What Makes a Good Response? Learning Personal Preferences from Interpretable Features

Amirhossein Afsharrad[1,*], Emi Soroka[1,*], Daniel O'Neill[1], Sanjay Lall[1]
[1]Department of Electrical Engineering
Stanford University, Stanford, California
Email: {afsharrad, esoroka, dconeill, lall}@stanford.edu

***Abstract*—Aligning large language models (LLMs) with human preferences typically assumes a single, universal reward function learned from large-scale annotations. However, users have diverse and sometimes conflicting preferences, and personalizing to individual users requires methods that can learn reward functions from very few examples. We present a framework for interpretable reward modeling that extracts human-understandable features from LLM responses and learns lightweight reward functions over these features. Our approach decouples *what features matter* (transferable across users) from *how much each feature matters* (personalizable per user), enabling effective few-shot adaptation to new users. We analyze which features drive human preferences, finding that response detail and organization are consistently important. We further investigate when prediction fails, showing that model confidence is well-calibrated to task difficulty. Finally, we extend our framework to multi-turn conversations, discovering that users exhibit a "weakest-link" behavior where conversation quality is judged by the worst individual response. Our interpretable, data-efficient approach provides a foundation for personalizable LLM alignment.**

***Index Terms*—preference learning, LLM alignment, interpretability, personalization, few-shot learning**

I. Introduction

Reinforcement Learning from Human Feedback (RLHF) [1] has become the dominant paradigm for aligning LLMs with human preferences. Central to RLHF is the reward model: a learned function that scores responses according to human preferences. Current approaches assume a single, universal reward function trained on large-scale annotations. However, users have diverse and often conflicting preferences—what constitutes a "good" response varies across individuals. Personalizing reward models for individual users requires methods that can learn from very few examples, provide interpretable insights into what drives preferences, and quantify uncertainty about predictions.

Existing reward models require extensive training data and provide little insight into what features of a response drive the predicted reward, making it difficult to understand why one response would be preferred over another.

We propose decoupling *what features to measure* from *how to weight those features* and develop a novel approach using frontier LLMs to extract interpretable features from a small (potentially personalized) dataset. This allows the use of simple regression models to predict preferred responses from the feature set. The features ϕ capture universal quality dimensions that transfer across users; only the weights w need to be personalized. This decomposition enables learning personalized reward functions from minimal data.

We show that frontier LLMs are capable of designing feature sets as suites of simple natural-language questions suitable for evaluation by smaller language models, and that the resulting interpretable features can be used to train regression models that predict human preferences with $\geq 80\%$ accuracy. While modern reward models can achieve higher accuracy, our method provides full interpretability, with features described in natural language and assigned numeric weights. We provide code and prompts to replicate our results[1].

II. Related Works

A. Reward Modeling and RLHF

Reinforcement Learning from Human Feedback (RLHF) [1] has become the dominant paradigm for aligning LLMs with human preferences. Early reward-based approaches used Proximal Policy Optimization to fine-tune models against learned reward functions [2], [3], [4]. More recent work has sought to simplify this pipeline by eliminating the explicit reward model entirely. Direct Preference Optimization [5] optimizes directly from binary preference data. RRHF uses a ranking loss to align LLMs [6] and similarly, Preference Ranking Optimization [7] extend RLHF to learn from user rankings of k responses. Recently, Odds Ratio Preference Optimization even [8] merged the supervised fine-tuning and alignment steps. Other work has explored alignment via textual feedback rather than numeric rewards [9]. These methods have improved training efficiency, but share a common limitation: they assume a single, universal notion of preference learned from large-scale annotations, offering little insight into why one response is preferred over another. Recent work has begun addressing the limitations of binary preference data, through methods such as incorporating richer preference signals into reward modeling [10], [11].

B. Personalized Preference Learning

Traditional alignment has focused on training LLMs to be broadly helpful and harmless [12], assuming humans share

*Equal contribution.

[1]github.com/AmirAfsharrad/interpretable-preference-learning

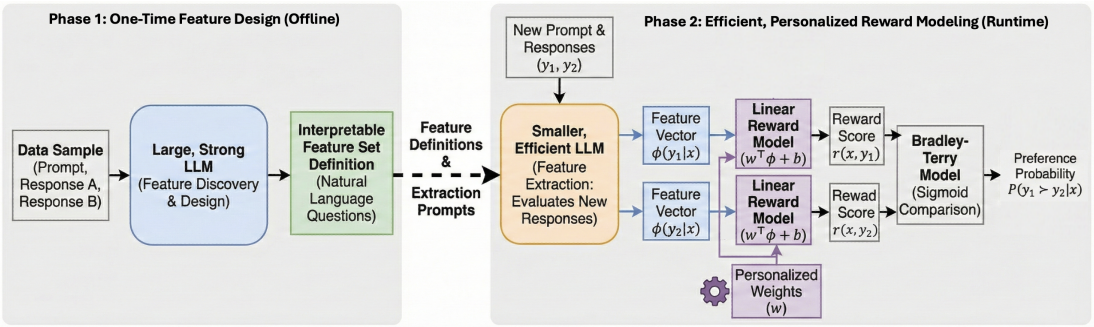

Fig. 1. Overview of our two-phase framework. In Phase 1 (offline), a large LLM analyzes sample data to discover interpretable quality dimensions and generates natural-language feature definitions. In Phase 2 (runtime), a smaller LLM evaluates responses against these features, producing feature vectors that feed into a linear reward model with personalized weights.

consistent utility functions–an assumption embedded in models like Bradley-Terry [13] that underlie most reward models. Yet human preferences are inherently variable [14]: users differ in how they value attributes such as helpfulness, brevity, accuracy, and tone [15]. Recognizing this, recent work has framed personalization as a key frontier for LLM alignment [16], [17], with new benchmarks emerging to measure both individual preferences [18], [19] and pluralistic alignment across populations [20], [21].

Personalization, however, involves two distinct challenges: *eliciting* user preferences and *aligning* models to them. On the alignment side, methods range from personalized RLHF [22] to test-time adaptation [23]. For elicitation, approaches include variational preference learning [24], which models latent user types; DRIFT [25], which tracks how preferences evolve over time; and soliciting preference descriptions from users [26]. Our work addresses the elicitation challenge through a different lens: rather than learning opaque representations of user preferences, we extract interpretable features that make personalization transparent and data-efficient.

