# Pareto-Optimal Learning from Preferences with Hidden Context

**Ryan Boldi** UMass Amherst
**Li Ding*** UMass Amherst
**Lee Spector** Amherst College
**Scott Niekum** UMass Amherst

## Abstract

Ensuring AI models align with human values is essential for their safety and functionality. Reinforcement learning from human feedback (RLHF) uses human preferences to achieve this alignment. However, preferences sourced from diverse populations can result in point estimates of human values that may be sub-optimal or unfair to specific groups. We propose Pareto Optimal Preference Learning (POPL), which frames discrepant group preferences as objectives with potential trade-offs, aiming for policies that are Pareto-optimal on the preference dataset. POPL utilizes Lexicase selection, an iterative process to select diverse and Pareto-optimal solutions. Our empirical evaluations demonstrate that POPL surpasses baseline methods in learning sets of reward functions, effectively catering to distinct groups without access to group numbers or membership labels.

## 1 Introduction

For both safety and functionality, it is critical for AI models to be aligned with the values of human users and stakeholders. Reinforcement learning from human feedback (RLHF) has emerged as an effective mechanism for model alignment, using preferences as a means of capturing human values. However, when preferences are sourced from large sets of potentially diverse people, methods that rely on point estimates of human values are bound to either be sub-optimal with respect to all groups or unfair to certain groups, both problematic in their own ways.

In this work, we build upon the notion of hidden context proposed by Siththaranjan et al. [49] and focus on the problem of Reinforcement Learning from Human Feedback with Hidden Context (RLHF-HC). Hidden context is information that is unavailable to a preference learning system that affects the preferences given. For example, a person's dominant hand might determine on what side a robotic assistant should hand them an object. Under this formulation, the goal is to build a *set* of policies that contains the policy that is optimal under the reward function of each group of people with the same hidden context. In practice we believe that these polices can be used to cater to specific groups, or to ensure fairness between groups.

Because preferences generated from individuals in different hidden context groups could be contradictory, we propose to frame these preferences as objectives with potential trade-offs between each other. With this re-framing, the optimal policy for each individual hidden context group would be Pareto-optimal (non-dominated) on the dataset of preferences. With this in mind, we propose Pareto Optimal Preference Learning (POPL), where we learn a set of reward functions or policies (directly) that are optimized towards being Pareto-optimal with respect to the set of preferences given by a potentially diverse set of human annotators. To do this, we use an iterative selection process known as Lexicase selection [50], which has been shown under mild assumptions to select individuals that are both Pareto-optimal and diverse. An outline of our method can be found in Figure 1.

*Now at Google

Pluralistic Alignment Workshop at NeurIPS 2024.

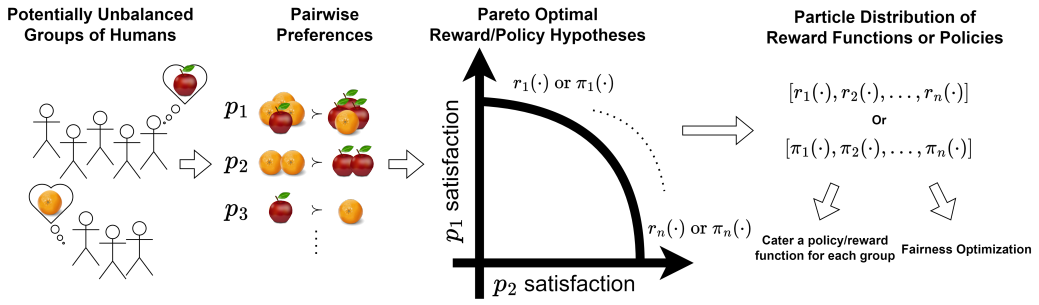



Figure 1: Outline of our proposed method. Given a set of pairwise preferences over trajectory segments from groups with potentially different ground truth reward functions, we infer a set of reward functions or policies that captures each group's ground truth, without group membership labels. To do this, we frame reward inference as multi-objective optimization, where each preference forms a single objective, and find a set of Pareto-optimal reward functions or policies.

## 2 Problem Statement and Theoretical Foundation

Given a dataset $D = \{\sigma_1, \cdots, \sigma_m\}$ of trajectory segments and a set $\mathcal{P} = \{(i,j) : \sigma_i \succ \sigma_j\}$ of pairwise preferences over these segments, we wish to infer an unknown reward function $r : \mathcal{S} \mapsto \mathbb{R}$ that respects the preferences. In order to account for hidden context in the preferences learned, Siththaranjan et al. [49] introduce Distributional Preference Learning, which relies on a single model of utility $u(s,z)$ to output a distribution of utility assignments $u(\cdot,z)$ for each state $s \in \mathcal{S}$, marginalizing over the hidden context variable $z$.

