AnchorAlign: Self-Explanations Enhancement via Anchored Alignment

Anonymous ACL submission

Abstract

In real-world applications, human-annotated rationales are often scarce or prohibitively expensive, making classification datasets a more accessible alternative. As a result, supervised fine-tuning (SFT) of large language models (LLMs) using only classification data is a widely adopted strategy for domain-specific adaptation. However, our analysis reveals that while SFT enhances task-specific accuracy, it weakens a model's ability to justify its reasoning—its *self-explanation* capability—underscoring the need for methods that improve explanation quality without compromising classification performance. To address this, we propose ANCHORALIGN, an end-to-end framework that aligns LLMs on classification tasks while enhancing their ability to produce meaningful self-explanations. ANCHORALIGN leverages ground-truth labels from classification datasets to enhance the creation of self-preference datasets. It categorizes model behavior in response to each input prompt into three groups—consistently correct, consistently incorrect, and variable—and applies tailored strategies to enhance preference-pair selection, improving the effectiveness of Direct Preference Optimization (DPO). Experimental results demonstrate that ANCHORALIGN consistently enhances explanation quality while preserving classification accuracy, outperforming alignment strategies that rely solely on judge-based evaluations.

1 Introduction

Real-world applications often face the challenge that datasets containing human-annotated rationales are either scarce or prohibitively expensive compared to classification datasets. In scenarios where only classification datasets are available for domain-specific adaptation, supervised fine-tuning (SFT) may lead to improved precision in the classification task but at the cost of compromising the model's generalization capabilities in other areas (Yang et al., 2024; Kirk et al., 2024).

In our study, we first investigate how SFT using only classification datasets for domain-specific adaptation can degrade performance on a secondary task—specifically, the model's ability to justify their own reasoning, a skill referred to as *self-explanation* (Madsen et al., 2024a).

To address this, we propose a framework for the automated qualitative assessment of self-explanations that are plausible to humans (Agarwal et al., 2024), leveraging an LLM-as-evaluator. Our framework evaluates a model's ability to generate holistic explanations—defined as those that excel across multiple criteria and capture the characteristics of high-quality explanations. Our findings indicate that while SFT improves task-specific accuracy, it often degrades self-explanation quality, highlighting the need for methods that enhance explanation quality without compromising accuracy gains.

The lack of annotated data of both high- and low-quality explanations can be framed in the context of model aligning without human preference data. Recent research has explored ways to align LLMs without direct human input. Some approaches generate self-instruct data to fine-tune models (Wang et al., 2023; Chen et al., 2023; Gulcehre et al., 2023), while others, like (Bai et al., 2022; Yuan et al., 2024; Wu et al., 2024), use LLM-generated feedback to train reward models.

Building on these advancements, we propose an end-to-end approach to align LLMs on downstream tasks while simultaneously ensuring the generation of high-quality self-explanations, in scenarios where only classification datasets are available for domain-specific adaptation. Since classification datasets inherently contain ground-truth labels, we leverage this information to design probes that improve the creation of self-preference datasets. We refer to this approach as ANCHORALIGN, a method

that enhances preference pair selection by categorizing model responses for a given input prompt into three distinct groups: consistently correct, consistently incorrect, and variable. For each category, we apply tailored strategies to construct preference pairs, which are then used in the Direct Preference Optimization (DPO) phase (Rafailov et al., 2023).

While our approach relies on ground-truth annotations from the classification task, such labels are naturally available in the domain adaptation settings we address. Our results demonstrate that ANCHORALIGN consistently improves explanation quality, mitigating the degradation typically caused by SFT. Moreover, we show that anchor preference pairs outperform self-alignment strategies that rely solely on judge-based evaluations for preference pair selection.

2 Related Work

LLM-as-Evaluator: This concept refers to the ability of large language models (LLMs) to evaluate the outputs of other LLMs, a technique commonly referred to as LLM-as-a-Judge. This approach has gained considerable traction in recent years (Dubois et al., 2023; Li et al., 2024; Fernandes et al., 2023; Bai et al., 2023) and is frequently used to assess LLM performance across various downstream tasks. It has proven particularly effective in automating evaluations, as demonstrated on platforms like LMSys Chatbot Arena (Zheng et al., 2023). Key implementations include direct scoring based on specific criteria (Bai et al., 2023), pairwise comparisons (Liu et al., 2024), and ensemble methods (Verga et al., 2024). While LLM-as-a-Judge offers scalability and consistency, it can also inherit biases from the evaluation model, potentially amplifying problematic outputs (Huang et al., 2024). Despite these challenges, it remains a valuable tool due to its efficiency and cost-effectiveness in evaluating LLM systems. In our work, we introduce a framework for the qualitative assessment of *self-explanations* using the LLM-as-a-Judge technique, designed to evaluate how effectively a model conveys its reasoning.

Self-Alignment: Several approaches have been developed to improve LLMs without requiring human-annotated feedback. One method involves fine-tuning models using high-quality, self-generated input-output pairs (Wang et al., 2023; Chen et al., 2023; Gulcehre et al., 2023), though this can perpetuate biases in example selection without a clear mechanism for improving selection quality. Another influential approach is Constitutional AI (Bai et al., 2022), where an LLM provides feedback and refines responses, which are then used to train a separate, static reward model. Building on this concept, (Yuan et al., 2024) and Wu et al. (2024) proposed using the LLM itself as a dynamic reward model, eliminating the need for a static one. This allows for continuous improvement in both generation and evaluation capabilities through iterative training processes. In our work, we introduce a novel method for creating self-preference datasets. Our approach, called ANCHORALIGN, enhances preference pair selection by categorizing model behavior in response to each input prompt and applying tailored strategies for each category. Evaluating a model's consistency for a given input prompt requires a probing mechanism. In our setup, this probe—or anchor—is derived from the ground-truth labels in the classification dataset used for domain adaptation.

LLM-as-a-Debater: This adversarial approach aims to improve model performance through argumentation. In Perez et al. (2019), debaters are limited to extracting relevant statements from a source text, rather than generating original arguments. Du et al. (2023) extended this concept by involving multiple LLM instances to debate their individual responses over several rounds, eventually converging on a shared final answer. Khan et al. (2024) further developed this approach by using debate-like scenarios to challenge and refine model outputs through simulated arguments. In our work, we adopt the LLM-as-a-Debater approach in the role of a consultant, specifically following Khan et al. (2024), for cases where the model's response to certain input prompts is *consistently incorrect*. This strategy enables the creation of self-preference examples that avoid reinforcing problematic behavior.

3 A Framework for Qualitative Assessment of Self-Explanations

3.1 Quality Criteria for Effective Self-Explanations

We focus on the model's ability to generate holistic explanations, which we define as one that excels across multiple criteria, collectively shaping what qualifies as a plausible and high-quality explanation. This approach contrasts with previous work that emphasized specific trustworthiness metrics,

such as faithfulness (Madsen et al., 2024b,a; Lanham et al., 2023; Lyu et al., 2023; Turpin et al., 2023; Parcalabescu and Frank, 2024) and truthfulness (Zhang et al., 2024; Sharma et al., 2023; Burns et al., 2022; Joshi et al., 2024). A high-quality, holistic explanation may be unfaithful in the sense that it does not accurately represent the model's internal reasoning, or conversely, a faithful explanation that truly reflects the model's reasoning may suffer from a lack of clarity, reducing its quality. While aiming for explanations that fulfill a holistic explanation framework might incidentally enhance faithfulness, it is not a strict requirement.

We evaluate self-explanations based on the following criteria: logical coherence, clarity, relevance, depth of argumentation and factual accuracy (see Appendix A).

3.2 Self-Explanations Evaluation Methodology

Let $\mathcal{M}$ represent a LLM tasked with generating responses for a classification problem. Each response consists of two components: a self-explanation, denoted as ε_i, and a predicted classification label, $\hat{y}_i$, corresponding to an input prompt x_i. The self-explanation ε_i is produced by prompting the model to articulate its reasoning before providing a final prediction, following the Chain-of-Thought prompting strategy (Wei et al., 2022).

Our methodology is inspired by recent approaches that utilize LLMs as evaluators of other models' outputs (Dubois et al., 2023; Li et al., 2024; Fernandes et al., 2023; Bai et al., 2023; Saha et al., 2024). This approach has shown versatility, extending beyond simple evaluation to various applications in model improvement and self-alignment strategies. For instance, researchers have employed this framework to generate self-instruct data for fine-tuning models (Wang et al., 2023; Chen et al., 2023; Gulcehre et al., 2023) and to create feedback for training reward models (Bai et al., 2022; Yuan et al., 2024; Wu et al., 2024).

To ensure a more reliable evaluation, we use a judge model, $\mathcal{M}_{\text{Judge}}$, from a different family than the base model generating the self-explanations. This distinction is critical because models within the same family—regardless of their size—tend to share training data, which can introduce correlations and bias the results.

In this work, the judge model, $\mathcal{M}_{\text{Judge}}$, evaluates the quality of the self-explanations, ε_i, based on predefined criteria (described in Section 3.1). The evaluation process proceeds as follows:

1. For each criterion κ, $\mathcal{M}_{\text{Judge}}$ assigns a qualitative verdict $v_{i,\kappa}$ from the set {excellent, good, fair, poor, bad}. The prompt used by $\mathcal{M}_{\text{Judge}}$ is provided in Appendix M.

2. Each verdict $v_{i,\kappa}$ is mapped to a numerical score $s_{i,\kappa}$ (see Appendix C.1).

3. The overall score for an explanation, s_i, is computed as the sum of scores across all criteria: $s_i = \sum_{k=1}^{K} s_{i,k}$

To assess the quality of self-explanations generated by different models, we adopt a pairwise evaluation (see Appendix B) strategy consistent with previous work (Chen et al., 2023; Yuan et al., 2024; Wu et al., 2024).

4 Impact of Supervised Fine-Tuning on Self-Explanations

In this study, we first investigate how SFT, using only classification datasets for domain-specific adaptation, can degrade performance on a secondary task—specifically, the model's ability to *self-explain*.

