Who Does the AI Judge Represent? Group-Level Evaluation of AI Judges on PRISM

Ben Gur
Department of Computer Science
Stanford University
Stanford, CA
bgur@stanford.edu

Abstract

Large language models are usually aligned using preference learning from human feedback, where a reward model or "judge" is trained to predict which outputs people prefer. Most research combines human feedback into one scalar signal, assuming human preferences are uniform or baking them together. In reality, preferences vary by region, demographics, and values, so a judge that works well on average may heavily misrepresent preferences for certain groups. In my project, I use the PRISM Alignment Dataset, which links survey data to live conversations and contextual feedback with multiple language models, to study performance of a trained judge on different groups. I will explore how accurate and well-calibrated a single global judge is overall, what can be drawn from its performance as it varies across different groups, and whether simple methods such as group-specific calibration or group-conditioned inputs can improve performance without significantly affecting global performance.

1 Introduction

Large language models (LLMs) are typically aligned using preference learning from human feedback. In these pipelines, models generate responses, humans compare them, and a reward model (judge) is trained to predict which response is preferred. This judge is then used for policy optimization or to score and filter generations. Work such as Christiano et al.[Christiano et al., 2017], Ziegler et al.[Ziegler et al., 2019], Ouyang et al.[Ouyang et al., 2022], and Rafailov et al.[Rafailov et al., 2023] have shown similar preference-based training improving model behavior substantially.

However in those examples feedback is drawn from multiple groups into a single scalar reward, effectively treating human preferences as uniform. Annotators differ in demographics, lived experience, and value priorities. A reward model that performs well on average may perform much worse for specific groups, but this seems to be rarely measured or discussed. This raises the central question for this project: how well does a single learned AI judge align with the contextual preferences of different groups of people, what can we learn from the differences and can simple group-aware adjustments reduce systematic mismatches?

I will heavily use the PRISM Alignment Dataset [Kirk et al., 2024] to answer these questions. It links survey data on socio-demographics and values for roughly 1,500 participants from 75 countries with live conversations between those participants and 21 LLMs, not to mention feedback on model behavior. This structure lets me ask not only how accurate a judge is overall, but also for whom it works well or poorly.

I focus on three research questions. One, how accurate and well-calibrated is a single global judge model on PRISM overall? Two, how does judge performance vary systematically across demographic

Foundation Models for Science Workshop,38th Conference on Neural Information Processing Systems (NeurIPS 2024).

and value-defined groups such as region, age bracket, gender, and selected value dimensions? Three, can lightweight group-aware methods, such as group-specific calibration or group-conditioned inputs, improve group performance without noticeably harming global performance?

This project connects directly to CS329H topics like discrete choice, Bradley–Terry style models (built on existing text embeddings), and classical choice theory [Luce, 1959, Train, 2003]. The "judge" we'll examine is a smaller-scale analogue of reward models in RLHF and DPO [Christiano et al., 2017, Ziegler et al., 2019, Ouyang et al., 2022, Rafailov et al., 2023]. The work also relates to ideas of multiple-humans and aggregation, including social choice (e.g., Sen, Moulin), Distributional Preference Learning [Siththaranjan et al., 2024], and scaling human judgment. The scope is realistic for a solo project: I will use existing data, a pre-trained open LLM as a frozen encoder, and small preference heads plus simple calibration layers rather than training large models from scratch.

2 Related Work

RLHF and preference-based reward models have become a standard alignment tool. Christiano et al.[Christiano et al., 2017] learns a reward function from pairwise human comparisons and train reinforcement learning agents against that learned reward. Ziegler et al.[Ziegler et al., 2019] apply a similar approach to fine-tune language models from human preferences over text responses. Ouyang et al.[Ouyang et al., 2022] scale this methodology to large instruction-following models for InstructGPT, and Rafailov et al.[Rafailov et al., 2023] introduce direct preference optimization, which optimizes a preference-based objective from logged data without an explicit RL loop. These works demonstrate the effectiveness and popularity of scalar reward models built from pairwise preferences, but they mostly report global performance and treat feedback as coming from a single aggregate "human grader" rather than analyzing group fits.