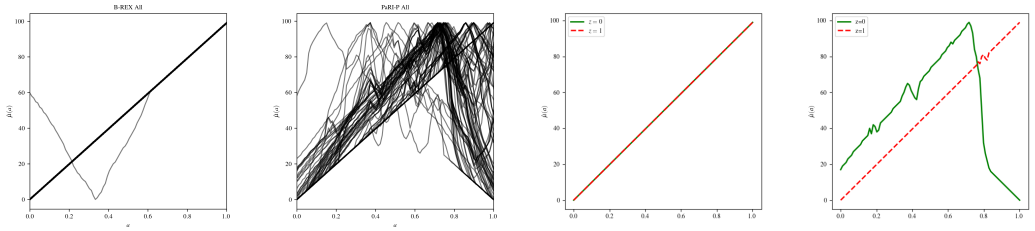
However, due this marginalization process, the utility function cannot attune to *persistent annotator identity*—the fact that hidden context transcends a single preference annotation. In light of this, we re-frame the problem of preference learning with hidden context as follows. The goal is to learn a set $\Pi = \{\pi_1, \pi_2 \cdots, \pi_n\}$ of policies such that, for the hidden context group represented by a variable $z \in \mathcal{Z}$, there is a policy $\pi_z \in \Pi$ that is the optimal policy for the ground truth reward function $r_z$ for the group. Note that this can be accomplished by standard (reward-based) or direct (reward-free) RLHF. For the standard approach, a series of reward functions $R = \{r_1, r_2 \cdots, r_n\}$ are first learned from preferences, then used to train $n$ policies. In the direct approach, the policies are learned directly from preferences such as done by Hejna et al. [31] and Rafailov et al. [45].

Related work and other preliminary information can be found in Appendix A and Appendix B respectively. Formal definitions of terms and proofs for our claims can be found in Appendix E.

## 3 Pareto Optimal Preference Learning

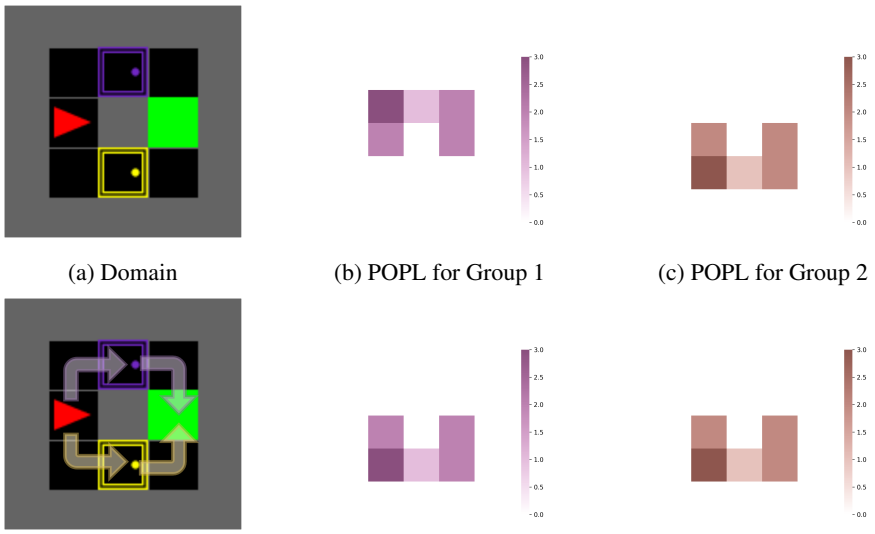
The intuition we rely on for POPL is that, in a setting with low noise and sufficient regularization, the set of policies or reward functions that coincide with the various ground truth reward functions (one for each hidden context group) exist on the Pareto front, where each axis corresponds to a single preference. To obtain a set of Pareto optimal policies, we adopt the idea of lexicase selection [50, 32]. A key property of lexicase selection is that it selects candidates that are Pareto-optimal relative to a starting set of candidates that tend to spread the *corners* of the Pareto front and thus be diverse [39]. This idea has been utilized in many machine learning optimization problems for improving generalization, as shown in recent work [21, 22, 43, 8, 20]. In the context of preference-based reward learning, preferences are used as binary metrics that filter down candidates.

With a method to select Pareto-optimal candidates such as lexicase selection in hand, one can infer a set of reward functions or policies directly from preferences. Initially, a random set of candidate models is created. The chosen method is then applied to select (with replacement) candidates resulting in a pool of Pareto-optimal candidates. This pool is perturbed by adding random Gaussian noise, generating a new set of candidates. This is repeated for a number of iterations. The final set of candidates should align with the preferences of hidden context groups. A full algorithm can be found in Appendix C.