To examine this effect, we supervised fine-tuned the base model, $\mathcal{M}_{\text{Base}}$, on classification datasets (the primary task) to obtain $\mathcal{M}_{\text{SFT}}$, simulating real-world scenarios where explanation annotations are unavailable. To ensure a realistic domain adaptation setting while preventing potential advantages from multi-task learning, we trained separate models for each task. During fine-tuning, the loss was computed only on the target tokens corresponding to the correct choice sentence, excluding both the system instruction and the question. Additionally, we generated the full text of the selected option to provide richer context and maintain the model's text generation capabilities. Further details on datasets and training configurations are provided in Section 6.1.

After obtaining $\mathcal{M}_{\text{SFT}}$, we evaluated its ability to generate both a self-explanation, ε_i, and a predicted classification label, $\hat{y}_i$, given an input prompt x_i. To assess explanation quality, we employed the methodology described in Section 3.

Our evaluation revealed a notable trade-off between classification accuracy and explanation quality. While SFT improved classification performance, it led to a substantial decline in self-explanation quality compared to the base model

(see Table 1). The average degradation[1] in $\mathcal{M}_{\text{SFT}}$, as assessed across judge models, ranged from 5.8% to 13.3% across benchmarks.

This decline aligns with prior findings that SFT enhances task-specific performance at the expense of a model's generalization capabilities (Yang et al., 2024; Kirk et al., 2024). Our results suggest that classification fine-tuning, by design, does not incentivize the model to articulate its reasoning. Since classification tasks primarily involve selecting predefined answers, the model becomes specialized in answer selection while neglecting explanation generation, leading to a deterioration in self-explanation quality.

These findings **underscore the need for alignment techniques** that preserve high-quality explanations in scenarios where datasets with annotated rationales are unavailable for fine-tuning.

5 Self-Explanation Alignment with Anchor Preference Pairs

We introduce a methodology for aligning LLMs to improve *self-explanation*, even in the absence of annotated rationales. While explanation data is often scarce or costly, classification datasets are more readily available. Our approach leverages these datasets for domain adaptation, ensuring practical applicability.

Building on prior work (Bai et al., 2022; Wang et al., 2023; Yuan et al., 2024; Wu et al., 2024), our framework incorporates self-preference dataset generation, LLM-based evaluation (*LLM-as-Judge*), preference pair selection, and model alignment. However, we introduce two key innovations: (1) an explanation quality assessment framework (Sections 3.1 and 3.2) and (2) ANCHORALIGN, a novel preference pair selection method to enhance DPO alignment.

Our approach consists of three main steps:

1. Fine-tune the base model, $\mathcal{M}_{\text{Base}}$, on a target classification task to obtain $\mathcal{M}_{\text{SFT}}$.

2. Construct a self-preference dataset using the anchor-based strategy, as detailed in Sections 5.2 and 5.2.

3. Apply DPO to align $\mathcal{M}_{\text{SFT}}$ using the self-preference dataset, yielding the final model, $\mathcal{M}_{\text{Anchor}}$.

[1]Degradation/improvement is measured as the deviation from the equilibrium point established by the pairwise evaluation of $\mathcal{M}_{\text{Base}}$ vs. $\mathcal{M}_{\text{Base}}$, which is 50%.

5.1 Self-Preference Dataset Creation

We generate self-preference data for alignment as follows:

1. **Generate candidate responses:** Sample N diverse pairs of explanations and predictions from $\mathcal{M}_{\text{SFT}}$, denoted as $\{\varepsilon_i^n, \hat{y}_i^n\}_{n=1}^N$, where ε_i^n represents the explanation for the n-th prediction $\hat{y}_i^n$ corresponding to the prompt x_i.

2. **Score responses:** Evaluate the self-explanations using the methodology described in Section 3.2, assigning a score s_i^n to each ε_i^n. To ensure a self-contained alignment process, we use $\mathcal{M}_{\text{Base}}$ as the judge ($\mathcal{M}_{\text{Judge}}$)[2]. This approach eliminates the need for external models during training. However, it is important to note that we employ a model from a different family for evaluation to reduce potential biases.

3. **Anchor Preference Pair Selection:** Construct preference pairs for the DPO phase using the ANCHORALIGN methodology detailed in Section 5.2.

5.2 ANCHORALIGN: Preference Pairs via Anchor Selection

We propose ANCHORALIGN, a method that improves preference pair selection by categorizing model responses to a given input prompt into three distinct groups—*consistently correct*, *consistently incorrect*, and *variable*. Each category follows a tailored strategy for constructing preference pairs, which are subsequently used in the DPO phase. Assessing a model's consistency for a given prompt requires a reliable ground truth reference, or *anchor*. To achieve this, we utilize classification task labels from the domain adaptation process as a probing mechanism, as these labels are naturally available in the settings we consider.

Preference Pairs for Consistently Correct Prompts: For input prompts x_i where $\mathcal{M}_{\text{SFT}}$ consistently produces correct answers (i.e., $\hat{y}_i^n = y_i$ for all $n \in \{1, \ldots, N\}$), preference pairs are constructed based on the quality of the explanations. Let s_i^n denote the score assigned by the judge $\mathcal{M}_{\text{Judge}}$ to the n-th explanation ε_i^n for prompt x_i. We define two sets: $\mathbb{A}_i^w = \{\varepsilon_i^n : s_i^n = \max_{j \in \{1,\ldots,N\}} s_i^j\}$, which contains all explanations

[2]We choose $\mathcal{M}_{\text{Base}}$ instead of $\mathcal{M}_{\text{SFT}}$ as the judge based on the observation that $\mathcal{M}_{\text{SFT}}$ exhibits a decline in explanation quality (see Section 4).

that achieve the highest score for prompt x_i, and $\mathbb{A}_i^l = \{\varepsilon_i^n : s_i^n < \max_{j\in\{1,\ldots,N\}} s_i^j\}$, which includes all explanations with scores lower than the maximum for prompt x_i.

Preference Pairs for Variable Performance: For input prompts x_i where $\mathcal{M}_{\text{SFT}}$ produces a mix of correct and incorrect predictions (i.e., $\hat{y}_i^n \neq y_i$ for some $n \in \{1,\ldots,N\}$), preference pairs are constructed contrastively. We define the set $\mathbb{B}_i^w = \{\varepsilon_i^n : \hat{y}_i^n = y_i\}$, which contains explanations associated with correct predictions. From this set, we extract $\mathbb{A}_i^w \subseteq \mathbb{B}_i^w$, the subset of explanations with the highest scores assigned by $\mathcal{M}_{\text{Judge}}$, i.e., $\mathbb{A}_i^w = \{\varepsilon_i^n \in \mathbb{B}_i^w : s_i^n = \max_{j\in\mathbb{B}_i^w} s_i^j\}$. The set $\mathbb{A}_i^l = \{\varepsilon_i^n : \hat{y}_i^n \neq y_i \text{ and } s_i^n < \max_{j\in\mathbb{A}_i^w} s_i^j\}$ contains explanations corresponding to incorrect predictions, with scores lower than the maximum score in $\mathbb{A}_i^w$.

Preference Pairs for Consistently Incorrect Prompts: For prompts where all predictions from $\mathcal{M}_{\text{SFT}}$ are incorrect (i.e., $\hat{y}_i^n \neq y_i$ for all $n \in \{1,\ldots,N\}$), all corresponding explanations are placed in the set $\mathbb{A}_i^l$. To generate a winning explanation, we employ the $\mathcal{M}_{\text{Base}}$ model in a consultant role, similar to the LLM-as-a-Debater approach proposed by Khan et al. (2024). Since the inference hyperparameters for the LLM in this consulting role might differ from those used during the generation of preference pairs, we refer to this model as $\mathcal{M}_{\text{Consultant}}$ to avoid confusion. Specifically, we provide the correct answer y_i to the LLM and request an argument supporting this answer, which is then assigned to the set $\mathbb{A}_i^w$ as the winning explanation.

Finally, preference pairs are constructed for each instruction prompt x_i by randomly sampling ε_i^w from $\mathbb{A}_i^w$ as the winning explanation and ε_i^l from $\mathbb{A}_i^l$ as the losing explanation. The resulting preference pair is denoted as $(x_i, \varepsilon_i^w, \varepsilon_i^l)$. The detailed methodology is presented in Algorithm 1.

6 Experiments

In all experiments, we used `Llama-3-8B-Instruct` as our base model and evaluated four distinct model configurations:

1. $\mathcal{M}_{\text{Base}}$: The unmodified base model.

2. $\mathcal{M}_{\text{SFT}}$: A supervised fine-tuned version of $\mathcal{M}_{\text{Base}}$, trained exclusively on classification tasks to simulate scenarios where explanation annotations are unavailable.

3. $\mathcal{M}_{\text{Rank}}$ (Baseline): Built upon $\mathcal{M}_{\text{SFT}}$, this model was further refined using DPO with a self-preference dataset composed of rank-ordered preference pairs, derived solely from judge-based evaluations of explanations. This methodology follows prior works (Bai et al., 2022; Wang et al., 2023; Yuan et al., 2024; Wu et al., 2024) and serves as our baseline for measuring the performance improvements of our proposed approach.

4. $\mathcal{M}_{\text{Anchor}}$ (Ours): Like $\mathcal{M}_{\text{Rank}}$, this model underwent additional refinement using DPO. However, instead of relying solely on judge-based rankings, it utilized a self-preference dataset constructed via our proposed preference pair selection method, ANCHORALIGN, as described in Section 5.2.

Both $\mathcal{M}_{\text{Rank}}$ and $\mathcal{M}_{\text{Anchor}}$ incorporate an additional DPO alignment phase with the self-preference dataset. Throughout our comparisons, we refer to these models collectively as *self-aligned models*, distinguishing them from $\mathcal{M}_{\text{SFT}}$.

6.1 Experimental Setup

Datasets: We selected four datasets for our experiments: `AQuA-Rat` (Ling et al., 2017), `ARC-Challenge` (Clark et al., 2018), `LogiQA` (Liu et al., 2020), and `OpenbookQA` (Mihaylov et al., 2018). These datasets are established benchmarks for reasoning tasks, requiring a challenging reasoning process, which makes them an ideal fit for evaluating the quality of self-explanations. A key factor in their selection was the size of their training sets, which provided a sufficient number of input prompts to support the creation of the self-preference dataset. For evaluation, we used the test split of each dataset. For detailed dataset descriptions, see Appendix F.