C. Positioning Our Work

Our work is most similar to data-efficient personalization methods such as DRIFT [25], which focuses on time-varying preferences, and to text-based alignment systems [9], [26]. We focus on preference elicitation: designing a system that learns user preferences from small datasets with full interpretability.

III. Problem Formulation

We formalize the single-turn preference learning problem before extending it to multi-turn interactions.

A. Single-Turn Preferences

Consider a single-turn interaction consisting of a conversational context x (e.g., a user prompt) and an LLM response y. Our goal is to learn a reward function $r(x, y)$ that reflects a user's preferences over responses.

Following standard practice in RLHF, we learn from pairwise comparisons rather than absolute ratings, as humans provide more consistent judgments when comparing options directly [1]. Given a pair of responses (y_1, y_2) to the same context x, we model the probability that y_1 is preferred using the Bradley-Terry model

$$P(y_1 \succ y_2 \mid x) = \sigma(r(x, y_1) - r(x, y_2)), \tag{1}$$

where $\sigma(z) = 1/(1 + e^{-z})$ is the sigmoid function.

A key challenge is that users have heterogeneous preferences—what constitutes a "good" response varies across individuals. An effective personalized reward model must therefore: learn from limited user-specific data, provide interpretable insights, and generalize across diverse contexts.

B. Multi-Turn Extension

A multi-turn conversation $\mathcal{C} = \{(x_i, y_i)\}_{i=1}^{N}$ consists of a sequence of prompt-response pairs. The reward function $r(\mathcal{C})$ must now aggregate turn-level quality into an overall assessment: a new challenge as no data exists on how users aggregate their assessments of individual turns into holistic judgments on conversation quality. We discuss this challenge and extend our work to multi-turn evaluation in Section VIII.

IV. Method

Our approach decomposes reward modeling into two phases, illustrated in Figure 1: (1) an offline phase where a frontier LLM discovers interpretable features from sample data, and (2) a runtime phase where a smaller model extracts these features and a lightweight regression model predicts preferences. This decomposition decouples *what features matter* (transferable across users) from *how much each feature matters* (personalizable per user).

A. Phase 1: Feature Discovery

The first phase identifies interpretable quality dimensions that distinguish between candidate responses. Given a small sample of response pairs $\{(x^{(i)}, y_A^{(i)}, y_B^{(i)})\}$, we prompt a capable LLM (e.g., Claude, GPT-4) to analyze the pairs and propose features along which the responses differ. Crucially, we do not reveal which response was preferred; the goal is to discover features that discriminate between responses independently of a particular user's preferences. This allows the same feature set to support diverse preference profiles. We provide our prompt in Appendix A.

The LLM is instructed to produce features that are:

- **Objective:** Assessable with high agreement.
- **Interpretable:** Meaningful to humans.
- **Discriminative:** Likely to differ between $y_A^{(i)}$ and $y_B^{(i)}$.

The output of this phase is a feature schema consisting of natural-language questions (e.g., "Does the response provide specific examples?") along with prompts for extracting each feature. As a baseline, one could design such a schema without access to data, relying solely on intuitions about response quality. We compare against this baseline in our experiments and find that grounding feature discovery in actual response pairs yields more discriminative features.

B. Phase 2: Feature Extraction and Reward Modeling

At runtime, we use the feature schema from Phase 1 to evaluate new responses. Given a context x and response y, a smaller, efficient LLM answers each feature question, producing a feature vector $\phi(y \mid x) \in \mathbb{R}^d$. We adopt the notation $\phi(y \mid x)$ to emphasize that feature extraction depends on both the response and its context; this is not a conditional probability, but rather a deterministic mapping. This design enables efficient inference, since the feature definitions are fixed and extraction can be performed by models much smaller than those used for discovery.

1) Reward Function: We parameterize the reward function as a linear combination of features,

$$r(x, y) = w^\top \phi(y \mid x) + b, \tag{2}$$

where $w \in \mathbb{R}^d$ is a weight vector and b is a bias term. This parameterization provides direct interpretability through the weights w_i and enables personalization by learning user-specific weights while sharing the feature schema across users.

2) Learning from Pairwise Comparisons: We learn the weights w from pairwise preference data using the Bradley-Terry model (1). Given that response y_1 is preferred over y_2, the likelihood is

$$P(y_1 \succ y_2 \mid x) = \sigma\big(w^\top(\phi(y_1 \mid x) - \phi(y_2 \mid x))\big), \tag{3}$$

where $\sigma(z) = 1/(1 + e^{-z})$ is the sigmoid function. The bias term cancels in $r(x, y_1) - r(x, y_2)$, so the bias remains a free parameter that can be used (for example) to shift rewards to be positive, but plays no role in learning.

Let $\Delta\phi^{(i)} = \phi(y^{(i)}_{\text{chosen}} \mid x^{(i)}) - \phi(y^{(i)}_{\text{rejected}} \mid x^{(i)})$ denote the feature difference for pair i. We minimize the negative log-likelihood, $\mathcal{L}(w) = -\sum_{i=1}^{N} \log \sigma(w^\top \Delta\phi^{(i)})$. To enforce symmetry (swapping chosen and rejected should flip the prediction), we augment the training data by including both $(\Delta\phi^{(i)}, 1)$ and $(-\Delta\phi^{(i)}, 0)$ for each pair. This formulation reduces to **logistic regression** on feature differences.

V. Experiments: Single-Turn Preferences

We evaluate our feature-based reward modeling approach on single-turn preference prediction, showing that our approach is sample efficient, performs well on real data, and can be applied using small, efficient LLMs for feature extraction.

A. Experimental Setup

Datasets. We validate our approach on three open-source datasets, summarized in Table I: OpenAssistant's OASST-1 [27], a multi-turn dataset with human preferences given as response ratings in $[0, 1]$; a random sample of "easy" and "hard" interactions from RewardBench, a carefully curated dataset intended for learning reward functions [28]; and a random sample of Chatbot Arena data (LMArena) with human preferences on a binary scale [29]. For RewardBench, we use "chat-easy" and "chat-hard" pairs, which were curated from two other datasets: AlpacaEval [30] and MT-Bench [31]. We selected these datasets to include real conversations on diverse topics (OASST-1, LMArena), including multi-turn and single-turn interactions, and high-quality preference pairs (RewardBench).

Feature Design. We prompt Claude Code, a frontier LLM with agentic and long-context reasoning capabilities, as described in Section IV-A, providing XX randomly selected, unlabeled samples.

