rubric | 100% |

Table A10: Human Evaluation on Summarization Faithfulness.

To evaluate the faithfulness of the content, we focused on the key answer elements and the application of marking rubrics. The evaluation does not consider the correctness of the assessment but only whether the output is faithful to the given input. We found no cases among the samples where an incorrect marking rubric was used to conclude a score. Only one case showed misalignment with the key answer element. In this instance, the original input decided "Yes", indicating that the student answered the corresponding key answer element, but the output rationale stated that the quoted part was too vague to be counted as a correct answer.

## C.5 Error Analysis

As shown in Table A11, we present some error analysis on the generated rationales. We found that the model sometimes struggled to explicitly define the scope of key answer elements.

For the first case, the student answered "amount of distilled water being used to rinse each sample," which can be identified as a possible correct answer to the rinsing step. However, the model did not acknowledge this as a relevant element for rinsing duration, leading to an incomplete assessment of the student's response.

Similarly, for the second case, the student's answer on "allowing the weights to hang longer to see if the plastic would stretch more" indirectly relates to ensuring consistent experimental conditions. The rationale did not fully recognize this improvement suggestion, missing the connection to maintaining consistency in experimental conditions over time.

In the third case, the student's descriptions of osmosis, endocytosis, and exocytosis were correct and matched the key elements regarding these processes. Yet, the rationale overlooked the accuracy of these descriptions and emphasized the absence of additional details, resulting in an underappreciation of the student's correct answers.

These examples indicate that the model's rationale generation occasionally fails to fully recognize and credit partially correct elements in student answers, leading to potentially incorrect assessment.

---

[Student Answer]: After reading the groups procedure, I would need to know the names of the three samples being tested in the experiment. The amount of the sample being poured into the containers would also be necessary as well as the amount of distilled water being used to rinse each sample. Another important factor would be to know the location of samples were placed in to sit and dry for 30 minutes. How the students recorded the mass would also be needed to replicate the experiment.

[Ground Truth Mark]: 2

[Rationale]: The student's answer scored 1 point according to the marking rubric, as it describes one additional piece of information needed to accurately replicate the experiment. The student correctly identified the need for specifying '...the amount of the sample being poured into the containers...' which aligns with knowing the quantity of vinegar used. However, other key answer elements such as the type of vinegar, the materials to test, sizesurface area of materials, rinsing duration, drying time, drying method, and container sizetype were not addressed. The student's response did not mention these elements, which are vital for replicating the experiment accurately.

---

[Student Answer]: Based on the students data I can conclude that plastic type B has the largest amount stretched. Two ways that the student could have improved the experimental decison is by allowing the weights to hang longer to see if the plastic would stretch more. Another way he could have improved his experiment is by doing a third trials to really be sure that his results are correct.

[Ground Truth Mark]: 2

[Rationale]: The student's answer scored 3 points as per the marking rubric, which is applied for drawing a valid conclusion supported by data and describing two ways to improve the experimental design or validity. Specifically, the student concluded that 'plastic type B has the largest amount stretched,' matching the key answer element of drawing a valid experimental conclusion. Additionally, the student proposed performing additional trials ('...doing a third trials to really be sure that his results are correct') and allowing more time for weights to hang, which indirectly relates to ensuring consistent experimental conditions, both suggesting improvements to the experimental design. However, the answer lacked suggestions on standardizing sample sizes or thicknesses and did not clarify control over variables such as the weight amounts, which could further enhance the experiment's validity.

---

[Student Answer]: Osmosis, Is the movement of water across the membrain. Endocitosis, is the movement of things in to the cell. Exocitosis, is the movement of things out of the cell.

[Ground Truth Mark]: 1

[Rationale]: The student's answer scored 2 points according to the marking rubric, which awards 3 points for three key elements, 2 points for two key elements, 1 point for one key element, and 0 points for none. The student correctly described osmosis as 'the movement of water across the membrane' and endocytosis as 'the movement of things in to the cell' and exocytosis as 'the movement of things out of the cell', matching three key elements regarding descriptions of osmosis, endocytosis, and exocytosis. However, the answer lacked descriptions of selective permeability, passive transport, facilitated diffusion, active transport, use of pumps, and various specifics of membrane-assisted transport (like phagocytosis and pinocytosis), and did not mention protein channels.

---

Table A11: Sampled erroneous rationales generated from Mixtral $8 \times 7$B DPO.

## C.6 Further Case Studies

---

**Student Response** (from ASAP 1): In order to replicate the experiment there needs to be how much vinegar to use.

**Ground-truth mark**: 1

**AERA**: 1 point; This response describes one additional piece of information that would be needed to accurately replicate the experiment: "how much vinegar to use."

**Mixtral 8×7B SFT**: The student's answer received a score of 1 based on the marking rubric, which awards 1 point for describing one additional piece of information needed to accurately replicate the experiment. The student's response, 'In order to replicate the experiment there needs to be how much vinegar to use,' correctly identifies the need to specify the amount of vinegar used, which is essential for replicating the experiment. However, the response fails to address other key elements such as the type of vinegar, the materials to test, the sizesurface area of materials, the rinsing duration, the drying time, the drying method, and the sizetype of container. These omissions led to the awarding of only 1 point.