Self-Alignment Details: Appendix G provides further insights into the self-alignment process, which includes SFT-based domain adaptation, the creation of the self-preference dataset, and the subsequent alignment through DPO.

Evaluation: We evaluated our models along two key dimensions: prediction accuracy and self-explanation quality. To capture variability in model outputs, we generated $N = 16$ explanation-prediction pairs per input prompt. The inference settings were consistent with those used to create the self-preference dataset, with a temperature of

Algorithm 1 Generating Preference Pairs Via Anchor Selection

Input: Instruction prompt x_i, model predictions $\{\hat{y}_i^n\}_{n=1}^N$, true label y_i, judge model $\mathcal{M}_{\text{Judge}}$, debater model $\mathcal{M}_{\text{Consultant}}$
Output: Preference pairs $(x_i, \varepsilon_i^w, \varepsilon_i^l)$
Initialize: $\mathbb{A}_i^w \leftarrow \emptyset$, $\mathbb{A}_i^l \leftarrow \emptyset$
for each explanation ε_i^n **do**
    Compute score s_i^n from $\mathcal{M}_{\text{Judge}}$
end for
if $\hat{y}_i^n = y_i$ for all $n \in \{1, \ldots, N\}$ **then** ▷ Consistently Correct Prompts
    $\mathbb{A}_i^w \leftarrow \{\varepsilon_i^n : s_i^n = \max_{j \in \{1,\ldots,N\}} s_i^j\}$
    $\mathbb{A}_i^l \leftarrow \{\varepsilon_i^n : s_i^n = \min_{j \in \{1,\ldots,N\}} s_i^j\}$
else if $\hat{y}_i^n \neq y_i$ for some $n \in \{1, \ldots, N\}$ **then** ▷ Variable Performance Prompts
    $\mathbb{B}_i^w \leftarrow \{\varepsilon_i^n : \hat{y}_i^n = y_i\}$
    $\mathbb{A}_i^w \leftarrow \{\varepsilon_i^n \in \mathbb{B}_i^w : s_i^n = \max_{j \in \mathbb{B}_i^w} s_i^j\}$
    $\mathbb{A}_i^l \leftarrow \{\varepsilon_i^n : \hat{y}_i^n \neq y_i \wedge s_i^n < \max_{j \in \mathbb{A}_i^w} s_i^j\}$
else ▷ Consistently Incorrect Prompts
    $\mathbb{A}_i^l \leftarrow \{\varepsilon_i^n : \hat{y}_i^n \neq y_i \text{ for all } n \in \{1, \ldots, N\}\}$
    Generate argument $\varepsilon_i^{\text{Consultant}}$ using $\mathcal{M}_{\text{Consultant}}$ given y_i
    $\mathbb{A}_i^w \leftarrow \{\varepsilon_i^{\text{Consultant}}\}$
end if
Sample ε_i^w from $\mathbb{A}_i^w$
Sample ε_i^l from $\mathbb{A}_i^l$
Return $(x_i, \varepsilon_i^w, \varepsilon_i^l)$

$T = 0.6$, top-k set to 0.9, and the same prompt (see Appendix O).

To assess the quality of the explanations, we used `Mistral-Large-123B-Instruct-2407` and `Qwen2.5-72B-Instruct` as judges. We then conducted head-to-head comparisons of the self-explanation scores across all models.

It is important to note that both the base and aligned models used `Llama3-8B-Instruct`, while the evaluation was conducted using judges from a different family of models. This distinction is crucial as it helps mitigate potential biases that could arise from using models within the same family, which often share training data.

Ablation Study: To validate our design choices, we create variants of $\mathcal{M}_{\text{Anchor}}$ by combining different strategies outlined in Section 5.2 to construct the anchor-preference dataset. The variants studied include $\mathcal{M}_{\text{Anchor (CC)}}$, $\mathcal{M}_{\text{Anchor (CC+V)}}$, and $\mathcal{M}_{\text{Anchor (CC+V+CI)}}$, where "Consistently Correct" (CC), "Variant" (V), and "Consistently Incorrect" (CI) denote the respective strategies. Notably, $\mathcal{M}_{\text{Anchor (CC+V+CI)}}$ represents the full version of $\mathcal{M}_{\text{Anchor}}$, and the terms may be used interchangeably throughout the discussion.

6.2 Analysis of Self-Aligned Models

Prediction Accuracy: The self-preference dataset used during the DPO phase is designed with the primary objective of improving explanation quality rather than maximizing accuracy. However, it is essential to ensure that enhancing the models' self-explanation capabilities does not compromise their performance on the primary task, as measured by classification accuracy. To assess this, we compute the average classification accuracy[3] and the standard deviation across multiple evaluation runs ($N = 16$), as shown in Table 1.

The results indicate that the self-aligned models, $\mathcal{M}_{\text{Rank}}$ and $\mathcal{M}_{\text{Anchor}}$, across all tested strategy combinations, either maintain or improve upon the classification accuracy gains achieved by the seed model, $\mathcal{M}_{\text{SFT}}$, relative to the base model, $\mathcal{M}_{\text{Base}}$. Notably, while the accuracy performances of $\mathcal{M}_{\text{Anchor}}$ and $\mathcal{M}_{\text{Rank}}$ are similar across most tasks, there is one exception: $\mathcal{M}_{\text{Anchor}}$ significantly outperforms $\mathcal{M}_{\text{Rank}}$ on the `Aqua-Rat` dataset, with statistical significance. Further analysis of the variability in $\mathcal{M}_{\text{Anchor}}$'s performance across different datasets is provided in Section 6.3.

[3]The maximum value is bolded, and results marked with a (*) indicate no statistical difference from the top performer.

Dataset	$\mathcal{M}_{\text{Base}}$ Acc. (%)	$\mathcal{M}_{\text{Align}}$ Type	$\mathcal{M}_{\text{Align}}$ Acc. (%)	$\varepsilon\,(W+\frac{T}{2})$ Rate (%) ↑		
				J_{Qwen}	J_{Mistral}	J_{Avg}
AQuA Rat	$47.1_{\pm 2.9}$	$\mathcal{M}_{\text{SFT}}$	$47.7_{\pm 2.7}$	47.2	41.1	44.2
		$\mathcal{M}_{\text{Rank (Baseline)}}$	$48.3_{\pm 2.1}$	48.3	41.5	44.9
		$\mathcal{M}_{\text{Anchor (CC)}}$	$49.3_{\pm 3.1}{}^{*}$	49.0	43.3	46.2
		$\mathcal{M}_{\text{Anchor (CC+V)}}$	$48.3_{\pm 2.9}$	48.1	42.6	45.3
		$\mathcal{M}_{\text{Anchor (CC+V+CI)}}$	$\mathbf{51.1}_{\pm 3.0}$	**49.5**	**46.3**	**47.9**
ARC-Challenge	$76.4_{\pm 0.7}$	$\mathcal{M}_{\text{SFT}}$	$81.0_{\pm 0.7}$	32.0	41.3	36.7
		$\mathcal{M}_{\text{Rank (Baseline)}}$	$81.9_{\pm 1.1}{}^{*}$	48.2	49.2	48.7
		$\mathcal{M}_{\text{Anchor (CC)}}$	$81.6_{\pm 1.3}{}^{*}$	46.7	48.0	47.3
		$\mathcal{M}_{\text{Anchor (CC+V)}}$	$\mathbf{82.0}_{\pm 1.1}$	48.7	49.7	49.2
		$\mathcal{M}_{\text{Anchor (CC+V+CI)}}$	$\mathbf{82.0}_{\pm 0.9}$	**52.1**	**52.4**	**52.3**
LogiQA	$41.4_{\pm 1.1}$	$\mathcal{M}_{\text{SFT}}$	$45.2_{\pm 0.7}$	34.6	42.6	37.6
		$\mathcal{M}_{\text{Rank (Baseline)}}$	$46.0_{\pm 1.5}{}^{*}$	45.0	47.8	46.4
		$\mathcal{M}_{\text{Anchor (CC)}}$	$45.8_{\pm 1.4}{}^{*}$	46.8	49.2	48.0
		$\mathcal{M}_{\text{Anchor (CC+V)}}$	$46.1_{\pm 1.7}{}^{*}$	45.2	48.1	46.7
		$\mathcal{M}_{\text{Anchor (CC+V+CI)}}$	$\mathbf{46.6}_{\pm 2.2}$	**50.9**	**51.3**	**51.1**
OpenbookQA	$71.7_{\pm 1.3}$	$\mathcal{M}_{\text{SFT}}$	$\mathbf{87.4}_{\pm 1.1}$	36.2	46.3	41.3
		$\mathcal{M}_{\text{Rank (Baseline)}}$	$87.0_{\pm 1.1}{}^{*}$	45.1	48.9	46.5
		$\mathcal{M}_{\text{Anchor (CC)}}$	$87.1_{\pm 0.5}{}^{*}$	45.4	49.2	47.3
		$\mathcal{M}_{\text{Anchor (CC+V)}}$	$\mathbf{87.4}_{\pm 0.8}$	46.0	49.6	47.8
		$\mathcal{M}_{\text{Anchor (CC+V+CI)}}$	$87.0_{\pm 0.9}{}^{*}$	**46.9**	**49.6**	**48.3**

Table 1: **Comparison of Aligned Models.** The table presents the average accuracy alongside pairwise evaluations of self-explanation quality. Both base and aligned models use `LLama3-8B-Instruct`, while pairwise evaluations are conducted using `Mistral-Large-Instruct-2407` and `Qwen2.5-72B-Instruct` as judges. Additionally, in the ablation study, we report the performance of $\mathcal{M}_{\text{Anchor}}$ under various strategy combinations. These strategies include Consistently Correct (CC), Variant (V), and Consistently Incorrect (CI).