Feature Extraction. We extract features using GPT-4o-2024-08-06 [32] which supports structured output, allowing all features for a sample to be evaluated in one API call. We verify robustness by comparing extractions across different LLMs, testing a variety of sizes and model architectures. For LLMs that do not support structured output, we use simple text processing to extract the features.

Evaluation Metrics. We evaluate the performance of our approach across easy and challenging samples (which are labeled in the RewardBench and OASST1 datasets). For each dataset and feature schema, we train a logistic regression classifier and report accuracy and AUC-ROC on held-out test pairs. Finally, we perform PCA and t-SNE analysis on our feature sets, showing that human preferences can be represented in a low-dimensional space.

Baselines. To evaluate the effectiveness of LLM feature discovery, we compare our approach against a baseline feature set, generated by GPT-4o following the same prompt used to extract dataset-specific features, with references to samples removed. These features are reasonable, but are not tailored to each dataset. We also compare the performance of our feature extraction pipeline using different LLMs, finding that small open-source models can match the performance of large proprietary ones on this task.

Table II shows that on OASST-1, the dataset-specific feature schema exceeds the baseline performance. RewardBench has high performance across all feature sets, and LMArena is difficult to classify. This may be due to the high variance in LMArena data, where preferences are noisy and influenced by multiple content-dependent factors [33], as well as the limited number of samples used in the feature discovery phase.

B. Robustness Across Different Evaluators

Table III compares the performance of different models on evaluating the OASST-1 feature schema. While the evaluations can exhibit inconsistency, human preferences are also inconsistent [14]. Instead, we directly measure the predictive power

TABLE I

COMPARISON OF THE THREE DATASETS USED TO BENCHMARK OUR PAPER. WE RANDOMLY SELECT 200 SAMPLES FROM EACH, FILTERING OUT TIES, NON-ENGLISH RESPONSES, AND RESPONSES FLAGGED FOR SAFETY OR TOXICITY.

Dataset	Description	Multi-Turn	Rating Type
OASST-1	Multi-turn conversation trees, branching on two possible responses at each turn	✓	Per-turn quality score in $[0, 1]$
RewardBench	Easy and hard preference pairs	✗	Binary preference
LMArena	Chat data from users in the wild; users select which response they prefer	✗	Binary preference

TABLE II

TEST SET ACCURACY AND AUC-ROC ACROSS DATASETS AND FEATURE SCHEMAS, EVALUATED WITH GPT-4O-2024-08-06. CONFIDENCE INTERVALS GIVEN AT 1σ.

Schema \ Data	OASST-1	RewardBench	LMArena
Baseline Accuracy	$83 \pm 4.7\%$	$95 \pm 3.4\%$	$67 \pm 6.2\%$
OASST-1 Accuracy	$85 \pm 3.5\%$	$98 \pm 2\%$	$60 \pm 5.8\%$
RewardBench Accuracy	$80 \pm 7.4\%$	$97 \pm 1.6\%$	$61 \pm 10.4\%$
LMArena Accuracy	$78 \pm 7.8\%$	$93 \pm 3.4\%$	$64 \pm 8.4\%$
Baseline AUC-ROC	$0.94 \pm 1.6\%$	$0.99 \pm 0.9\%$	$0.75 \pm 7.2\%$
OASST-1 AUC-ROC	$0.95 \pm 2.9\%$	$1.00 \pm 0.5\%$	$0.67 \pm 6.0\%$
RewardBench AUC-ROC	$0.92 \pm 3.1\%$	$0.98 \pm 1.9\%$	$0.66 \pm 12\%$
LMArena AUC-ROC	$0.91 \pm 4.9\%$	$0.98 \pm 1.6\%$	$0.71 \pm 9.6\%$

TABLE III

PREDICTION PERFORMANCE ON OASST-1 DATA USING OASST-1 FEATURES EXTRACTED BY DIFFERENT MODELS.

Evaluator	Acc.	Acc. p-val	AUC-ROC
GPT-4o [32]	$83 \pm 6.0\%$		$0.95 \pm 2.4\%$
GPT-4o-mini [34]	$85 \pm 5.9\%$	0.128	$0.95 \pm 1.8\%$
Claude-4.5-sonnet [35]	$85 \pm 4.9\%$	0.194	$0.94 \pm 3.6\%$
Claude-3-haiku [35]	$80 \pm 7.9\%$	0.404	$0.87 \pm 4.8\%$
LLaMA-3-70B [36]	$79 \pm 10.5\%$	1.15×10^{-2}	$0.94 \pm 4.0\%$
LLaMA-4.1-Scout.	$85 \pm 5.4\%$	0.604	$0.94 \pm 4.0\%$
LLaMA-3.1-8B [36]	$73 \pm 6.0\%$	1.35×10^{-2}	$0.83 \pm 8.3\%$

of features extracted by small LLMs [2].We find that Llama-4.1-Scout-17B-12E-Instruct [37] , an open-source mixture-of-experts (MoE) model with 17 billion parameters and 12 experts, matches or exceeds the performance of large commercial models. Using a two-sided t-test to compare accuracy scores, we compute p-values for the null hypothesis that each evaluator's performance is indistinguishable from GPT-4o's and find that we can only reject the null hypothesis for LLaMA-3-70B and LLaMA-3.1-8B.

C. What Drives Human Preferences?

Key findings: Level of detail, organization, and helpful tone are consistently the strongest predictors of human preference. We report top-10 features and their corresponding weights for each dataset in Appendix C. We also observe the effects of individual users' diversity: in Table II, we achieve the highest performance on the relatively clean RewardBench data and the lowest on LMArena, an unfiltered chat dataset.

[2]While GPT-4o-mini and Claude-3-haiku are advertised as small models, parameter counts for these proprietary models are unknown.

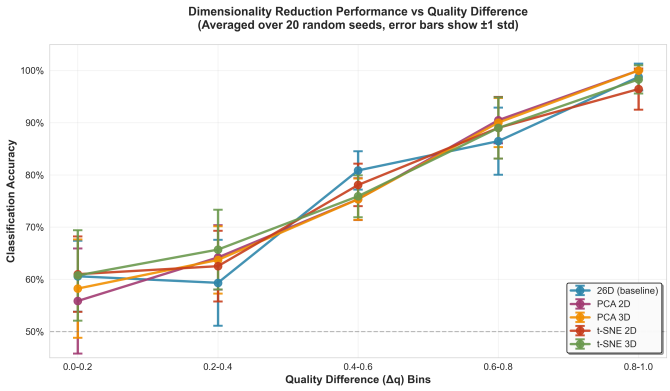

Fig. 2. Classification accuracy versus quality difference bins for different dimensionality reduction methods. All methods show clear monotonic improvement as Δq increases. Error bars represent ± 1 standard deviation across 20 random seeds. The gray dashed line indicates chance performance (50%).