Siththaranjan et al.[Siththaranjan et al., 2024] argues that standard preference learning is affected heavily by annotator identity, instructions, and situational context, and that RLHF can form a choice rule over these contexts. He discusses distributional preference learning (DPL), which models distributions over scores rather than a single scalar reward to better capture heterogeneity and uncertainty. My project adopts a simpler version of this idea: in PRISM, parts of the context (demographics and values) are observed, so I can explicitly slice performance across these contexts and test small group-related modifications.

Classical social choice work (e.g., Sen, Moulin) shows that turning many people's preferences into one decision always involves trade-offs, and recent alignment work often responds by designing new mechanisms that keep multiple human perspectives explicitly in the loop, such as Tessler et al.'s Habermas Machine and Li et al.[Li et al., 2024] using LLMs to help scale Community Notes. In summary, most RLHF and reward-model papers report global metrics and treat feedback as a single aggregate signal, and DPL provides a theoretical analysis of hidden context but not an empirical group-level study on real heterogeneous feedback. My project takes a different angle: instead of creating yet another aggregation scheme, it takes a step back to focus specifically on the performance of a single aggregate "judge" and uses PRISM to run group-level audits. Within a concrete preference-learning setting I will discover whose preferences a single learned judge actually represents, which groups it systematically misfits, and investigate high-impact personalization techniques to narrow these gaps without needing completely different pipeline architecture.

3 Methods and Analysis Plan

3.1 Data and task

As stated I will use the PRISM Alignment Dataset [Kirk et al., 2024]. The survey file contains sociodemographic attributes (such as country or region, age bracket, and gender) and value-related responses for each participant. The conversations file contains, for each conversation, the user ID, the conversation history (human and model turns), and feedback on model responses. I will join these to obtain examples where a participant with a known profile evaluates particular model outputs.

Whenever possible, I will frame this as a pairwise preference task. For each example, I take a prompt (or short conversation context) and two candidate responses from different models, and the label is simply which response the participant liked more. If PRISM already contains explicit comparisons

between responses, I will use those directly. If it only has ratings, I will create pairs by saying A is preferred to B when A has a higher rating than B and ignore ties. If this turns out to be hard in some parts of the dataset, I can fall back to predicting scalar ratings, but the main analysis will focus on pairwise comparisons, since this matches the Bradley–Terry and Plackett–Luce style models used in class.

3.2 Judge model

For the judge, I will use a pre-trained open-source instruction-tuned model, such as Llama 3.1 8B Instruct, and treat it as a frozen text encoder. For each example, I will feed in the conversation context and a candidate response and take hidden state as a fixed-length embedding. For a pair of responses, I will compute embeddings for both and build a simple feature (for example, the difference between the two vectors). On top of this feature, I will train a small logistic classifier that predicts which response the participant preferred. In the main experiments, the LLM weights will stay frozen and only this classifier will be trained. If time and compute allow, I may do a small follow-up experiment where I unfreeze a few of the top layers, but this is not required for the core analysis.

3.3 Group definitions

Using the PRISM survey, I will define a small set of groups with adequate sample sizes. For region, I plan to cluster countries into a few broad groups (for example, North America, Western Europe, other high-income, and other regions), depending on the actual distribution in PRISM. For age, I will use PRISM's age bins and may merge them into brackets such as 18–29, 30–44, and 45+. For gender, I will use the recorded categories. For values, I will select at least one scale and define groups in ways like "high" and "low" based on a median or other split. I will only report detailed metrics for groups with sufficient data and will clearly mark any experimental slices.

3.4 Experimental conditions

All experiments will use participant-level splits, so that train, validation, and test sets are specific to user IDs.

First, I will train a global judge that does not receive any group information. This model will be trained on all training examples and evaluated on the test set using overall accuracy and log-loss, as well as group-wise metrics by slicing the test set by region, age, gender, and at least one value dimension. This addresses our first two research questions.

Second, I will add group-aware modifications. I will keep the global judge fixed, but adjust its output separately for each group using the validation set. For each group, I will learn a small correction (for example, a simple rescaling of the scores) and apply it at test time. I will then compare calibration metrics (such as expected calibration error) and accuracy before and after calibration, both overall and by group. This will show whether performance gaps are mostly due to miscalibration or deeper inaccuracies. This addresses our second and third research questions.