Self-Explanation Quality: Pairwise evaluations of self-explanation quality (see Table 1) show that the initial decline in explanation performance observed in $\mathcal{M}_{\text{SFT}}$ is partially inherited by both $\mathcal{M}_{\text{Rank}}$ and $\mathcal{M}_{\text{Anchor}}$, since they both use $\mathcal{M}_{\text{SFT}}$ as the seed model during the DPO alignment phase. Nevertheless, both $\mathcal{M}_{\text{Rank}}$ and $\mathcal{M}_{\text{Anchor}}$ achieve significant improvements in explanation quality over $\mathcal{M}_{\text{SFT}}$, with $\mathcal{M}_{\text{Anchor}}$ demonstrating the strongest performance across benchmarks and evaluation judges. When compared to the base model, $\mathcal{M}_{\text{Anchor}}$ shows similar performance, winning half of the benchmarks, and significantly narrows the gap in explanation quality introduced by $\mathcal{M}_{\text{SFT}}$ on the remaining benchmark datasets. Regarding the ablation study, we observe that the highest explanation quality is achieved when the strategies (CC), (V), and (CI) are combined.

6.3 Impact of Preference Pairs Category Distribution

We define λ as the proportion of the self-preference dataset used to align $\mathcal{M}_{\text{Anchor}}$, corresponding to preference pairs selected under the (CI) or (V) strategies (see Appendix J).

Since the (CI) and (V) cases are not explicitly distinguished—instead, they are treated the same as (CC) cases—when the DPO alignment phase relies solely on judge-assigned scores (as in $\mathcal{M}_{\text{Rank}}$), λ provides valuable insight into the improvements in both accuracy and explanation quality achieved by $\mathcal{M}_{\text{Anchor}}$ compared to $\mathcal{M}_{\text{Rank}}$, relative to dataset-specific characteristics.

In the case of $\mathcal{M}_{\text{Anchor}}$, the (CI) and (V) strategies ensure—assuming the self-explanation is faithful—that the winning explanation ε_i^w supports the ground-truth label y_i. For the (V) strategy, this is achieved by sampling ε_i^w from the set $\mathbb{B}_i^w = \{\varepsilon_i^n :$

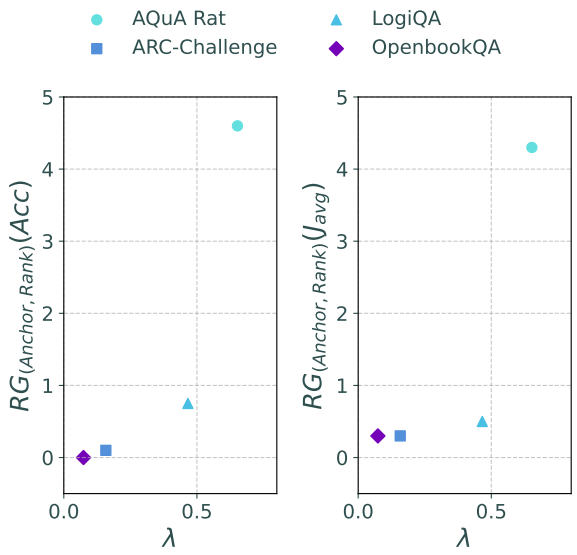

Figure 1: **Impact of Preference Pairs Category Distribution:** Presents the Relative Gains (RG) (see Appendix K) in accuracy (*left*) and J_{Avg} between $\mathcal{M}_{\text{Anchor}}$ and $\mathcal{M}_{\text{Rank}}$ (*right*) with respect to λ.

$\hat{y}_i^n = y_i\}$. Under the (CI) strategy, $\mathcal{M}_{\text{Consultant}}$ is employed to provide arguments that explicitly support y_i.

In contrast, $\mathcal{M}_{\text{Rank}}$ selects preference pairs solely based on scores assigned by judges during dataset creation. As a result, it does not guarantee that the winning explanations, ε_i^w, will support the ground-truth label y_i. The likelihood of selecting cases where ε_i^w aligns with an outcome different from y_i increases, particularly as λ grows.

We evaluated these improvements by analyzing the Relative Gains (RG) (see Appendix K) in accuracy and the average explanation quality score assigned by judges, J_{Avg}, between $\mathcal{M}_{\text{Anchor}}$ and $\mathcal{M}_{\text{Rank}}$ in relation to λ (see Figure 1). In both cases, we observed a trend indicating that $\mathcal{M}_{\text{Anchor}}$ demonstrates a greater relative improvement compared to $\mathcal{M}_{\text{Rank}}$ as λ increases. Conversely, when the alignment dataset consists primarily of (CC) instances, the performance of $\mathcal{M}_{\text{Anchor}}$ and $\mathcal{M}_{\text{Rank}}$ remains comparable. This supports our design principle that tailoring alignment strategies based on model behavior is crucial for improving the quality of self-preference datasets and avoiding the reinforcement of problematic behavior.

7 Analysis of Individual Evaluation Dimensions

Appendix H reports the average scores for each evaluation criterion used to assess self-explanations, as outlined in Section 3.1, across all evaluated models and benchmark datasets.

Overall, the self-aligned models outperform $\mathcal{M}_{\text{SFT}}$ across all evaluation criteria, with $\mathcal{M}_{\text{Anchor}}$ consistently achieving better results than $\mathcal{M}_{\text{Rank}}$.

Additionally, we observe that the degradation in self-explanation quality due to SFT varies significantly depending on the dataset used for fine-tuning. Two notable trends emerge from the analysis. First, for more complex tasks—where complexity is measured by lower test accuracy—such as `AQuA-Rat` and `LogiQA`, the decline in explanation quality is more pronounced across all criteria. Second, evaluation dimensions for which the base model originally received lower scores tend to experience a more significant drop in performance after SFT.

8 Conclusion

In this work, we introduce ANCHORALIGN, an end-to-end framework for aligning LLMs on classification tasks while enhancing their ability to generate high-quality self-explanations. Our approach addresses a key challenge in real-world applications: the scarcity of annotated rationales, which limits direct supervision for explanation quality.

ANCHORALIGN leverages ground-truth labels inherently available in classification datasets for domain adaptation to construct self-preference datasets. It categorizes model responses into three groups—consistently correct, consistently incorrect, and variable—applying targeted strategies to improve preference pair selection. These anchor preference pairs are then used in the DPO phase to refine explanation quality.

Our empirical results show that ANCHORALIGN consistently mitigates the degradation in explanation quality typically caused by SFT, ensuring models remain interpretable while maintaining classification performance gains. Furthermore, we demonstrate that ANCHORALIGN outperforms self-alignment strategies that rely solely on judge-based evaluations for preference pair selection.

9 Limitations

We acknowledge some limitations in our approach. First, evaluating the model's consistency on a given

input prompt requires a anchor—ground truth reference. Consequently, the selection of preference pairs via the anchor strategy relies on a classification task as the probing mechanism, which restricts its applicability. Second, when ranking the quality of self-explanations, we assign equal weights across all evaluation dimensions. This uniform weighting may not accurately reflect the varying significance of different aspects of explanation quality, which can differ depending on the user or specific application. Moreover, this approach may overlook instances where individual explanations degrade in separate criteria, potentially leading to preference pairs where score differences arise from unrelated factors.

References

Chirag Agarwal, Sree Harsha Tanneru, and Himabindu Lakkaraju. 2024. Faithfulness vs. Plausibility: On the (Un)Reliability of Explanations from Large Language Models. *International Conference on Machine Learning (ICML)*. ArXiv:2402.04614 [cs].

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, Carol Chen, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Eli Tran-Johnson, Ethan Perez, Jamie Kerr, Jared Mueller, Jeffrey Ladish, Joshua Landau, Kamal Ndousse, Kamile Lukosuite, Liane Lovitt, Michael Sellitto, Nelson Elhage, Nicholas Schiefer, Noemi Mercado, Nova DasSarma, Robert Lasenby, Robin Larson, Sam Ringer, Scott Johnston, Shauna Kravec, Sheer El Showk, Stanislav Fort, Tamera Lanham, Timothy Telleen-Lawton, Tom Conerly, Tom Henighan, Tristan Hume, Samuel R. Bowman, Zac Hatfield-Dodds, Ben Mann, Dario Amodei, Nicholas Joseph, Sam McCandlish, Tom Brown, and Jared Kaplan. 2022. Constitutional AI: Harmlessness from AI Feedback. *arXiv preprint*. ArXiv:2212.08073 [cs].

Yushi Bai, Jiahao Ying, Yixin Cao, Xin Lv, Yuze He, Xiaozhi Wang, Jifan Yu, Kaisheng Zeng, Yijia Xiao, Haozhe Lyu, Jiayin Zhang, Juanzi Li, and Lei Hou. 2023. Benchmarking Foundation Models with Language-Model-as-an-Examiner. *Advances in Neural Information Processing Systems (NeurIPS)*, 36:78142–78167.

Dan Biderman, Jacob Portes, Jose Javier Gonzalez Ortiz, Mansheej Paul, Philip Greengard, Connor Jennings, Daniel King, Sam Havens, Vitaliy Chiley, Jonathan Frankle, Cody Blakeney, and John Patrick Cunningham. 2024. LoRA Learns Less and Forgets Less. *Transactions on Machine Learning Research*.

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. 2022. Discovering Latent Knowledge in Language Models Without Supervision. *International Conference on Learning Representations (ICLR)*.

Lichang Chen, Shiyang Li, Jun Yan, Hai Wang, Kalpa Gunaratna, Vikas Yadav, Zheng Tang, Vijay Srinivasan, Tianyi Zhou, Heng Huang, and Hongxia Jin. 2023. AlpaGasus: Training a Better Alpaca with Fewer Data. *International Conference on Learning Representations (ICLR)*.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have Solved Question Answering? Try ARC, the AI2 Reasoning Challenge. *arXiv preprint*. ArXiv:1803.05457 [cs].

Yilun Du, Shuang Li, Antonio Torralba, Joshua B. Tenenbaum, and Igor Mordatch. 2023. Improving Factuality and Reasoning in Language Models through Multiagent Debate. *Advances in Neural Information Processing Systems (NeurIPS)*.

Yann Dubois, Chen Xuechen Li, Rohan Taori, Tianyi Zhang, Ishaan Gulrajani, Jimmy Ba, Carlos Guestrin, Percy S. Liang, and Tatsunori B. Hashimoto. 2023. AlpacaFarm: A Simulation Framework for Methods that Learn from Human Feedback. *Advances in Neural Information Processing Systems (NeurIPS)*, 36:30039–30069.