(a) B-REx Distribution (b) POPL Final Set (c) B-REx Catering (d) POPL Catering

Figure 2: (a) and (b) shows all the reward functions that are the output of the reward inference methods. (c) and (d) show the catered reward functions for each of the two hidden context groups $z = 0$, $z = 1$. To ensure the reward functions are comparable, for all these figures, the $y$-axis shows how the underlying reward functions rank the 100 differing state interpolations ($x$-axis). From a set of reward functions that is inferred from a diversity of human preferences, we select a single reward function for each unique group with a small number of preferences (2% the size of the training set). POPL is able to cater for both groups, while B-REx is only able to cater for one of the two groups.



(a) Domain (b) POPL for Group 1 (c) POPL for Group 2

(d) Hidden Context Groups (e) MultiCPL for Group 1 (f) MultiCPL for Group 2

Figure 3: (a) An example domain where hidden context is present. The agent (red triangle) must make it to the solid green goal tile as fast as possible. The agent must choose one of the two doors to use to reach the goal. (d) Depending on their identity, a human might prefer a policy that uses the top door over the bottom door, or vice versa. (b) and (c) show the aggregate state occupancy of the policies personalized for each group using POPL. (e) and (f) show it for MultiCPL. The set that is generated by MultiCPL does not contain the optimal policy for Group 1. The set generated by POPL contains the optimal policy for both groups, without having access to annotator identity labels.

## 4 Experiments

In this section, we detail our experimental results to validate the proposed POPL method. Further implementation details are provided in Appendix F.

**Synthetic Stateless Experiment** Following the synthetic experiments outlined by Siththaranjan et al. [49], the first set of experiments we perform will test whether POPL is able to recover the optimal reward functions from a series of preferences from two groups of humans defined by a hidden context variable $z \sim \mathcal{B}(0.5)$ where $\mathcal{B}(0.5)$ is a Bernoulli distribution. We compare the distributions based on their representation of the hidden context groups that might have different preferences. The utility in this scenario can be modeled as $u(a, z) = a$ if $a < 0.8$ and $2az$ otherwise.

Table 1: Results on LLM jailbreaks. POPL has the lowest jailbreak rate across all methods without any fairness optimization. For fairness optimization, POPL has a lower jailbreak rate than B-REx, standard RLHF, as well as Mean & var. DPL, and is competitive with categorical DPL, without needing to parameterize a hidden context distribution in advance.

| Method | Training data | Jailbreak rate (%) | Helpfulness acc. (%) |
|---|---|---|---|
| Standard | Helpful | 52.4 | 72.6 |
| Standard | Harmless | 3.7 | 49.5 |
| Standard | Combined | 25.1 | 68.2 |
| Mean & var. DPL | Combined | 30.5 | **68.4** |
| ↳ Fair | | 20.3 | **66.4** |
| Categorical DPL | Combined | 32.1 | 66.2 |
| ↳ Fair | | **13.4** | 66.2 |
| Bayesian REx | Combined | 28.3 | 67.5 |
| ↳ Fair | | 27.8 | 50.4 |
| POPL (Ours) | Combined | **17.6** | 66.1 |
| ↳ Fair | | 15.0 | 65.7 |

**Minigrid Policy Inference** We perform a second set of experiments to verify whether 1) POPL is able to generate *policies* directly from preferences, and 2) POPL is able to perform in a sequential RL domain, where annotators' hidden context is persistent (i.e. potentially affects more than one segment preference annotation).The domain used in these experiments is outlined in Figure 3a. The hidden context groups in this scenario delineate whether the annotator inherently prefers the bottom or top door to be used to get to the goal. The preferences were labeled according to the regret preference model from members of both groups (extracted from the optimal policy for each group's ground truth reward model). We find that POPL is able to successfully cater policies for both groups of people, despite not having labels regarding their group membership.

**Language Model Experiments** A final set of experiments that we perform tests the ability of POPL to scale to domains involving human annotations. We investigate whether POPL can be sensitive to hidden context in whether annotators prefer *harmless* or *helpful* responses [1] by a language model. Wei et al. [52] find examples of user prompts that directly pit harmlessness and helpfulness against each other, leading a language model to output harmful outputs, a phenomenon known as *jailbreaking*. An RLHF system built with hidden context in mind would help detect jailbreaking before a harmful output would be given to a user. Table 1 presents the jailbreak rates and helpfulness accuracy for standard RLHF, B-REx, DPL, and our proposed POPL.