Third, I will train a group-conditioned judge that directly uses group information as input. For each example, I will add tokens that encode the user's region, age bracket, and one value dimension before the prompt and response text. I will retrain the pref-head on these inputs and, if time and resources allow, lightly fine-tune part of the encoder. I will evaluate this model with the same overall and group-wise metrics as before. Comparing it to the global and calibrated judges will round off answering our third research question.

3.5 Evaluation metrics and hypotheses

For prediction, I will report pairwise accuracy, defined as the fraction of pairs where the judge's prediction matches the participant's preference, and log-loss. For calibration, I will compute reliability curves and a scalar measure such as expected calibration error, both globally and for groups. To report heterogeneity, I will provide group-wise accuracy and calibration, as well as summary statistics such as worst-group accuracy and the gap between best and worst groups.

I hypothesize that the global judge will show noticeable performance differences across groups, with some underrepresented or value-polar groups experiencing lower accuracy and worse calibration. I

expect group-specific calibration to cause a conditioned judge to have improved performance for at least select groups and raise worst-group accuracy significantly while global accuracy raises slightly.

4 Timeline and Risks

I am working solo, so I am responsible for implementation, experiments, and writing.

Week 1 (Nov 5–9): Load the PRISM data and understand how the survey and conversation files are structured. Decide on the main groups I care about (regions, age ranges, and at least one value dimension). Write preprocessing code that links survey rows to conversations by user ID and extracts (prompt, response, feedback) examples. Train a very simple rating model with a frozen encoder just to make sure the whole pipeline runs end to end.

Week 2 (Nov 10–16): Turn the feedback into pairwise labels (or use explicit comparisons if PRISM already has them). Implement and train a global judge model with a frozen encoder and a small classifier head, using appropriate data splits between train, validation, and test. Compute overall and per-group performance numbers, and make a few basic tables or plots to see how performance differs across groups.

Week 3 (Nov 17–23): Add group-specific calibration on top of the global judge and check how much it improves calibration and group gaps. Build the group-conditioned judge that takes group tokens as input and retrain the head. Compare the global, calibrated, and group-conditioned models across groups and review main findings.

Week 4 (Nov 24–Dec 3): Clean up and finalize all metrics, plots, and summary numbers, focusing on group differences and calibration. Look at concrete examples where the judge disagrees with people in specific groups and describe the main patterns. Write the full paper and prepare my project code.

The main risks are data sparsity for some groups, dataset schema complexity, and compute or time limitations. I will address sparsity by using population thresholds in my groupings and restricting claims to those supported by enough group data. I will address schema complexity by building a simple rating-prediction model using the most clearly labeled feedback fields. Once I understand the data format well, I will convert these ratings into pairwise preference labels and move to the full pairwise preference setting. I will address compute constraints by relying primarily on a frozen encoder and small heads; any encoder fine-tuning or secondary datasets will be treated as stretch goals.

References

References

Paul Christiano, Jan Leike, Tom Brown, et al. Deep reinforcement learning from human preferences. In *Advances in Neural Information Processing Systems*, 2017.

Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, et al. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2019.

Long Ouyang, Jeff Wu, Xu Jiang, et al. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, 2022.

Nikolay Rafailov, Eric Mitchell, Archit Sharma, et al. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems*, 2023.

Hannah Rose Kirk, et al. The PRISM Alignment Dataset: Mapping socio-demographics, values and ethical stances to LLM preferences. *NeurIPS*, 2024 (to appear).

R. Duncan Luce. *Individual Choice Behavior: A Theoretical Analysis*. Wiley, 1959.

Kenneth Train. *Discrete Choice Methods with Simulation*. Cambridge University Press, 2003.

Kalesh Siththaranjan, Cyrus A. DiGenova, David Krueger, and Dylan Hadfield-Menell. Distributional preference learning: Modeling preference diversity and uncertainty in RLHF. *arXiv preprint arXiv:2406.XXXXX*, 2024.

Fedor Z. Li, et al. Scaling human judgment in Community Notes with large language models. *arXiv preprint arXiv:2403.XXXXX*, 2024.