Patrick Fernandes, Daniel Deutsch, Mara Finkelstein, Parker Riley, André Martins, Graham Neubig, Ankush Garg, Jonathan Clark, Markus Freitag, and Orhan Firat. 2023. The Devil Is in the Errors: Leveraging Large Language Models for Fine-grained Machine Translation Evaluation. *Proceedings of the Eighth Conference on Machine Translation*, pages 1066–1083.

Caglar Gulcehre, Tom Le Paine, Srivatsan Srinivasan, Ksenia Konyushkova, Lotte Weerts, Abhishek Sharma, Aditya Siddhant, Alex Ahern, Miaosen Wang, Chenjie Gu, Wolfgang Macherey, Arnaud Doucet, Orhan Firat, and Nando de Freitas. 2023. Reinforced Self-Training (ReST) for Language Modeling. *arXiv preprint*. ArXiv:2308.08998 [cs].

Hui Huang, Yingqi Qu, Hongli Zhou, Jing Liu, Muyun Yang, Bing Xu, and Tiejun Zhao. 2024. On the Limitations of Fine-tuned Judge Models for LLM Evaluation. *arXiv preprint*. ArXiv:2403.02839 [cs].

Nitish Joshi, Javier Rando, Abulhair Saparov, Najoung Kim, and He He. 2024. Personas as a Way to Model Truthfulness in Language Models. *arXiv preprint*. ArXiv:2310.18168 [cs].

Akbir Khan, John Hughes, Dan Valentine, Laura Ruis, Kshitij Sachan, Ansh Radhakrishnan, Edward Grefenstette, Samuel R. Bowman, Tim Rocktäschel, and Ethan Perez. 2024. Debating with More Persuasive LLMs Leads to More Truthful Answers. *International Conference on Machine Learning (ICML)*, pages 23662–23733. ISSN: 2640-3498.

Robert Kirk, Ishita Mediratta, Christoforos Nalmpantis, Jelena Luketina, Eric Hambro, Edward Grefenstette, and Roberta Raileanu. 2024. Understanding the Effects of RLHF on LLM Generalisation and Diversity. *arXiv preprint*. ArXiv:2310.06452 [cs].

Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson Denison, Danny Hernandez, Dustin Li, Esin Durmus, Evan Hubinger, Jackson Kernion, Kamilė Lukošiūtė, Karina Nguyen, Newton Cheng, Nicholas Joseph, Nicholas Schiefer, Oliver Rausch, Robin Larson, Sam McCandlish, Sandipan Kundu, Saurav Kadavath, Shannon Yang, Thomas Henighan, Timothy Maxwell, Timothy Telleen-Lawton, Tristan Hume, Zac Hatfield-Dodds, Jared Kaplan, Jan Brauner, Samuel R. Bowman, and Ethan Perez. 2023. Measuring Faithfulness in Chain-of-Thought Reasoning. *arXiv preprint*. ArXiv:2307.13702 [cs].

Xian Li, Ping Yu, Chunting Zhou, Timo Schick, Omer Levy, Luke Zettlemoyer, Jason Weston, and Mike Lewis. 2024. Self-Alignment with Instruction Backtranslation. *arXiv preprint*. ArXiv:2308.06259 [cs].

Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. 2017. Program Induction by Rationale Generation: Learning to Solve and Explain Algebraic Word Problems. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 158–167.

Jian Liu, Leyang Cui, Hanmeng Liu, Dandan Huang, Yile Wang, and Yue Zhang. 2020. LogiQA: A Challenge Dataset for Machine Reading Comprehension with Logical Reasoning. *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, pages 3622–3628.

Yinhong Liu, Han Zhou, Zhijiang Guo, Ehsan Shareghi, Ivan Vulić, Anna Korhonen, and Nigel Collier. 2024. Aligning with Human Judgement: The Role of Pairwise Preference in Large Language Model Evaluators.

Qing Lyu, Shreya Havaldar, Adam Stein, Li Zhang, Delip Rao, Eric Wong, Marianna Apidianaki, and Chris Callison-Burch. 2023. Faithful Chain-of-Thought Reasoning. *International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational*, pages 305–329.

Andreas Madsen, Sarath Chandar, and Siva Reddy. 2024a. Are self-explanations from Large Language Models faithful? *Association for Computational Linguistics (ACL)*, pages 295–337.

Andreas Madsen, Siva Reddy, and Sarath Chandar. 2024b. Faithfulness Measurable Masked Language Models. *International Conference on Machine Learning (ICML)*, pages 34161–34202. ISSN: 2640-3498.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a Suit of Armor Conduct Electricity? A New Dataset for Open Book Question Answering. *Empirical Methods in Natural Language Processing (EMNLP)*, pages 2381–2391.

Letitia Parcalabescu and Anette Frank. 2024. On Measuring Faithfulness or Self-consistency of Natural Language Explanations. *Association for Computational Linguistics (ACL)*, pages 6048–6089.

Ethan Perez, Siddharth Karamcheti, Rob Fergus, Jason Weston, Douwe Kiela, and Kyunghyun Cho. 2019. Finding Generalizable Evidence by Learning to Convince Q&A Models. *Empirical Methods in Natural Language Processing (EMNLP)*, pages 2402–2411.

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. *Advances in Neural Information Processing Systems (NeurIPS)*, 36:53728–53741.

Swarnadeep Saha, Omer Levy, Asli Celikyilmaz, Mohit Bansal, Jason Weston, and Xian Li. 2024. Branch-Solve-Merge Improves Large Language Model Evaluation and Generation. *Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 8352–8370.

Mrinank Sharma, Meg Tong, Tomasz Korbak, David Duvenaud, Amanda Askell, Samuel R. Bowman, Newton Cheng, Esin Durmus, Zac Hatfield-Dodds, Scott R. Johnston, Shauna Kravec, Timothy Maxwell, Sam McCandlish, Kamal Ndousse, Oliver Rausch, Nicholas Schiefer, Da Yan, Miranda Zhang, and Ethan Perez. 2023. Towards Understanding Sycophancy in Language Models. *arXiv preprint*. ArXiv:2310.13548 [cs, stat].

Megh Thakkar, Quentin Fournier, Matthew Riemer, Pin-Yu Chen, Amal Zouaq, Payel Das, and Sarath Chandar. 2024. A Deep Dive into the Trade-Offs of Parameter-Efficient Preference Alignment Techniques. *Association for Computational Linguistics (ACL)*, pages 5732–5745.

Miles Turpin, Julian Michael, Ethan Perez, and Samuel Bowman. 2023. Language Models Don't Always Say What They Think: Unfaithful Explanations in Chain-of-Thought Prompting. *Advances in Neural Information Processing Systems (NeurIPS)*, 36:74952–74965.

Pat Verga, Sebastian Hofstatter, Sophia Althammer, Yixuan Su, Aleksandra Piktus, Arkady Arkhangorodsky, Minjie Xu, Naomi White, and Patrick Lewis. 2024. Replacing Judges with Juries: Evaluating LLM Generations with a Panel of Diverse Models. *arXiv preprint*. ArXiv:2404.18796 [cs].

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. Self-Instruct: Aligning Language Models with Self-Generated Instructions. *Association for Computational Linguistics (ACL)*, pages 13484–13508.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. *Advances in Neural Information Processing Systems (NeurIPS)*, 35:24824–24837.

Tianhao Wu, Weizhe Yuan, Olga Golovneva, Jing Xu, Yuandong Tian, Jiantao Jiao, Jason Weston, and Sainbayar Sukhbaatar. 2024. Meta-Rewarding Language Models: Self-Improving Alignment with LLM-as-a-Meta-Judge. *International Conference on Learning Representations (ICLR)*. ArXiv:2407.19594 [cs].

Haoran Yang, Yumeng Zhang, Jiaqi Xu, Hongyuan Lu, Pheng Ann Heng, and Wai Lam. 2024. Unveiling the Generalization Power of Fine-Tuned Large Language Models. *arXiv preprint*. ArXiv:2403.09162 [cs].

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason Weston. 2024. Self-Rewarding Language Models. *Advances in Neural Information Processing Systems (NeurIPS)*. ArXiv:2401.10020 [cs].

Xiaoying Zhang, Baolin Peng, Ye Tian, Jingyan Zhou, Lifeng Jin, Linfeng Song, Haitao Mi, and Helen Meng. 2024. Self-Alignment for Factuality: Mitigating Hallucinations in LLMs via Self-Evaluation. *arXiv preprint*. ArXiv:2402.09267 [cs].

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. *Advances in Neural Information Processing Systems (NeurIPS)*.

A Quality Criteria for Effective Self-Explanations

We evaluate self-explanations based on the following criteria [4]:

1. **Logical coherence**: The explanation should follow a clear and logical reasoning process, with all components cohesively connected to form a unified, non-contradictory narrative.

2. **Clarity**: The explanation must present ideas clearly and precisely, using appropriate terminology to effectively communicate complex concepts without unnecessary complexity.

3. **Relevance**: The explanation should comprehensively address the task at hand, directly answering the specific context or requirements without omitting critical information.

4. **Depth of argumentation**: The explanation must provide strong reasoning and credible evidence to support its conclusions, reflecting a deep understanding of the task.

5. **Factual accuracy**: This criterion assesses the correctness of individual claims within the explanation. While related to truthfulness, factual accuracy focuses on whether specific statements align with established knowledge.

[4] Appendix L provides a complementary analysis of the correlation between LLM and human judges across the evaluation criteria.

B Pairwise Model Evaluation

To compare the performance of two models, denoted as $\mathcal{M}_1$ and $\mathcal{M}_2$, we perform a pairwise evaluation of the self-explanations generated for a given prompt x_i. Each model produces N explanations, and we compare each explanation from $\mathcal{M}_1$ with every explanation from $\mathcal{M}_2$, resulting in N^2 pairwise comparisons.