VI. PREDICTION DIFFICULTY AND MODEL CALIBRATION

OASST-1 contains fractional quality scores $q \in [0, 1]$ where the higher-scoring response is preferred. These scores were assigned by humans based on their holistic evaluation of each turn; users are instructed to "Judge quality based on how well the reply adheres to the guidelines. Factual accuracy and helpfulness are first and foremost." [27]. A natural question is whether our model performs better on pairs with larger quality differences. We investigate this by stratifying test performance according to the quality gap between paired responses.

We partition test pairs into five bins based on quality difference Δq: [0.0–0.2], [0.2–0.4], [0.4–0.6], [0.6–0.8], and [0.8–1.0]. Figure 2 shows classification accuracy across these bins for the full 26-dimensional feature space as well as several dimensionality-reduced variants. The results reveal a clear monotonic relationship across all methods: accuracy increases from approximately 60% on the hardest pairs ($\Delta q < 0.2$) to nearly 100% on the easiest pairs ($\Delta q > 0.8$). This pattern validates that the features capture meaningful quality gradients aligned with human judgments.

Crucially, model confidence tracks this pattern: confidence increases from 0.685 on small-gap pairs to 0.995 on large-gap pairs. This calibration is valuable for deployment, as the model exhibits appropriate uncertainty on ambiguous cases, providing a mechanism to flag predictions that may require human review.

TABLE IV
CLASSIFICATION ACCURACY FOR DIFFERENT DIMENSIONALITY REDUCTION METHODS. CONFIDENCE INTERVALS ARE AT 1σ.

Method	Accuracy	Retention
26D (baseline)	75.9% ± 3.6%	100%
t-SNE 2D	75.8% ± 3.1%	99.9%
t-SNE 3D	75.8% ± 3.0%	99.9%
PCA 3D	75.1% ± 2.7%	98.9%
PCA 2D	74.8% ± 3.1%	98.5%

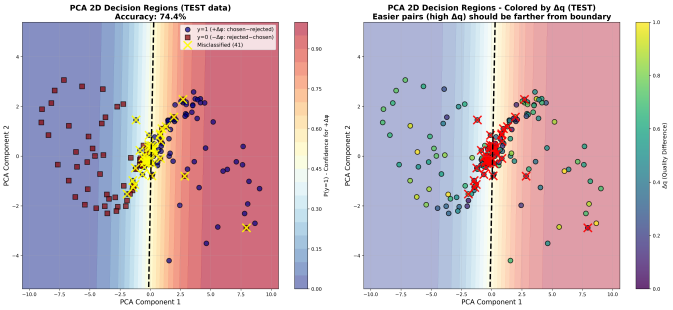

Fig. 3. PCA 2D decision regions. Left: Background shows $P(y = 1)$ as a color gradient, with the dashed line marking the decision boundary. Points are colored by label direction. Right: Same regions with points colored by quality difference Δq, revealing that easier pairs (yellow) lie farther from the boundary.

A. Low-Dimensional Structure.

Figure 2 also reveals a striking finding: low-dimensional projections retain nearly all predictive power. The t-SNE 2D and 3D curves nearly overlap with the 26D baseline across all difficulty levels, achieving 75.8% overall accuracy compared to 75.9% for the full feature space. Even PCA 2D retains 98.5% of baseline performance. Table IV summarizes these results.

This compression works because the feature space is intrinsically low-dimensional. PCA reveals that the first principal component (PC1) accounts for 45% of total variance and aligns with a "quality gradient" heavily weighted toward features like level of detail and organization. The learned decision boundary is almost entirely determined by PC1, meaning classification reduces to thresholding on one axis.

Figure 3 visualizes the PCA 2D decision regions. Points with large quality differences (shown in yellow) cluster far from the decision boundary, explaining the perfect accuracy on these cases. Points with small quality differences (shown in purple) cluster near the boundary, where the model's predictions are uncertain. The mean Euclidean separation between paired difference vectors scales from 2.2 units for hard pairs to 9.3 units for easy pairs, a 4.3× increase that directly explains the accuracy gap.

These findings confirm that hard pairs are **fundamentally challenging** due to feature limitations, not model capacity. The current features do not capture the subtle distinctions humans perceive in near-ties, regardless of whether we use 2D or 26D representations. Notably, for pairs with $\Delta q > 0.8$, PCA 2D achieves 100% accuracy, proving that responses with large quality differences are perfectly linearly separable even after projection to two dimensions.

TABLE V
LABEL DISAGREEMENT RATES BETWEEN PERSONAS AND POPULATION BASELINE. DISAGREEMENT IS THE PERCENTAGE OF EXAMPLES WHERE THE PERSONA PREFERS A DIFFERENT RESPONSE THAN THE POPULATION.

Persona	Depth-seeker	Structure-lover	Minimalist
Disagreement Rate	30.0%	31.5%	55.0%

VII. PERSONALIZATION EXPERIMENTS

Having shown that our model can predict population-level preferences, we now investigate whether our framework can be personalized to individual users with divergent preferences. Since no datasets of real per-user preference annotations exist at sufficient scale, we conduct a controlled proof-of-concept study using synthetic personas. We emphasize that these experiments are intended as an ablation: they stress-test whether the framework *can* accommodate strongly divergent preference profiles, not as a claim that synthetic personas faithfully represent the full complexity of real user diversity.

A. Experimental Design

We construct three synthetic user personas with distinct preference profiles and use them to re-label the same 400 response pairs from OASST-1. This design enables direct comparison: all personas evaluate identical pairs, but assign different preference judgments based on specific criteria.

We define three personas representing different user types:

- **Depth-seeker**: Prefers comprehensive, detailed responses with thorough explanations.
- **Minimalist**: Prefers concise, direct responses; penalizes verbosity.
- **Structure-lover**: Prioritizes well-organized responses with clear sections and logical flow.