## 5 Conclusion

When learning from human preferences for the sake of aligning to human values, one must first decide *which* humans to pay attention to. As these systems rely on point estimates of return or regret, they can at most align with a single human's preferences. We have formalized this as the problem of reinforcement learning from human preferences with hidden context. Under this conception, a set of policies must be generated that contains the optimal policy for each group of people.

To solve this problem, we relied on the concept of Pareto-optimality to generate a series of reward functions and/or policies that are optimal with respect to unique sub-sets of preferences. To optimize towards Pareto-optimality, we used a technique known as lexicase selection, that selects individuals from a large set based on a randomized (lexicographic) prioritization of the training data.

We verified that lexicase selection can be used to generate diverse distributions of either reward functions or policies that align with the diverse preferences that human annotators have. We evaluated and verified the performance of POPL in a variety of domains, including a synthetic stateless domain, a sequential RL domain, and even language model jailbreak detection. Across these domains, we have demonstrated POPL's efficacy when compared to contemporary algorithms in dealing with hidden context in the preferences.

## References


[1] Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

[2] Chris L. Baker, Rebecca Saxe, and Joshua B. Tenenbaum. Action understanding as inverse planning. *Cognition*, 113(3):329–349, December 2009. ISSN 00100277. doi: 10.1016/j.cognition.2009.07.005. URL https://linkinghub.elsevier.com/retrieve/pii/S0010027709001607.

[3] Peter Barnett, Rachel Freedman, Justin Svegliato, and Stuart Russell. Active reward learning from multiple teachers. *arXiv preprint arXiv:2303.00894*, 2023.

[4] Connor Baumler, Anna Sotnikova, and Hal Daumé III. Which examples should be multiply annotated? active learning when annotators may disagree. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 10352–10371, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.658. URL https://aclanthology.org/2023.findings-acl.658.

[5] Erdem Biyik and Dorsa Sadigh. Batch active preference-based learning of reward functions. In Aude Billard, Anca Dragan, Jan Peters, and Jun Morimoto (eds.), *Proceedings of The 2nd Conference on Robot Learning*, volume 87 of *Proceedings of Machine Learning Research*, pp. 519–528. PMLR, 29–31 Oct 2018. URL https://proceedings.mlr.press/v87/biyik18a.html.

[6] Andreea Bobu, Dexter RR Scobee, Jaime F Fisac, S Shankar Sastry, and Anca D Dragan. Less is more: Rethinking probabilistic models of human behavior. In *Proceedings of the 2020 acm/ieee international conference on human-robot interaction*, pp. 429–437, 2020.

[7] Andreea Bobu, Andi Peng, Pulkit Agrawal, Julie Shah, and Anca D Dragan. Aligning robot and human representations. *arXiv preprint arXiv:2302.01928*, 2023.

[8] Ryan Boldi, Li Ding, and Lee Spector. Objectives are all you need: Solving deceptive problems without explicit diversity maintenance. In *Second Agent Learning in Open-Endedness Workshop*, 2023.

[9] Herbie Bradley, Andrew Dai, Hannah Benita Teufel, Jenny Zhang, Koen Oostermeijer, Marco Bellagente, Jeff Clune, Kenneth Stanley, Gregory Schott, and Joel Lehman. Quality-diversity through AI feedback. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=owokKCrGYr.

[10] Ralph Allan Bradley and Milton E. Terry. RANK ANALYSIS OF INCOMPLETE BLOCK DESIGNS: THE METHOD OF PAIRED COMPARISONS. *Biometrika*, 39(3-4):324–345, 1952. ISSN 0006-3444, 1464-3510. doi: 10.1093/biomet/39.3-4.324. URL https://academic.oup.com/biomet/article-lookup/doi/10.1093/biomet/39.3-4.324.

[11] Daniel Brown, Russell Coleman, Ravi Srinivasan, and Scott Niekum. Safe imitation learning via fast bayesian reward inference from preferences. In *International Conference on Machine Learning*, pp. 1165–1177. PMLR, 2020.

[12] Daniel Brown, Scott Niekum, and Marek Petrik. Bayesian Robust Optimization for Imitation Learning. In *Advances in Neural Information Processing Systems*, volume 33, pp. 2479–2491. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/hash/1a669e81c8093745261889539694be7f-Abstract.html.

[13] Daniel S. Brown and Scott Niekum. Deep Bayesian Reward Learning from Preferences, December 2019. URL http://arxiv.org/abs/1912.04472. arXiv:1912.04472 [cs, stat].

[14] Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, et al. Open problems and fundamental limitations of reinforcement learning from human feedback. *arXiv preprint arXiv:2307.15217*, 2023.

[15] Souradip Chakraborty, Jiahao Qiu, Hui Yuan, Alec Koppel, Furong Huang, Dinesh Manocha, Amrit Singh Bedi, and Mengdi Wang. MaxMin-RLHF: Towards Equitable Alignment of Large

Language Models with Diverse Human Preferences, February 2024. URL http://arxiv.org/abs/2402.08925. arXiv:2402.08925 [cs].