For a given comparison between the n-th explanation from model $\mathcal{M}_1$ and the m-th explanation from model $\mathcal{M}_2$, where $n, m \in \{1, \ldots, N\}$, we compare the corresponding scores, $s_i^n(\mathcal{M}_1)$ and $s_i^m(\mathcal{M}_2)$. A *win* for $\mathcal{M}_1$ is recorded if the score from $\mathcal{M}_1$ is strictly greater than that from $\mathcal{M}_2$:

$$s_i^n(\mathcal{M}_1) > s_i^m(\mathcal{M}_2)$$

Conversely, a *loss* for $\mathcal{M}_1$ occurs if the score from $\mathcal{M}_1$ is strictly less than the score from $\mathcal{M}_2$:

$$s_i^n(\mathcal{M}_1) < s_i^m(\mathcal{M}_2)$$

A *tie* is defined when both scores are equal:

$$s_i^n(\mathcal{M}_1) = s_i^m(\mathcal{M}_2)$$

For each prompt x_i, we count the total number of wins, losses, and ties across all N^2 comparisons between the explanations from both models. To summarize the performance of the models across the entire dataset, we compute the win rate, tie rate, and loss rate.

The win rate $W(\mathcal{M}_1, \mathcal{M}_2)$ is the average proportion of pairwise comparisons in which model $\mathcal{M}_1$ outperforms model $\mathcal{M}_2$ across all prompts in the set $\mathcal{X}$. It is computed as:

$$W(\mathcal{M}_1, \mathcal{M}_2) = \frac{1}{|\mathcal{X}|} \sum_{x_i \in \mathcal{X}} \left(\frac{1}{N^2} \sum_{n=1}^{N} \sum_{m=1}^{N} \mathbb{1}_{win} \right)$$

Here, $\mathcal{X}$ is the set of all prompts, and $\mathbb{1}[\cdot]$ is the indicator function, which returns 1 if the condition

inside the brackets is true (i.e., if $\mathcal{M}_1$ wins) and 0 otherwise.

Similarly, we define the tie rate $T(\mathcal{M}_1, \mathcal{M}_2)$ as the proportion of pairwise comparisons where the models perform equally:

$$T(\mathcal{M}_1, \mathcal{M}_2) = \frac{1}{|\mathcal{X}|} \sum_{x_i \in \mathcal{X}} \left(\frac{1}{N^2} \sum_{n=1}^{N} \sum_{m=1}^{N} \mathbb{1}_{tie} \right)$$

The loss rate $L(\mathcal{M}_1, \mathcal{M}_2)$ captures the proportion of comparisons where $\mathcal{M}_1$ performs worse than $\mathcal{M}_2$:

$$L(\mathcal{M}_1, \mathcal{M}_2) = \frac{1}{|\mathcal{X}|} \sum_{x_i \in \mathcal{X}} \left(\frac{1}{N^2} \sum_{n=1}^{N} \sum_{m=1}^{N} \mathbb{1}_{loss} \right)$$

To measure overall performance, we define the *win overall rate*, combining wins and half of the ties:

$$W_{\text{overall}} = W + \frac{1}{2}T.$$

Throughout the evaluations presented in this work, $\mathcal{M}_2$ refers to the baseline model $\mathcal{M}_{\text{Base}}$.

C Judge Component

The judge model $\mathcal{M}_{\text{Judge}}$ evaluates the quality of self-explanation, denoted as ε_i, associated with an input prompt x_i. based on predefined criteria, which are elaborated in Section 3.1. The evaluation process proceed as follows:

1. For each criterion κ, $\mathcal{M}_{\text{Judge}}$ assigns a qualitative verdict $v_{i,\kappa}$ from the set {excellent, good, fair, poor, bad}. The prompt used by $\mathcal{M}_{\text{Judge}}$ is provided in Appendix M.

   $$\mathcal{M}_{\text{Judge}}(x_i, \varepsilon_i) \rightarrow \{v_{i,\kappa}\} \text{ for } \kappa \in \{1, \ldots, K\}$$

2. Each verdict $v_{i,\kappa}$ is mapped to a numerical score $s_{i,\kappa}$ (see Appendix C.1).

3. The overall score for an explanation, s_i, is computed as the sum of scores across all criteria:

   $$s_i = \sum_{k=1}^{K} s_{i,k}$$

C.1 Judge Score Mapping

Each verdict $v_{i,\kappa}$, assigned by $\mathcal{M}_{\text{Judge}}$ for criterion κ on self-explanation ε_i corresponding to prompt x_i, is mapped to a numerical score $s_{i,\kappa}$ as follows:

$$s_{i,\kappa} = \begin{cases} 1.0 & \text{if } v_{i,\kappa} = \texttt{Excellent}, \\ 0.8 & \text{if } v_{i,\kappa} = \texttt{Good}, \\ 0.6 & \text{if } v_{i,\kappa} = \texttt{Fair}, \\ 0.2 & \text{if } v_{i,\kappa} = \texttt{Poor}, \\ 0.0 & \text{if } v_{i,\kappa} = \texttt{Bad}. \end{cases}$$

Higher scores $s_{i,\kappa}$ indicate superior performance.

D Consultant Component

In cases where the model $\mathcal{M}_{\text{SFT}}$ behaves consistently incorrectly for the input prompt x_i, we employ the model $\mathcal{M}_{\text{Base}}$ in a consultant role. Specifically, we provide the correct answer y_i to the LLM and request an explanation ε_i supporting this answer.

$$\mathcal{M}_{\text{Consultant}}(x_i, y_i) \rightarrow \varepsilon_i$$

E Inference Parameters

Table 2 summarizes the inference parameters, including temperature and top-k, used for each component, such as the judge, consultant, and sampler.

Component	Temperature	Top-k
Judge	0.0	
Consultant	0.5	0.9
Sampler	0.6	0.9

Table 2: **Inference parameters per component**

F Dataset Details

All datasets used in our experiments are established reasoning benchmarks. The questions, along with related context and answer options, were inserted into our template (provided in 10) and used as input prompts for the model.

F.1 AQuA-Rat

`AQuA-Rat` (Ling et al., 2017) contains approximately 100,000 algebraic word problems, each accompanied by a natural language rationale explaining the solution steps. Each problem includes a

question and five answer options. Due to computational constraints, we sampled 5,000 examples from the training set and used 254 test samples for our experiments. We utilized only the questions and answer options and excluded the provided rationales since our study aims to enhance the model's ability to generate self-explanations in natural language while performing the primary classification task, without relying on human-annotated reasoning patterns.

F.2 ARC-Challenge

The `ARC-Challenge` dataset (Clark et al., 2018) is a subset of the ARC dataset containing grade-school level, multiple-choice science questions. We used 1,119 training samples and 1,172 test samples from ARC-Challenge dataset. These samples are selected for being challenging, as they could not be answered correctly by either retrieval-based or word co-occurrence algorithms. In our experiments, we incorporated the questions and their corresponding four answer options. We omitted the associated corpus of sentences to focus purely on the model's reasoning capabilities rather than external knowledge retrieval.

F.3 LogiQA

`LogiQA` (Liu et al., 2020) consists of logical reasoning problems derived from the National Civil Servants Examination of China. The questions are designed to assess critical thinking and problem-solving abilities, requiring examinees to read a context passage and answer questions based on logical reasoning. In our experiments, we utilized English versions of 7,376 training samples and 651 test samples, including the context passages, questions, and answer options. The context passages were retained because they were integral to understanding and answering the questions.

F.4 OpenBookQA

`OpenBookQA` (Mihaylov et al., 2018) features elementary-level science questions requiring multi-step reasoning and common knowledge application. The dataset simulates an "open book" exam setting to assess understanding beyond simple fact retrieval. It includes a corpus of scientific facts alongside multiple-choice questions, each with four answer options. In our experiments, we used 4,957 training samples and 500 test samples while excluding the related facts to align with our goal of developing self-explanation capabilities without leveraging pre-existing explanatory content.

G Self-Alignment Details

G.1 SFT Training Details

For $\mathcal{M}_{\text{SFT}}$, we used the AdamW optimizer with a learning rate of 5×10^{-5} for one epoch, following a cosine schedule with 10% warmup steps. Gradient clipping was set to 0.3, and we used an effective batch size of 12. Loss was computed only on the assistant's completions. Instead of fine-tuning the entire model, we applied a LoRA adapter ($\alpha = 128$, dropout = 0.05, rank $r = 256$) to all linear layers. LoRA adapters were used to accelerate training and to act as a regularization method (Biderman et al., 2024), addressing the overfitting tendencies of DPO (Thakkar et al., 2024), which is applied during the later alignment phase.

G.2 Self-Preference Dataset

To ensure the integrity of our evaluation process, we constructed separate self-preference datasets for each benchmark. These datasets were created using input prompts specific to each task, ensuring that the DPO alignment data remained unaffected by cross-task contamination. This approach prevents potential result inflation, which could occur if models were aligned across diverse tasks—unlike SFT models, which are fine-tuned on a single classification task at a time.

For aligning $\mathcal{M}_{\text{Rank}}$ and $\mathcal{M}_{\text{Anchor}}$, we generated the self-preference dataset by sampling $N = 4$ responses from $\mathcal{M}_{\text{SFT}}$ for each input prompt (with settings: temperature $T = 0.6$ and top-k value of 0.9). The specific prompt used for this process is provided in Appendix O.

In cases where the responses were consistently incorrect, we employed $\mathcal{M}_{\text{Consultant}}$ to generate candidate explanations based on the correct answer y_i. The consultant model was configured with parameters $T = 0.5$ and top-k = 0.9. The corresponding prompt used for generating these explanations is detailed in Appendix N.

The responses were scored by $\mathcal{M}_{\text{Judge}}$, which was the same base model used in the alignment process, ensuring a self-contained procedure. This setup contrasts with the evaluation phase, where a more capable model, drawn from a different family of models, serves as the judge. The scoring methodology followed the approach described in Section 3.2, with $\mathcal{M}_{\text{Judge}}$ utilizing fixed inference

parameters ($T = 0$). The specific prompt used by $\mathcal{M}_{\text{Judge}}$ is detailed in Appendix M.

For $\mathcal{M}_{\text{Rank}}$, preference pairs were selected based on the assigned scores, with the highest-scoring explanation chosen as the winner, and the losing explanation randomly selected from the remaining candidates. For $\mathcal{M}_{\text{Anchor}}$, preference pairs were selected using the methodology described in Section 5.2.