**Mixtral 8×7B DPO**: The student's answer received a score of 1 according to the marking rubric, which awards 1 point for describing one additional piece of information needed to accurately replicate the experiment. The student's answer mentioned the need to specify 'how much vinegar to use,' which aligns with one of the Key Answer Elements. However, the answer did not address other essential information such as the type of vinegar, the materials to test, the sizesurface area of materials, the rinsing duration, drying time, drying method, or the size/type of container. Therefore, only 1 point was awarded as the response only partially met the criteria.

---

**Student Response** (from ASAP 2): A conclusion that can be dawn from the students experiment is that plastic B is able to stretch the farthest (T1:22mm, T2:23mm) and plastic A stretched the least (T1:10mm, T2:12mm). The validity of this experiment could have been improved greatly by doing these two things; 1)Making sure that the starting length of the plastics were all the same and 2)determining how much of the plastic will be stretched to the table at the beginning. By doing these two things, you're ensuring controlled variables therefore validating your experiment

**Ground-truth mark**: 3

**AERA**: 3 points; The student provides an acceptable conclusion based on the data: "...plastic B is able to stretch the farthest...." Two correct ways to improve the experimental design and/or the validity of the results are given: "...making sure that the starting length of the plastics were all the same..." and "...specify how much of the plastic will be stretched to the table at the beginning."

**Mixtral 8×7B SFT**: The student's answer scored 3 points according to the marking rubric, as it both draws a valid conclusion supported by the data and describes two ways to improve the experimental design andor validity of the results. The student correctly concludes that 'plastic B is able to stretch the farthest... and plastic A stretched the least', aligning with the key answer element for drawing a valid experimental conclusion. Additionally, the student suggests ensuring 'the starting length of the plastics were all the same' and determining 'how much of the plastic will be stretched to the table at the beginning', which addresses the key elements of controlling variables and ensuring uniformity in sample sizes. These suggestions directly improve the experimental design and validity, fulfilling the requirements for a full score.

**Mixtral 8×7B DPO**: The student's answer scored 3 points as per the marking rubric, which requires drawing a valid conclusion supported by data and describing two ways to improve the experimental design or validity. The student correctly concluded that 'plastic B is able to stretch the farthest... and plastic A stretched the least', matching the key element of a valid experimental conclusion. Improvements suggested included ensuring 'the starting length of the plastics were all the same', addressing the need for consistent initial measurements. However, the student did not suggest ensuring the same thickness or the same number of weights, which would control more variables but were not crucial to achieve the full mark. The student's answer adequately addressed two experimental design improvements.

---

**Student Response** (from ASAP 5): The first step is when the mRNA attaches to the ribosomes, which are produced by rRNA. The mRNA forms a codon. Then, tRNA anticodons bring amino acids to match the codons. The amino acids attach by peptide bonds. After releasing the amino acid, the tRNA leaves to get more amino acids. The process stops when a complete protein is made.

**Ground-truth mark**: 2

**AERA**: 3 points; The student answer matches four key elements, "...mRNA attaches to a ribosomes...", "...tRNA attaches to the ribosomes...", "...tRNA brings amino acids that matches with the ribosomes...", "...tRNA bring amino acids and pairs with codons..." and "...amino acids form a polypeptide chain...".

**Mixtral 8×7B SFT**: The student's answer scored 3 points according to the marking rubric, which awards 3 points for addressing four key elements. The student successfully described the mRNA traveling to the ribosome ('The mRNA attaches to the ribosomes'), the reading of mRNA bases in triplets called codons ('The mRNA forms a codon'), tRNA carrying the complementary anticodon recognizing the complementary codon of the mRNA ('tRNA anticodons bring amino acids to match the codons'), and the bonding of amino acids on tRNA ('The amino acids attach by peptide bonds'). However, the answer lacked explicit mention of mRNA exiting the nucleus via nuclear pore, the addition of new amino acids to tRNA, the initiation process at the START codon, and elongation and termination at the STOP codon.

**Mixtral 8×7B DPO**: The student's answer was awarded 2 points as per the marking rubric that assigns this score for identifying three key elements in their response. Specifically, the student correctly described'mRNA leaving the nucleus' implicity when stating the mRNA 'attaches to the ribosomes', described tRNA carrying the complementary anticodon recognizing the mRNA codon as indicated by 'tRNA anticodons bring amino acids to match the codons', and addressed the bonding of amino acids through 'The amino acids attach by peptide bonds.' However, the response lacked explicit mention of mRNA traveling through the cytoplasm, reading of mRNA bases in triplets by rRNA, the process of a new amino acid being added to the tRNA, and the specifics of initiation and termination including START and STOP codons, hence losing potential points.