Using GPT-4o with structured outputs, we generate persona-specific quality scores by providing the persona description and asking the model to evaluate responses from that perspective. To mitigate the risk of synthetic label noise, all generated labels were reviewed by human annotators; samples whose chain-of-thought reasoning neither reflected the persona's stated priorities nor resembled plausible human judgment were discarded. We then train separate pairwise logistic regression models for each persona using 80/20 train/test splits, repeating all experiments with 20 random seeds.

B. Label Disagreement

Table V shows how often each persona's preferences differ from the population baseline. While depth-seeker and structure-lover disagree with population preferences on approximately 30% of examples, the minimalist persona shows 55% disagreement—preferences are nearly inverted compared to the population. This suggests that minimalist represents an outlier user type with fundamentally different quality criteria.

TABLE VI
CROSS-PERSONA EVALUATION ACCURACY MATRIX (MEAN ± STD OVER 20 RANDOM SEEDS). ROWS INDICATE TRAINING DATA; COLUMNS INDICATE TEST DATA. DIAGONAL ELEMENTS (BOLD) REPRESENT PERSONA-SPECIFIC MODELS. THE BOTTOM ROW SHOWS PERSONALIZATION GAIN OVER THE POPULATION MODEL. CONFIDENCE INTERVALS ARE COMPUTED AT 1 STANDARD DEVIATION.

Train → Test	Population	Depth-seeker
Population	74.6 ± 4.8%	79.8 ± 3.3%
Depth-seeker	79.2 ± 4.2%	84.7 ± 3.2%
Structure-lover	76.0 ± 4.2%	81.0 ± 3.4%
Minimalist	42.3 ± 4.1%	41.7 ± 4.4%
Personalization Gain	—	+4.9 pp
Train → Test	**Structure-lover**	**Minimalist**
Population	79.6 ± 4.1%	45.5 ± 5.5%
Depth-seeker	82.8 ± 4.1%	46.0 ± 3.9%
Structure-lover	85.1 ± 4.2%	48.8 ± 4.2%
Minimalist	44.0 ± 4.4%	67.5 ± 5.0%
Personalization Gain	+5.5 pp	+22.1 pp

C. Cross-Persona Evaluation

To quantify the value of personalization, we evaluate all four trained models (three personas plus population) on all four test sets (Table VI).

We draw two conclusions from this controlled setting. First, **modeling a heterogeneous population is inherently harder than modeling a consistent individual.** Under our synthetic conditions, the population model achieves only 74.6% test accuracy, while persona-specific models achieve 84.7% (depth-seeker) and 85.1% (structure-lover). Intuitively, population-level data aggregates diverse preferences, creating noise and inconsistency, while a single user's preferences are more internally consistent. This is a pattern we would expect to hold for real users as well.

Second, **population models fail on outlier users.** For example, our population model achieves only 45.5% accuracy on minimalist test data, as the minimalist user's preference for brevity is anti-correlated with the population-level preference for detail. While the majority of real users are unlikely to be as extreme or internally consistent as our synthetic personas, this result illustrates a structural failure mode: personalization most strongly benefits users whose preferences diverge from the population mean, and a universal model may actively harm such users by inverting their preferences.

D. Feature Weight Analysis

The learned feature weights reveal the source of these differences. Depth-seeker strongly rewards level of detail (+1.99) and penalizes brevity (-0.77), while minimalist exhibits the opposite pattern: it penalizes level of detail (-0.42) and depth (-0.53) while rewarding brevity (+0.38). Structure-lover, as expected, places the highest weight on organization (+1.78) and coherence (+1.11). These opposing coefficients explain why no single linear model can satisfy both depth-seeker and minimalist: the features that one persona rewards are precisely those that another penalizes.

The interpretable nature of our approach makes personalization transparent. Users can inspect the learned weights to understand how their preferences differ from the population, and developers can identify which user types require specialized models based on the geometry of their feature weights. Systems can also detect outlier users by monitoring disagreement with population-model predictions: users who consistently reject recommendations (like minimalist, whose preferences anti-correlate with population) signal the need for personalized modeling.

VIII. EXTENSION TO MULTI-TURN CONVERSATIONS

While the preceding experiments focus on single-turn interactions, many real-world applications involve multi-turn conversations where quality must be assessed across an entire dialogue. Extending our framework to this setting reveals an important gap in existing preference datasets and introduces a new dimension of personalization.

A. The Aggregation Problem

A fundamental limitation of current multi-turn preference datasets is that they provide only turn-level quality scores, not holistic conversation ratings. This creates a modeling challenge: we observe how users rate individual responses, but not how they combine these judgments into an overall assessment of a conversation. Yet this aggregation process is itself a source of individual variation. Some users may judge a conversation by its best moments, others by its worst, and still others by the final impression. This represents a distinct layer of personalization beyond feature-level preferences: even users who agree on turn-level quality may disagree on conversation-level preferences due to different mental aggregation strategies. Related work has approached multi-turn evaluation by hierarchically decomposing conversations and learning bias corrections from human judgments [38]; our work is complementary, focusing on preference prediction and flexible aggregation strategies rather than evaluation calibration.

We address this gap by experimenting with multiple aggregation hypotheses, demonstrating that our framework can accommodate different assumptions about user behavior. We emphasize that the three strategies we evaluate are not exhaustive; they represent plausible hypotheses that illustrate the flexibility of our approach.

B. Problem Setting

In multi-turn conversation path comparison, we are given a common conversation prefix and two alternative paths, each containing 2–5 assistant responses. The goal is to predict which conversation path a user would prefer. We extract 200 conversation path pairs from the OASST1 dataset, where each pair consists of two conversation trajectories diverging from a common prefix.

C. Feature Representation

Multi-turn conversations require handling variable-length sequences. For each turn i, we extract the same single-turn features $\phi(x_i, y_i) \in \mathbb{R}^{26}$ used in previous experiments.

TABLE VII
MULTI-TURN CONVERSATION PATH COMPARISON RESULTS (200 PAIRS, 80/20 SPLIT).