[16] Maxime Chevalier-Boisvert, Bolun Dai, Mark Towers, Rodrigo de Lazcano, Lucas Willems, Salem Lahlou, Suman Pal, Pablo Samuel Castro, and Jordan Terry. Minigrid & miniworld: Modular & customizable reinforcement learning environments for goal-oriented tasks. *CoRR*, abs/2306.13831, 2023.

[17] P. Christiano, J. Leike, Tom B. Brown, Miljan Martic, S. Legg, and Dario Amodei. Deep Reinforcement Learning from Human Preferences. *ArXiv*, June 2017. URL https://www.semanticscholar.org/paper/5bbb6f9a8204eb13070b6f033e61c84ef8ee68dd.

[18] Josef Dai, Xuehai Pan, Ruiyang Sun, Jiaming Ji, Xinbo Xu, Mickel Liu, Yizhou Wang, and Yaodong Yang. Safe rlhf: Safe reinforcement learning from human feedback. *arXiv preprint arXiv:2310.12773*, 2023.

[19] Oliver Daniels-Koch and Rachel Freedman. The expertise problem: Learning from specialized feedback. *arXiv preprint arXiv:2211.06519*, 2022.

[20] Li Ding and Lee Spector. Optimizing neural networks with gradient lexicase selection. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=J_2xNmVcY4.

[21] Li Ding, Ryan Boldi, Thomas Helmuth, and Lee Spector. Lexicase Selection at Scale. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, pp. 2054–2062, July 2022. doi: 10.1145/3520304.3534026. URL http://arxiv.org/abs/2208.10719. arXiv:2208.10719 [cs].

[22] Li Ding, Edward Pantridge, and Lee Spector. Probabilistic lexicase selection. In *Proceedings of the Genetic and Evolutionary Computation Conference*, GECCO '23, pp. 1073–1081, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9798400701191. doi: 10.1145/3583131.3590375. URL https://doi.org/10.1145/3583131.3590375.

[23] Li Ding, Jenny Zhang, Jeff Clune, Lee Spector, and Joel Lehman. Quality diversity through human feedback: Towards open-ended diversity-driven optimization. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=9zlZuAAb08.

[24] Michael Feffer, Hoda Heidari, and Zachary C Lipton. Moral machine or tyranny of the majority? *arXiv preprint arXiv:2305.17319*, 2023.

[25] Chelsea Finn, Sergey Levine, and Pieter Abbeel. Guided cost learning: Deep inverse optimal control via policy optimization. In *International conference on machine learning*, pp. 49–58. PMLR, 2016.

[26] Gaurav R Ghosal, Matthew Zurek, Daniel S Brown, and Anca D Dragan. The effect of modeling human rationality level on learning rewards from multiple feedback types. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 5983–5992, 2023.

[27] Noah D. Goodman and Andreas Stuhlmüller. Knowledge and Implicature: Modeling Language Understanding as Social Cognition. *Topics in Cognitive Science*, 5(1):173–184, January 2013. ISSN 1756-8757, 1756-8765. doi: 10.1111/tops.12007. URL https://onlinelibrary.wiley.com/doi/10.1111/tops.12007.

[28] Noah D. Goodman, Tomer D. Ullman, and Joshua B. Tenenbaum. Learning a theory of causality. *Psychological Review*, 118(1):110–119, January 2011. ISSN 1939-1471, 0033-295X. doi: 10.1037/a0021336. URL https://doi.apa.org/doi/10.1037/a0021336.

[29] Mitchell L. Gordon, Kaitlyn Zhou, Kayur Patel, Tatsunori Hashimoto, and Michael S. Bernstein. The disagreement deconvolution: Bringing machine learning performance metrics in line with reality. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI '21, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450380966. doi: 10.1145/3411764.3445423. URL https://doi.org/10.1145/3411764.3445423.

[30] Mitchell L. Gordon, Michelle S. Lam, Joon Sung Park, Kayur Patel, Jeff Hancock, Tatsunori Hashimoto, and Michael S. Bernstein. Jury Learning: Integrating Dissenting Voices into Machine Learning Models. In *CHI Conference on Human Factors in Computing Systems*, pp. 1–19, New Orleans LA USA, April 2022. ACM. ISBN 978-1-4503-9157-3. doi: 10.1145/3491102.3502004. URL https://dl.acm.org/doi/10.1145/3491102.3502004.