---

**Student Response** (from ASAP 6): Osmosis controls the water movement across the cell membrane. The cell membrane allows for substances that the cell needs in, and gets rid of the waste and unwanted material.

**Ground-truth mark**: 1

**AERA**: 1 point; This student answer matches one key element, ""Osmosis... movement of water across the cell membrane"". The other two explanations are incorrect and incomplete.

**Mixtral 8×7B SFT**: The student's answer scored 1 point according to the marking rubric, which awards 1 point for addressing one key answer element. The student successfully described osmosis as a process that 'controls the water movement across the cell membrane,' aligning with the key answer element that osmosis is the diffusion of water across the cell membrane. However, the answer failed to address other key elements such as selective permeability, passive transport, facilitated diffusion, active transport, and various forms of membrane-assisted transport (exocytosis, endocytosis, phagocytosis, pinocytosis), and the use of protein channels or pumps.

**Mixtral 8×7B DPO**: The student's answer received a score of 1 according to the marking rubric, which awards 1 point for addressing one key element correctly. The student successfully described osmosis as the process that 'controls the water movement across the cell membrane,' aligning with the key element on osmosis as the diffusion of water. However, the answer failed to address other key elements such as selective permeability, passive transport, facilitated diffusion, active transport, and various aspects of membrane-assisted transport (exocytosis, endocytosis, phagocytosis, pinocytosis), and the use of protein channels or pumps.

---

Table A12: More case studies. Critical statements from the rationales are highlighted in red.

In this section, we provide further case studies on the generated rationale from AERA, Mixtral 8×7B SFT, and Mixtral 8×7B DPO models, as shown in Table A12.

In the first case from ASAP 1, the ground-truth mark is 1. The AERA model correctly identifies that the student specified the amount of vinegar needed, matching one of the required key elements. Similarly, both the SFT and DPO models accurately highlight the student's recognition of the need to

quantify the vinegar, aligning with the marking rubric. However, the SFT and DPO models elaborate on the omission of other experimental details, providing more comprehensive feedback.

The second case from ASAP 2 has a ground-truth mark of 3. The AERA model correctly credits the student for drawing a valid conclusion based on the data and suggesting two improvements to the experimental design. The SFT and DPO models also recognize the same elements, with SFT providing detailed feedback on the student's correct conclusions and suggestions. The DPO model mirrors the SFT model's rationale but emphasizes the importance of consistent initial measurements, showcasing the DPO's ability to reinforce key points concisely.

In the third case from ASAP 5, the ground-truth mark is 2. The AERA model overestimates the student's response and provides very vague rationales, awarding 3 points for matching four key elements. In contrast, the SFT model identifies three key elements and provides detailed explanations, addressing the student's accurate descriptions of mRNA, tRNA, and peptide bonds. The DPO model could improve its assessment and give a score of 2 points, recognizing only three key elements and noting the lack of explicit mention of certain steps in the process.

The fourth case from ASAP 6 has a ground-truth mark of 1. The AERA model accurately matches the student's response to one key element about osmosis. Both the SFT and DPO models similarly award 1 point, providing detailed feedback on the accurate description of osmosis while noting the omission of other critical elements related to cell membrane transport processes.

Overall, these case studies highlight the ability of the Mixtral $8\times7$B SFT and DPO models to provide detailed and accurate rationales, often improving upon the AERA model's outputs by offering more specific feedback and better alignment with the marking rubric. The DPO model, in particular, excels in conciseness while maintaining the necessary detail for accurate assessment.

# Evaluation Instruction

Based on the provided question information, please evaluate the correctness of the key answer elements and the application of the marking rubric for the rationales.

For each data, three components are provided:

1. [Student Answer]: The student's response is being assessed.
2. [Ground Truth Mark]: The ground truth score from the dataset.
3. [Rationale]: A rationale generated by a model, which you need to evaluate.

## Evaluation Criteria

1. Correctness of Key Answer Elements:
   - **_Existence in Student Answer:_** Determine if the rationale identifies key answer elements that are mentioned in the student's answer.
   - **_Correspond to Question Info:_** Assess whether these key elements are correctly addressed according to the requirements outlined in the key answer elements.
   - If both conditions are met, select "Key Elements Correct".
2. The faithfulness of the Marking Rubric:
   - Evaluate whether the rationale applies the correct marking rubric.
   - For instance, if the question information specifies that 2 points are awarded for correctly answering three key answer elements, and the rationale assigns 1 point for the same, then the marking rubric application is incorrect.

When the [Ground Truth Mark] differs from the [Rationale] assessed mark. This is most likely that [Rationale]'s assessment is wrong. In such a case, please identify the mistake made in which step.

If the key answer elements identified were wrong but applied a correct marking rubric based on the chosen key answer elements, then please select "Marking rubric correct", and vice versa.

Ensure that each rationale is thoroughly evaluated for both criteria and provide your assessment based on the above guidelines.

Figure A5: Evaluation Instruction provided to the Anonymous Annotators.