Features	Quality	Train Acc	Test Acc	AUC
Mean	Mean	74.1%	65.8%	0.737
Mean	Last	72.5%	66.7%	0.731
Mean	Min	74.9%	71.2%	0.817
Pad	Mean	79.9%	73.4%	0.813
Pad	Last	85.0%	65.2%	0.759
Pad	Min	87.3%	75.3%	0.858

Additionally, we compute multi-turn features that capture cross-turn dynamics such as consistency and topic coherence. The resulting representation for a conversation of N turns is inherently variable-length, which we address through two aggregation strategies:

- **Mean pooling**: Average features across all turns, producing a fixed-dimensional vector, discarding temporal ordering and positional information.
- **Zero-padding**: Pad shorter conversations to the maximum length (5 turns) with zeros, preserving temporal structure and positional information.

D. Quality Aggregation Hypotheses

Since ground-truth conversation-level ratings are unavailable, we construct labels by aggregating turn-level quality scores under different hypotheses about user behavior:

- **Mean quality**: Users judge by the average quality across all turns, weighing each interaction equally.
- **Last quality**: Users judge primarily by the final response, reflecting a recency bias.
- **Min quality**: Users judge by the worst turn, reflecting a "weakest-link" model.

These represent three plausible strategies, but others are possible. For instance, users might weight early turns more heavily (primacy bias), apply exponential discounting, or use complex nonlinear combinations. Our framework can accommodate any hypothesis by changing how labels are aggregated.

1) Results: Table VII presents results for all six combinations of feature aggregation and quality aggregation. Several findings emerge from this analysis.

a) Zero-padding outperforms mean pooling.: Across all quality aggregation hypotheses, zero-padding achieves higher test accuracy and AUC than mean pooling. This demonstrates that preserving temporal structure and positional information is valuable for multi-turn preference prediction: when in a conversation something happens matters, not just what happens.

b) Our method accommodates different aggregation hypotheses.: The framework achieves reasonable performance across all three aggregation strategies, with test AUC ranging from 0.731 to 0.858. This flexibility is important because different users may aggregate turn-level quality differently, and practitioners may not know in advance which strategy best matches their user population. The variation in accuracy across hypotheses (e.g., 0.858 AUC for min-quality versus 0.759 for last-quality with zero-padding) reflects how well our current feature set aligns with each aggregation assumption, not a claim about which aggregation strategy is more natural or correct. In practice, the appropriate aggregation hypothesis depends on the target user population and application context.

E. Implications

These results highlight that multi-turn preference prediction involves two distinct modeling choices: how to represent variable-length feature sequences, and how to hypothesize user aggregation behavior. Our framework handles both flexibly, achieving reasonable accuracy under different assumptions.

The absence of ground-truth conversation-level ratings in existing datasets is a significant limitation. For practitioners building multi-turn reward models, we recommend collecting conversation-level feedback directly whenever possible, rather than relying on turn-level scores with assumed aggregation. Direct conversation-level ratings would eliminate the need to hypothesize aggregation behavior and provide more training signals. Understanding how different users aggregate turn-level quality—and whether this varies systematically across user types—remains an important direction for future research and would enable richer models of multi-turn personalization.

IX. CONCLUSION

We presented a framework for interpretable, few-shot preference learning using LLM-extracted features. Our approach achieves strong performance with minimal data by decoupling universal quality features from personalizable preference weights. We demonstrated effective adaptation to diverse user personas in both single-turn and multi-turn settings, and showed that model confidence is well-calibrated to prediction difficulty. Further, we show that multi-turn preferences follow a "weakest-link" pattern, providing actionable insight for conversational AI systems. Our work provides a foundation for practical, personalizable LLM alignment.

APPENDIX A
CLAUDE CODE FEATURE GENERATION PROMPT

```
# Feature Design Prompt for LLM Preference
Prediction
## Task
I'm working on predicting which of two LLM
responses a human would prefer. I need you to
design a set of features that can be extracted
from (context, response) pairs to train a
pairwise preference model.
## Dataset
I have preference pairs where:
- Each example has a conversational context
(may be single-turn or multi-turn)
- Two candidate responses: one preferred by
humans, one not preferred
- Quality scores for each response (0 to 1)
Below, I'll provide sample data showing
preferred vs non-preferred responses.
## Requirements
Design ~20-30 features that are:
1. **Objective**: Observable characteristics,
not open to interpretation. Same response
should always yield same feature values.
2. **Simple**: Extractable by weaker-than-
```

```
frontier models. No deep reasoning or
specialized knowledge required. Clear,
unambiguous definitions.
3. **Discriminative**: Good at distinguishing
preferred from non-preferred responses. Look
for patterns in the sample data that
consistently differ.
## What I Need
1. **Examine the sample data** and identify
patterns between preferred vs non-preferred
responses
2. **Define features**: For each feature
provide:
- Feature name (snake_case)
- Type (bool, int with range 1-5, or
categorical)
- Clear description
3. **Group into categories** (e.g., structure,
content quality, tone, organization, etc.)
4. **Write the extraction prompt**: The
complete prompt that will be given to an LLM
API to extract these features from a (context,
response) pair
## Output Format
### Feature Definitions
**Category 1: [Name]**
- `feature_name` (type): [description]
- ...
**Category 2: [Name]**
- ...
### Extraction Prompt
```
[The complete prompt that will be used to
extract features via LLM API]
```
## Sample Data
[paste examples here]
```

For the baseline schema, we deleted the line `Below, I'll provide...` and the `Sample Data` section.

Appendix B
LLM Inference Details

Our experiments used the following models.

- GPT-4o-2024-08-06 via OpenAI API
- GPT-4o-mini-2024-07-18 via OpenAI API
- Claude-4.5-sonnet via the Stanford AI Playground API
- Claude-3-haiku via the Stanford AI Playground API
- LLaMA-3-70B via the Stanford AI Playground API
- meta-llama/Llama-4-Scout-17B-16E-Instruct, weights from Hugggingface. Deployed on Marlowe [39].
- meta-llama/Llama-3.1-8B-Instruct, weights from Huggingface. Deployed on Marlowe.

Appendix C
Feature Definitions

See Tables VIII, IX, and X.

Appendix D
Persona Definitions

These definitions were used to generate personalized quality scores for OASST-1 data by prompting GPT-4o to evaluate each conversation as a specific persona.

Definition for **Depth-seeker** persona.

TABLE VIII
TOP 10 FEATURES ON OASST-1 DATA USING THE OASST-1 FEATURE SCHEMA (TOP) AND BASELINE SCHEMA (BOTTOM).