[31] Joey Hejna, Rafael Rafailov, Harshit Sikchi, Chelsea Finn, Scott Niekum, W. Bradley Knox, and Dorsa Sadigh. Contrastive preference learning: Learning from human feedback without reinforcement learning. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=iX1RjVQODj.

[32] Thomas Helmuth, Lee Spector, and James Matheson. Solving Uncompromising Problems With Lexicase Selection. *IEEE Transactions on Evolutionary Computation*, 19(5):630–643, October 2015. ISSN 1089-778X, 1089-778X, 1941-0026. doi: 10.1109/TEVC.2014.2362729. URL http://ieeexplore.ieee.org/document/6920034/.

[33] Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

[34] Joel Jang, Seungone Kim, Bill Yuchen Lin, Yizhong Wang, Jack Hessel, Luke Zettlemoyer, Hannaneh Hajishirzi, Yejin Choi, and Prithviraj Ammanabrolu. Personalized Soups: Personalized Large Language Model Alignment via Post-hoc Parameter Merging, October 2023. URL http://arxiv.org/abs/2310.11564. arXiv:2310.11564 [cs].

[35] Hong Jun Jeon, Smitha Milli, and Anca Dragan. Reward-rational (implicit) choice: A unifying formalism for reward learning. *Advances in Neural Information Processing Systems*, 33:4415–4426, 2020.

[36] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[37] Hannah Rose Kirk, Bertie Vidgen, Paul Röttger, and Scott A Hale. Personalisation within bounds: A risk taxonomy and policy framework for the alignment of large language models with personalised feedback. *arXiv preprint arXiv:2303.05453*, 2023.

[38] W. Bradley Knox, Stephane Hatgis-Kessell, Serena Booth, Scott Niekum, Peter Stone, and Alessandro Allievi. Models of human preference for learning reward functions, June 2022. URL https://arxiv.org/abs/2206.02231v3.

[39] William La Cava, Thomas Helmuth, Lee Spector, and Jason H. Moore. A Probabilistic and Multi-Objective Analysis of Lexicase Selection and $\epsilon$-Lexicase Selection. *Evolutionary Computation*, 27(3):377–402, 2019. ISSN 1530-9304. doi: 10.1162/evco_a_00224.

[40] Kimin Lee, Laura Smith, Anca Dragan, and Pieter Abbeel. B-pref: Benchmarking preference-based reinforcement learning. *arXiv preprint arXiv:2111.03026*, 2021.

[41] David J. C. MacKay. A Practical Bayesian Framework for Backpropagation Networks. *Neural Computation*, 4(3):448–472, May 1992. ISSN 0899-7667, 1530-888X. doi: 10.1162/neco.1992.4.3.448. URL https://direct.mit.edu/neco/article/4/3/448-472/5654.

[42] Vivek Myers, Erdem Bıyık, Nima Anari, and Dorsa Sadigh. Learning Multimodal Rewards from Rankings, October 2021. URL http://arxiv.org/abs/2109.12750. arXiv:2109.12750 [cs].

[43] Andrew Ni, Li Ding, and Lee Spector. Dalex: Lexicase-like selection via diverse aggregation. *arXiv preprint arXiv:2401.12424*, 2024.

[44] Andi Peng, Aviv Netanyahu, Mark K Ho, Tianmin Shu, Andreea Bobu, Julie Shah, and Pulkit Agrawal. Diagnosis, feedback, adaptation: A human-in-the-loop framework for test-time policy adaptation. 2023.

[45] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36, 2024.

[46] Alexandre Rame, Guillaume Couairon, Corentin Dancette, Jean-Baptiste Gaya, Mustafa Shukor, Laure Soulier, and Matthieu Cord. Rewarded soups: towards pareto-optimal alignment by interpolating weights fine-tuned on diverse rewards. *Advances in Neural Information Processing Systems*, 36, 2024.

[47] Alexandre Ramé, Nino Vieillard, Léonard Hussenot, Robert Dadashi, Geoffrey Cideron, Olivier Bachem, and Johan Ferret. WARM: On the Benefits of Weight Averaged Reward Models, January 2024. URL http://arxiv.org/abs/2401.12187. arXiv:2401.12187 [cs].

[48] Shibani Santurkar, Esin Durmus, Faisal Ladhak, Cinoo Lee, Percy Liang, and Tatsunori Hashimoto. Whose Opinions Do Language Models Reflect?, March 2023. URL http://arxiv.org/abs/2303.17548. arXiv:2303.17548 [cs].

[49] Anand Siththaranjan, Cassidy Laidlaw, and Dylan Hadfield-Menell. Distributional preference learning: Understanding and accounting for hidden context in rlhf. In *arXiv preprint*, 2023.

[50] Lee Spector. Assessment of Problem Modality by Differential Performance of Lexicase Selection in Genetic Programming: A Preliminary Report. 2012.