G.3 DPO Training Details

For DPO-aligned models ($\mathcal{M}_{\text{Rank}}$, $\mathcal{M}_{\text{Anchor}}$), we used similar hyperparameters as in the SFT phase but reduced the learning rate to 5×10^{-7} and trained for 2.6k steps with an effective batch size of 6. The DPO process used a β value of 0.1 and updated the LoRA weights obtained during SFT.

G.4 Infrastructure

All LLMs used in this study were directly downloaded from Hugging Face. Regarding computational costs, each stage—including SFT, building the preference dataset, and DPO alignment—was executed on a single NVIDIA H100 or H200 GPU, completing within 24 hours.

H Analysis of Individual Evaluation Dimensions

Figure 2 presents the average scores for each evaluation criterion used to assess self-explanations, as described in Section 3.1, for all evaluated models across the benchmark datasets.

I Generated Instructions

Table 3 presents the distribution of categories—Consistently Correct (CC), Consistently Incorrect (CI), and Variable (V)—across the datasets used during the DPO alignment phase of $\mathcal{M}_{\text{Anchor}}$.

J Definition λ

We define λ as the proportion of the self-preference dataset used to align $\mathcal{M}_{\text{Anchor}}$, corresponding to preference pairs selected under the (CI) or (V) strategies:

$$\lambda = \frac{\text{CI} + \text{V}}{\text{CC} + \text{CI} + \text{V}} \tag{1}$$

Dataset	Category	Samples	Ratio
AQuA-Rat	V	1196	41.17
	CC	1010	34.77
	CI	699	24.06
ARC-Chg.	V	62	8.09
	CC	645	84.20
	CI	59	7.70
LogiQA	V	1251	26.86
	CC	2487	53.39
	CI	920	19.75
OpenbookQA	V	176	5.13
	CC	3178	92.60
	CI	78	2.27

Table 3: **Distribution of anchor categories**.

K Relative Gains: Measuring Self-Alignment Improvement

The effectiveness of self-alignment strategies in improving explanation quality and classification accuracy can be assessed by measuring the performance gains of $\mathcal{M}_{\text{Anchor}}$ relative to a ranking-based self-alignment approach, both evaluated against the baseline supervised fine-tuning ($\mathcal{M}_{\text{SFT}}$). This improvement is captured by the *Relative Gain (RG)* metric.

K.1 Individual Gains

Performance gains are first computed relative to the SFT baseline. Given a performance metric Metric$(\cdot)$, which can represent either accuracy or explanation quality, the individual gains for each approach are defined as:

$$G_{\text{Rank}} = \text{Metric}(\mathcal{M}_{\text{Rank}}) - \text{Metric}(\mathcal{M}_{\text{SFT}}) \tag{2}$$

$$G_{\text{Anchor}} = \text{Metric}(\mathcal{M}_{\text{Anchor}}) - \text{Metric}(\mathcal{M}_{\text{SFT}}) \tag{3}$$

K.2 Relative Gain (RG)

The *Relative Gain* quantifies the effectiveness of $\mathcal{M}_{\text{Anchor}}$ compared to $\mathcal{M}_{\text{Rank}}$:

$$RG = \frac{G_{\text{Anchor}}}{G_{\text{Rank}}} - 1 \tag{4}$$

This metric captures the additional improvement achieved by $\mathcal{M}_{\text{Anchor}}$ beyond what is obtained

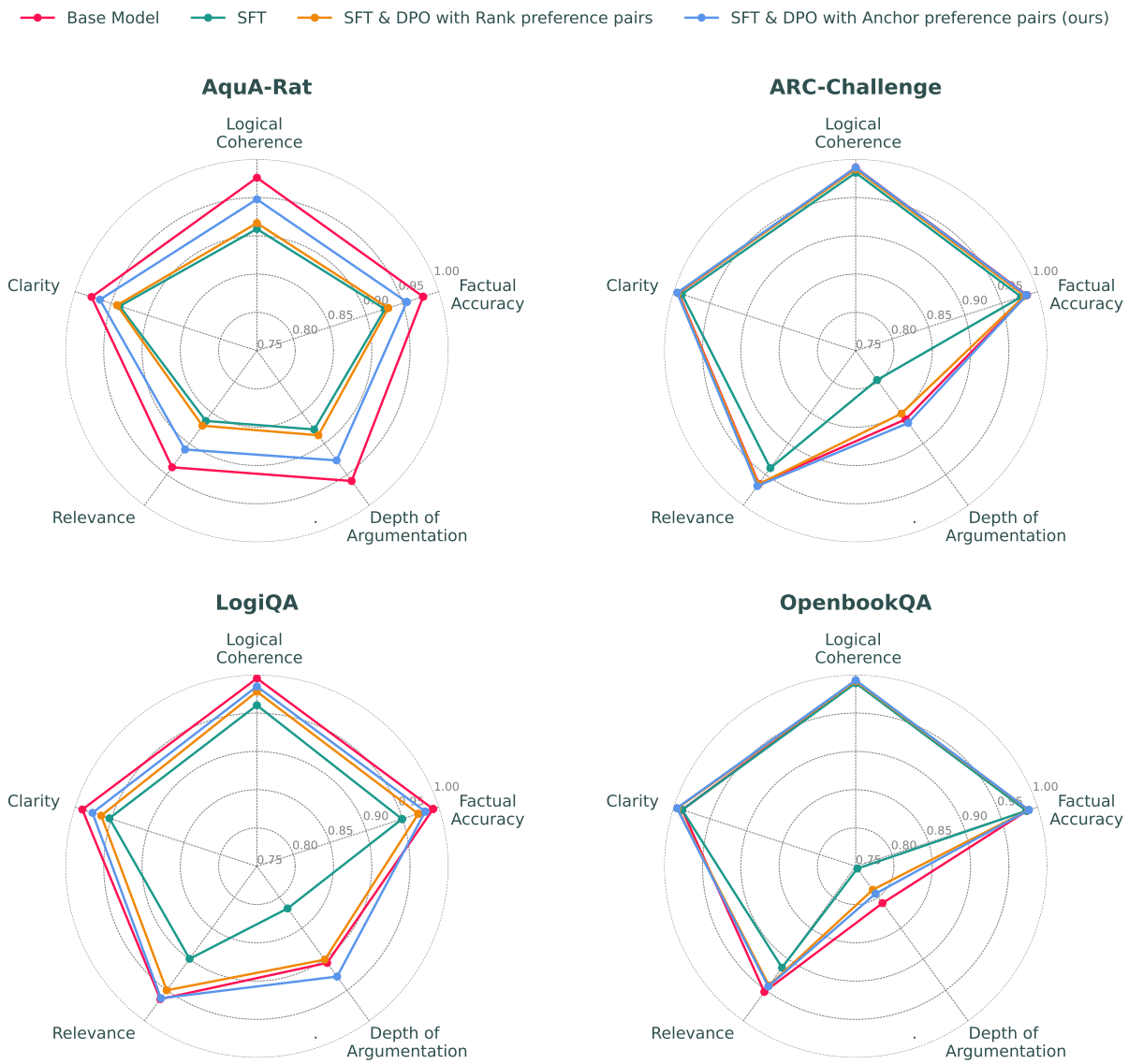

Figure 2: **Average Self-Explanation Scores per Evaluation Criterion**. Average scores for each evaluation criterion used to assess self-explanations, as described in Section 3.1.

through ranking-based self-alignment alone. A positive RG indicates that $\mathcal{M}_{\text{Anchor}}$ provides greater enhancement in explanation quality or accuracy compared to ranking-based methods, while an RG near zero suggests comparable performance.

L Correlation with Human Judgments

We conducted a **complementary** human evaluation to assess whether the LLM-as-a-Judge approach aligns with human raters across the evaluation criteria.

This evaluation provides insight into whether the criteria used by the LLM-based judge are effectively captured by the model. A lack of positive correlation with human ratings would indicate that a specific criterion is not well understood by the LLM, suggesting the need for modification or removal from the evaluation framework.

While this experiment serves as supporting evidence for the approach, **it is not a scalable method** for general evaluation.

For this study, we sampled 30 explanations generated by either an aligned model or a base model across 10 distinct questions from the `LogiQA` dataset, which covers a diverse range of logical reasoning problems across different domains.

To evaluate these explanations, we employed three state-of-the-art language models—`Mistral-Large-Instruct-2407` and `Qwen2.5-72B-Instruct`—as automated judges.

These models rated the explanations on a five-point scale: `Bad`, `Poor`, `Fair`, `Good`, and `Excellent`.

In addition, the same set of questions were included in a survey administered to multiple human raters. Each rater independently evaluated the quality of the explanations based on a specific criterion (e.g., Depth of Argumentation). To enhance diversity and reliability, each explanation was evaluated by at least three human raters, all of whom voluntarily participated in the study.

Overall, the participant pool consisted of a mix of graduate and undergraduate volunteers, with no compensation provided. Participants were informed about how their data would be used, and the experimental design received ethics approval. The demographic breakdown was 22% female and 78% male.

We computed the Spearman correlation between LLM ratings and human consensus ratings to evaluate the strength and direction of their monotonic relationship. Results are summarized in Table 4.

The analysis revealed moderate positive correlations for both judge models. `Mistral-Large-Instruct-2407` exhibited the strongest alignment with human judgments, consistently achieving correlation coefficients above 0.45 across all criteria ($p < 0.01$).

In contrast, `Qwen2.5-72B-Instruct` showed more variable performance. While it achieved significant correlations for Clarity ($\rho = 0.48$, $p < 0.01$) and Factual Accuracy ($\rho = 0.45$, $p < 0.02$), its correlation for Relevance was notably weaker and non-significant ($\rho = 0.28$, $p = 0.15$).

Figure 5 visualizes the relationship between LLM ratings and human consensus scores across five evaluation criteria. Each point represents the mean LLM rating for a given human consensus score, with error bars indicating standard deviation. The plots indicate a positive correlation with human judgments; however, sensitivity to specific criteria varies. While human raters utilize the full scoring range, LLMs—particularly for criteria like Clarity—tend to concentrate ratings within the mid-to-high range while still maintaining a positive correlation.