OASST-1 Feature	w_i
is_very_short	-1.5399
technical_appropriate	+1.4544
is_polite	+1.1486
has_caveats_when_appropriate	+1.1007
has_numbered_list	+1.0862
has_bulleted_list	+1.0437
is_rude_or_dismissive	-0.7664
has_examples	+0.7399
has_conclusion	+0.6516
paragraph_count	+0.6479
Baseline Feature	w_i
response_length	+1.5926
information_completeness	+1.2065
fact_check	+1.1061
use_of_personal_pronouns	-0.8904
sentence_length_variability	+0.8316
relevance_to_context	+0.8240
question_usage	-0.7719
use_of_stories_or_examples	-0.7293
uniqueness_of_response	+0.7113
spelling	-0.7090

TABLE IX
TOP 10 FEATURES ON REWARDBENCH DATA USING THE REWARDBENCH FEATURE SCHEMA (TOP) AND BASELINE SCHEMA (BOTTOM).

RewardBench Feature	w_i
uses_appropriate_formatting	+1.4832
is_engaging_to_read	+1.2897
is_appropriately_confident	-1.2441
has_caveats_when_appropriate	+1.1533
level_of_detail	+1.0249
provides_sufficient_depth	+1.0182
is_complete	-1.0098
has_vivid_details	+0.8855
has_logical_flow	+0.8562
addresses_all_parts	+0.7935
Baseline Feature	w_i
information_completeness	+2.6637
num_sentences	+1.0163
quality_score	+0.9248
relevance_to_context	+0.8977
readability	+0.8377
empathy	+0.8283
formality	+0.7704
list_usage	+0.7269
num_paragraphs	+0.7054
grammar_correctness	-0.6954

```
"name": "The Depth Seeker",
"description": "Values comprehensive, detailed
↪ explanations. Wants to deeply understand the
↪ topic.",
"priorities": [
    "High level of detail and depth",
    "Concrete examples and specifics",
    "Well-organized presentation of complex
↪ information",
    "Thorough coverage of the topic",
    "Clear explanations with evidence" ],
"penalties": [
```

TABLE X
TOP 10 FEATURES ON LMARENA DATA USING THE LMARENA FEATURE SCHEMA (TOP) AND BASELINE SCHEMA (BOTTOM).

LMArena Feature	w_i
avoids_hallucinations	+1.6613
addresses_underlying_need	+1.5715
respects_ethical_boundaries	-0.9725
is_factually_accurate	-0.9289
is_substantive_not_filler	-0.9118
handles_sensitive_topics_appropriately	+0.7627
has_logical_flow	-0.6629
uses_vivid_engaging_language	+0.6502
uses_appropriate_formatting	+0.6165
is_logically_coherent	+0.5199
Baseline Feature	w_i
punctuation_appropriateness	+0.6419
relevance_to_context	+0.5787
question_usage	+0.5565
num_paragraphs	+0.5112
list_usage	+0.4873
use_of_slang	+0.3869
creativity	+0.3708
information_completeness	+0.3708
politeness	+0.3420
uniqueness_of_response	-0.3346

```
  "Superficial or brief answers",
  "Lack of examples or concrete details",
  "Missing important aspects of the topic",
  "Vague or hand-wavy explanations" ]
```

Definition for **Structure-lover** persona.

```
"name": "The Structure Lover",
"description": "Values clear organization and
↪ presentation. Believes a well-structured
↪ answer is a good answer.",
"priorities": [
  "Clear, logical organization",
  "Well-defined sections and structure",
  "Good use of transition words",
  "Proper introduction and conclusion",
  "Use of lists or formatting when appropriate"
↪ ],
"penalties": [
  "Disorganized or rambling responses",
  "Lack of clear structure",
  "Poor flow between ideas",
  "Missing organizational elements" ]
```

Definition for **Minimalist** persona.

```
"name": "The Minimalist",
"description": "Values brevity and directness.
↪ Prefers concise answers without unnecessary
↪ elaboration.",
"priorities": [
  "Short, direct responses",
  "Gets to the point quickly",
  "No long introductions or conclusions",
  "Few paragraphs preferred",
  "Addresses the question concisely" ],
"penalties": [
  "Long-winded explanations",
  "Unnecessary detail",
  "Lengthy introductions or conclusions",
  "Many paragraphs when fewer would suffice" ]
```

REFERENCES

[1] P. F. Christiano, J. Leike, T. Brown, M. Martic, S. Legg, and D. Amodei, "Deep reinforcement learning from human preferences," *Advances in neural information processing systems*, vol. 30, 2017.

[2] L. Ouyang et al., "Training language models to follow instructions with human feedback," *Advances in neural information processing systems*, vol. 35, pp. 27 730–27 744, 2022.

[3] D. M. Ziegler et al., "Fine-tuning language models from human preferences," *arXiv preprint arXiv:1909.08593*, 2019.

[4] R. Zheng et al., *Secrets of rlhf in large language models part i: Ppo*, 2023. arXiv: 2307.04964 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2307.04964.

[5] R. Rafailov, A. Sharma, E. Mitchell, C. D. Manning, S. Ermon, and C. Finn, "Direct preference optimization: Your language model is secretly a reward model," in *Advances in Neural Information Processing Systems*, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, Eds., vol. 36, Curran Associates, Inc., 2023, pp. 53 728–53 741.

[6] H. Yuan, Z. Yuan, C. Tan, W. Wang, S. Huang, and F. Huang, "Rrhf: Rank responses to align language models with human feedback," *Advances in Neural Information Processing Systems*, vol. 36, pp. 10 935–10 950, 2023.

[7] F. Song et al., "Preference ranking optimization for human alignment," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, 2024, pp. 18 990–18 998.

[8] J. Hong, N. Lee, and J. Thorne, *Orpo: Monolithic preference optimization without reference model*, 2024. arXiv: 2403.07691 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2403.07691.

[9] S. A. Lloret, S. Dhuliawala, K. Murugesan, and M. Sachan, *Towards aligning language models with textual feedback*, 2025. arXiv: 2407.16970 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2407.16970.

[10] Z. Wang et al., "HelpSteer2-Preference: Complementing ratings with preferences," in *The Thirteenth International Conference on Learning Representations*, 2025. [Online]. Available: https://arxiv.org/abs/2410.01257.

[11] A. Afsharrad, R. Zhou, L. Viano, S. Lall, and M. Ghavamzadeh, "Beyond binary preferences: A principled framework for reward modeling with ordinal feedback," in *The Fourteenth International Conference on Learning Representations*, 2026. [Online]. Available: https://openreview.net/forum?id=mteZOi0xyu.