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

[52] Alexander Wei, Nika Haghtalab, and Jacob Steinhardt. Jailbroken: How does llm safety training fail? *Advances in Neural Information Processing Systems*, 36, 2024.

## A Related Work

Data used for RLHF systems often comes from multiple people, who are diverse in their preferences and values [7, 44, 5, 48]. This data, when considered in its aggregated form, can not be captured perfectly by a decision-making model that relies on a point estimate of utility [14]. These models try to find a single reward function that is most likely, which is often not the optimal reward function for any one single person. When the groups are not perfectly balanced, the minority groups might be underrepresented in the inferred reward function [49, 24, 37, 42] or simply treated as noise [4]. There have been attempts at explicitly modeling different people with different levels of expertise [29, 19, 30, 3], but these methods generally rely on concrete ways to distinguish between groups.

In the context of RL, Myers et al. [42] outlines an approach to learning a multi-modal reward function from online interaction between a human expert and a preference learning system. Ramé et al. [47] learn a set of reward models by optimizing for diversity amongst the outputs. Whilst similar to our approach, we also aim to align our reward models with hidden context groups through optimizing for Pareto-optimality. For generative AI models and LLMs, there have been a variety of studies attempting to align large models with diverse human preferences. Chakraborty et al. [15] and Siththaranjan et al. [49] learn a mixture of preference distributions or a parameterized reward distribution, respectively. However, both these techniques operate under a contextual bandit setting which results in sub-optimal performance when used in the more general RL setting (discussed further in Section 2). Bradley et al. [9] and Ding et al. [23] leverage fine-tuning to improve the diversity of model responses for better alignment and creativity, which do not directly address the ambiguity and hidden context in human preferences. Jang et al. [34] and Dai et al. [18] elicit preferences specifically along different dimensions in order to cater custom reward functions for users in test time, and to be safe with respect to conflicting objectives, respectively. Whilst we also aim to cater reward functions in test time as well as optimize fairness between groups, we do not have access to labels regarding the context of the preferences generated. Finally, Rame et al. [46] also generates a set of Pareto-optimal reward functions. However, in their setting, the system has access to ground truth reward functions for each group, and the Pareto-front is generated through weight interpolation between these functions.

Bayesian Reward Extrapolation (B-REx) [12] instead learns a distribution of reward models from pairwise human preferences. B-REx is then able to perform Bayesian inference using MCMC [41] to sample from the posterior of reward functions. With this distribution, a practitioner can establish high confidence performance bounds that can be used to assess risk in evaluated policies as well as detect reward hacking behaviors. However, B-REx and other reward inference methods often rely on a faulty assumption that humans provide preferences in a Boltzmann-rational way.

On a separate front, there have been recent efforts on direct, reward-free methods for RLHF [45, 31], where a policy is learned directly from the preferences. However, these methods still implicitly rely on single-point estimates of reward, making them not directly applicable for reward function curation or risk minimization in safety-critical applications with hidden context. Furthermore, the hidden context in these preferences can lead to optimization issues when using these systems [15].

## B Preliminaries

**Learning from Human Preferences** Reinforcement learning from human feedback considers human preferences over trajectories (or more generally, outputs of a model) in order to learn a reward model or policy that respects the preferences [13, 45, 31, 14, 25].

In order to learn meaningfully from human preferences, one must characterize how preferences are generated from some parameterized preferences model $P(\sigma_i \prec \sigma_j)$. Usually, this preference model is based on the notion of Boltzmann-rationality, where humans generate preferences in accordance to the Bradley-Terry (BT) model [10]. The probability of pairwise preference $(\sigma_i \succ \sigma_j)$ between two trajectories given some utility function $f(\sigma)$ can be written as

$$P(\sigma_i \succ \sigma_j) = \frac{e^{\beta f(\sigma_i)}}{e^{\beta f(\sigma_j)} + e^{\beta f(\sigma_i)}} \tag{1}$$

where $\beta$ models the confidence in the preference labels. $\beta \to \infty$ signals that the preference provider is perfectly rational, and $\beta = 0$ signals that preferences are random. The Bradley-Terry model is

**Data:** A dataset of demonstrations $\mathcal{D}$ and a series of pairwise preferences $\mathcal{P}$
**Result:** A set of reward function or policy hypotheses

```
candidates ← randomly initialize p hypotheses
for iter 1 → N do
    for ind 1 → p do
        shuffled_prefs ← Shuffle(P)
        for pref in shuffled_prefs do
            old_subset ← candidates
            candidates ← subset of candidates that pass pref.
            // if all individuals have failed, we skip this preference
            // as it is likely to be contradictory with a previous preference
            if candidates contains no candidates then
                candidates ← old_subset
            end
            if candidates contains only one candidate then
                break
            end
        end
        candidate ← a random individual from candidates
        Append candidate to new population
    end
    candidates ← add random noise to candidates
end
return candidates
```

**Algorithm 1:** One Step of Pareto Optimal Preference Learning

used in many fields, such as psychology [2, 28, 27], However, this model does not perfectly capture the mechanisms driving these preferences [26, 35, 38, 6, 40].