Notably, (a) `Mistral-Large-Instruct-2407` demonstrates the most consistent alignment with human judgments, showing a steadily increasing relationship compared to (b) `Qwen2.5-72B-Instruct`.

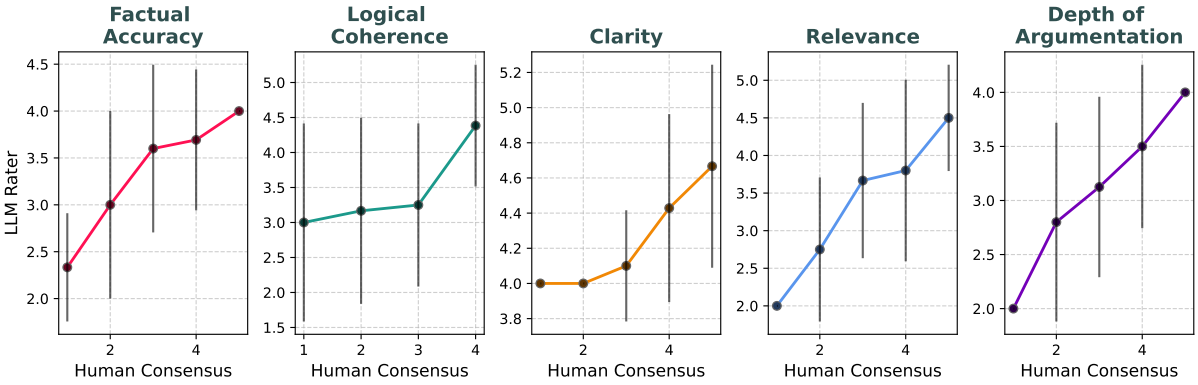

Figure 3: `Mistral-Large-2407`

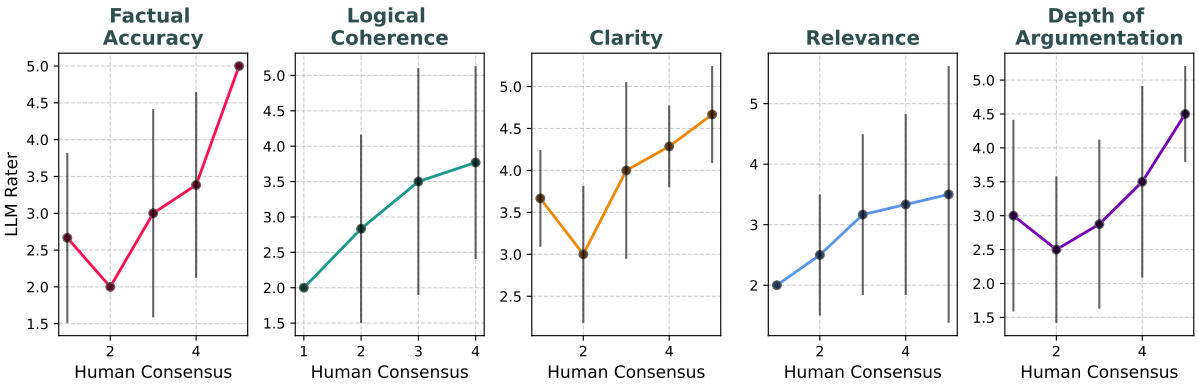

Figure 4: `Qwen2.5-72B-Instruct`

Figure 5: **LLM vs Human Consensus Ratings Across Evaluation Criteria:** Mean LLM ratings plotted against human consensus scores for each evaluation dimension with standard deviation error bars.

LLM-as-a-Judge	Criteria	Spearman Correlation	p-value
`Mistral-Large Instruct-2407`	Factual Accuracy	0.51	0.01*
	Logical Coherence	0.49	0.01*
	Clarity	0.52	0.01*
	Relevance	0.45	0.01*
	Depth of Argumentation	0.50	0.01*
`Qwen2.5 72B-Instruct`	Factual Accuracy	0.45	0.02*
	Logical Coherence	0.34	0.07
	Clarity	0.48	0.01*
	Relevance	0.28	0.15
	Depth of Argumentation	0.40	0.04*

Table 4: **Analysis of Agreement Between LLM and Human Evaluations:** A comparative analysis of agreement between LLM-based and human consensus ratings using Spearman Correlation (measuring ranking consistency). Results are broken down by evaluation criteria for each LLM judge, with p-values indicating statistical significance ($*p < 0.05$).

M Judge prompt

Table 5 presents the complete prompt that guides the judge model in evaluating the self-explanations.

Table 5: **Judge prompt**.

Prompt Judge

System: You are an expert judge tasked with evaluating the quality and correctness of a STATEMENT in response to a given QUESTION.

Your goal is to assess whether the STATEMENT accurately and comprehensively answers the QUESTION while providing sound reasoning and clear explanations. Be vigilant for any errors, misleading information, or gaps in the reasoning.

Evaluate the STATEMENT based on the following criteria:

- **Factual Accuracy:** Are all specific claims, data points, and facts in the STATEMENT verifiably correct?
- **Logical Coherence:** Does the STATEMENT follow a clear, logical sequence of reasoning without contradictions, and do all parts logically connect to form a cohesive and unified explanation?
- **Clarity and Comprehensibility:** Is the STATEMENT expressed clearly and precisely, using appropriate terminology and effectively explaining complex ideas?
- **Relevance and Completeness:** Does the STATEMENT fully address all relevant aspects of the QUESTION, ensuring that it directly responds to the specific context or requirements, leaving no key details omitted?
- **Depth of Argumentation:** Does the STATEMENT provide strong reasoning and credible evidence to support its conclusions?

For each criterion, select one of the following assessments:

- **EXCELLENT:** The STATEMENT exemplifies this criterion, setting a high standard.
- **GOOD:** The STATEMENT meets this criterion well, with only minor room for improvement.
- **FAIR:** The STATEMENT adequately meets this criterion but has noticeable areas for enhancement.
- **POOR:** The STATEMENT partially meets this criterion but requires significant revisions.
- **BAD:** The STATEMENT fails to meet this criterion.

Provide the assessments (EXCELLENT/GOOD/FAIR/POOR/BAD) for each criterion without additional justification. Respond with the following structure:

- **Factual Accuracy:** [Assessment]
- **Logical Coherence:** [Assessment]
- **Clarity:** [Assessment]
- **Relevance:** [Assessment]
- **Depth of Argumentation:** [Assessment]

User: Here is the question: **[QUESTION]**

Here is the statement you need to evaluate: **[STATEMENT]**

N Consultant prompt

Table 6: **Consultant prompt**.

Consultant prompt

System: You will act as an expert consultant. You will be shown a QUESTION and one OPTION to argue for. Your goal is to construct convincing arguments supporting your option as correct.

Maintain a scholarly yet assertive tone, focusing on evidence-based analysis while crafting your arguments and answering questions. Avoid phrases like "In summary", "In conclusion", or similar at the end of your argument. Be as concise as possible.

Your argument must fulfill the following criteria:

- **Truthfulness**: Any factual information mentioned in your argument must be true and accurate.
- **Logical Consistency**: Your argument must be logically sound, free of logical fallacies or contradictions.
- **Clarity**: Your argument directly addresses and answers the question. It is clear, concrete, concise, and well-structured.

User: Here is the question: **[QUESTION]**

Here is option you need to back up: **[OPTION]**

O Prompts inference per task

Table 7: **Prompt for LogiQA task**.

Prompt for LogiQA task

System: You will be presented with a CONTEXT passage and a corresponding QUESTION with four answer CHOICES. Carefully read the passage to understand its content. Then, read the QUESTION and CHOICES thoroughly. Choose the correct CHOICE and explain your reasoning.

Your response will consist of two parts: an EXPLANATION followed by your selected CHOICE.

Enclose your explanation within tags as follows:
<explanation>[Your EXPLANATION here]</explanation>

Enclose your chosen choice (e.g., if the question has only 4 choices, then A, B, C, or D) within tags as follows:
<choice>[Your CHOICE here]</choice>

User: Context: **[CONTEXT]**

Question: **[QUESTION]**

Choices: **[CHOICES]**

Table 8: **Prompt for AQuA-Rat task**.

Prompt for AQuA-Rat task

System: You will be given a QUESTION along with multiple answer CHOICES, involving a math problem that requires step-by-step reasoning to determine the correct answer. Carefully read the QUESTION and CHOICES. Choose the correct CHOICE and explain your reasoning.

Your response will consist of two parts: an EXPLANATION followed by your selected CHOICE.

Enclose your explanation within tags as follows:
<explanation>[Your EXPLANATION here]</explanation>

Enclose your chosen choice (e.g., if the question has only 4 choices, then A, B, C, or D) within tags as follows:
<choice>[Your CHOICE here]</choice>

User: Context: **[CONTEXT]**

Question: **[QUESTION]**

Choices: **[CHOICES]**

Table 9: **Prompt for ARC-Challenge task**.

Prompt for ARC-Challenge task

System: You will be presented a QUESTION with multiple answer CHOICES. Carefully read the QUESTION and CHOICES. Choose the correct CHOICE and explain your reasoning.

Your response will consist of two parts: an EXPLANATION followed by your selected CHOICE.

Enclose your explanation within tags as follows:
<explanation>[Your EXPLANATION here]</explanation>

Enclose your chosen choice (e.g., if the question has only 4 choices, then A, B, C, or D) within tags as follows:
<choice>[Your CHOICE here]</choice>

User: Context: **[CONTEXT]**

Question: **[QUESTION]**

Choices: **[CHOICES]**

Table 10: **Prompt for OpenbookQA task**.

Prompt for OpenbookQA task

System: You will be presented a QUESTION with multiple answer CHOICES. Carefully read the QUESTION and CHOICES. Choose the correct CHOICE and explain your reasoning.

Your response will consist of two parts: an EXPLANATION followed by your selected CHOICE.

Enclose your explanation within tags as follows:
<explanation>[Your EXPLANATION here]</explanation>

Enclose your chosen choice (e.g., if the question has only 4 choices, then A, B, C, or D) within tags as follows:
<choice>[Your CHOICE here]</choice>

User: Context: **[CONTEXT]**

Question: **[QUESTION]**

Choices: **[CHOICES]**