[12] Y. Bai et al., *Training a helpful and harmless assistant with reinforcement learning from human feedback*, 2022. arXiv: 2204.05862 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2204.05862.

[13] R. A. Bradley and M. E. Terry, "Rank analysis of incomplete block designs: I. the method of paired comparisons," *Biometrika*, vol. 39, no. 3/4, pp. 324–345,

1952, ISSN: 00063444, 14643510. Accessed: Jan. 19, 2026.

[14] G. M. Becker, M. H. DeGroot, and J. Marschak, "Stochastic models of choice behavior," *Behavioral science*, vol. 8, no. 1, pp. 41–55, 1963.

[15] H. Wang et al., *Arithmetic control of llms for diverse user preferences: Directional preference alignment with multi-objective rewards*, 2024. arXiv: 2402.18571 [cs.LG].

[16] Z. Zhang et al., "Personalization of large language models: A survey," *arXiv preprint arXiv:2411.00027*, 2024.

[17] J. Guan, J. Wu, J.-N. Li, C. Cheng, and W. Wu, *A survey on personalized alignment – the missing piece for large language models in real-world applications*, 2025. arXiv: 2503.17003 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2503.17003.

[18] L. Alberts, B. Ellis, A. Lupu, and J. Foerster, *Curate: Benchmarking personalised alignment of conversational ai assistants*, 2025. arXiv: 2410.21159 [cs.HC]. [Online]. Available: https://arxiv.org/abs/2410.21159.

[19] S. Zhao, M. Hong, Y. Liu, D. Hazarika, and K. Lin, "Do llms recognize your preferences? evaluating personalized preference following in llms," *arXiv preprint arXiv:2502.09597*, 2025.

[20] L. Castricato, N. Lile, R. Rafailov, J.-P. Fränken, and C. Finn, "PERSONA: A reproducible testbed for pluralistic alignment," in *Proceedings of the 31st International Conference on Computational Linguistics*, O. Rambow, L. Wanner, M. Apidianaki, H. Al-Khalifa, B. D. Eugenio, and S. Schockaert, Eds., Abu Dhabi, UAE: Association for Computational Linguistics, Jan. 2025, pp. 11 348–11 368.

[21] H. R. Kirk et al., "The prism alignment dataset: What participatory, representative and individualised human feedback reveals about the subjective and multicultural alignment of large language models," *Advances in Neural Information Processing Systems*, vol. 37, pp. 105 236–105 344, 2024.

[22] S. Poddar, Y. Wan, H. Ivison, A. Gupta, and N. Jaques, "Personalizing reinforcement learning from human feedback with variational preference learning," in *Advances in Neural Information Processing Systems*, A. Globerson et al., Eds., vol. 37, Curran Associates, Inc., 2024, pp. 52 516–52 544. DOI: 10.52202/079017-1664.

[23] Z. Zhang et al., "Amulet: Realignment during test time for personalized preference adaptation of llms," *arXiv preprint arXiv:2502.19148*, 2025.

[24] S. Poddar, Y. Wan, H. Ivison, A. Gupta, and N. Jaques, "Personalizing reinforcement learning from human feedback with variational preference learning," *Advances in Neural Information Processing Systems*, vol. 37, pp. 52 516–52 544, 2024.

[25] M. Kim, K.-i. Lee, S. Joo, H. Lee, T. Thonet, and K. Jung, "Drift: Decoding-time personalized alignments with implicit user preferences," *arXiv preprint arXiv:2502.14289*, 2025.

[26] S. Lee, S. H. Park, S. Kim, and M. Seo, "Aligning to thousands of preferences via system message generalization," *Advances in Neural Information Processing Systems*, vol. 37, pp. 73 783–73 829, 2024.

[27] A. Köpf et al., "Openassistant conversations-democratizing large language model alignment," *Advances in neural information processing systems*, vol. 36, pp. 47 669–47 681, 2023.

[28] N. Lambert et al., "Rewardbench: Evaluating reward models for language modeling," in *Findings of the Association for Computational Linguistics: NAACL 2025*, 2025, pp. 1755–1797.

[29] W.-L. Chiang et al., "Chatbot arena: An open platform for evaluating llms by human preference," in *Forty-first International Conference on Machine Learning*, 2024.

[30] Y. Dubois, B. Galambosi, P. Liang, and T. B. Hashimoto, *Length-controlled alpacaeval: A simple way to debias automatic evaluators*, 2025. arXiv: 2404.04475 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2404.04475.

[31] L. Zheng et al., "Judging llm-as-a-judge with mt-bench and chatbot arena," *Advances in neural information processing systems*, vol. 36, pp. 46 595–46 623, 2023.

[32] OpenAI, *Gpt-4o system card*, 2024. arXiv: 2410.21276 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2410.21276.

[33] A. Afzali, A. Afsharrad, S. S. Mousavi, and S. Lall, "One goal, many challenges: Robust preference optimization amid content-aware and multi-source noise," *arXiv preprint arXiv:2503.12301*, 2025.

[34] *Gpt-4o mini: Advancing cost-efficient intelligence*, Accessed 2026-01-24, 2024. [Online]. Available: https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/.

[35] "The claude 3 model family: Opus, sonnet, haiku. "[Online]. Available: https://api.semanticscholar.org/CorpusID:268232499.

[36] Meta, *The llama 3 herd of models*, 2024. arXiv: 2407.21783 [cs.AI]. [Online]. Available: https://arxiv.org/abs/2407.21783.

[37] A. Singh, "Meta llama 4: The future of multimodal ai," *SSRN Electronic Journal*, p. 15, Apr. 2025. DOI: 10.2139/ssrn.5208228.

[38] A. Afsharrad, S. Jaladi, N. Yazdani, A. Ansari, S. S. Mousavi, and S. Lall, "MELISSA: Multi-level evaluation with LLM-based integrated self-scrutiny and auditing," in *NeurIPS 2025 Workshop on Multi-Turn Interactions in Large Language Models*, 2025. [Online]. Available: https://openreview.net/forum?id=ZaxCviYEvT.

[39] C. Kapfer, K. Stine, B. Narasimhan, C. Mentzel, and E. Candes, *Marlowe: Stanford's gpu-based computational instrument*, version 0.1, Jan. 2025. DOI: 10.5281/zenodo.14751899.