**Contrastive Preference Learning** Contrastive Preference Learning (CPL) [31] learns a policy directly from preferences without needing to learn an intermediate reward function. This method uses a regret-based model of preference rather than the standard partial return interpretation. The probability of a preference under a candidate policy can be written as the ratio of the exponentiated sum of log-likelihoods of the chosen segment to the disregarded segment.

**Hidden Context** Siththaranjan et al. [49] introduce the problem of preference learning with hidden context. This is the idea that preferences are generated not only based on the exponential utility (partial return or regret), but also on some latent hidden context variable $z$. This variable is not accessible to preference learning systems and poses a challenge as it is often the case that this variable results in breaking the assumption that preferences are generated Boltzmann-rationally.

## C Full Algorithm

Algorithm 1 gives an outline of a single step of Pareto Optimal Preference Learning (POPL).

## D Issues with Marginalized Disributional Preference Learning Systems (MDPLs)

Concretely, assume that users that give the preference $\sigma_A \succ \sigma_B$, are likely to also give the preference $\sigma_B \succ \sigma_C$. Being aware of persistent annotator identity would allow for statements to be made like: "the *same people* that prefer $\sigma_A$ to $\sigma_B$ are also likely to prefer $\sigma_B$ to $\sigma_C$." Mathematically, MDPLs represent the marginalized probability $P(R|s)$ of a specific reward value $R$ in a state $s$. However, to be sensitive to persistent annotator identity, a preference learning framework would need to maintain the full conditional $P(R|s, z)$. In a contextual bandit setting such as those often found for finetuning LLMs [45], this is not an issue, as determining that an output has high risk simply depends on the distribution of rewards attributed to that specific state. However, when learning a policy using this

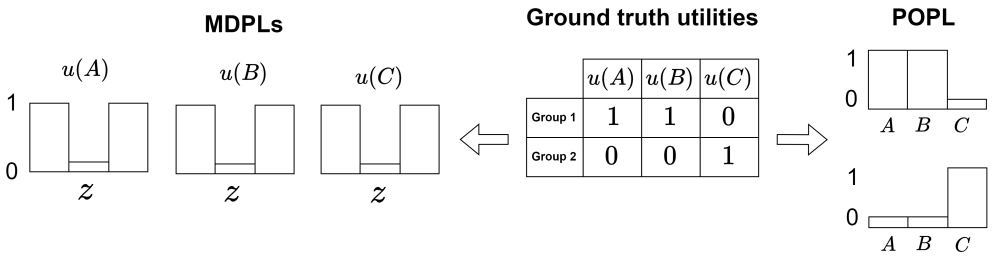



Figure 4: An example of a situation where using POPL is preferable to using a Marginalized Distributional Preference Learning (MDPL) system. Due to the fact that these systems marginalize over the hidden context $z$ for each state, MDPLs are unable to be sensitive to persistent annotator identity. MDPLs represent the distribution of utility values in a column-wise fashion, or maintain a distribution of utilities for each state, that is decoupled from that for other states. POPL, on the other hand, represents the distribution row-wise, finding a set of utility functions that should include the ground truth for each group.

reward function, none of the inter-segment preference dependence is captured. Below, we outline an example to help clarify why this is an issue.

**Example 1: MDPLs fail to distinguish between trajectories with different fairness profiles.** Let us consider two groups of people that have differing utilities for states $A, B, C$, as outlined in Figure 4. MDPL systems represent these utilities column-wise, as they maintain a distribution $u(s)$ that has been marginalized over $z$ for every state $s$. Now, consider the three trajectories $AB$, $BC$ and $AC$. These trajectories have different levels of utility when considering individual groups. For example, choosing $AB$ over either of the other two trajectories would have low utility and potentially be unfair (under certain notions of fairness) to an individual from group 2. MDPL systems would not be able to make this distinction, however, ignoring the group-specific utility. Despite the fact that group labels are not available to the preference learning framework, it would be better to maintain the full (un-marginalized) distribution $u(s, z)$ in order to be sensitive to persistent annotator identity. However, as we do not have group labels, we do not have information regarding the number of groups nor the $z$ values that correspond to them. Instead, we can find utility models that satisfy a diverse set of mutually compatible preferences. With enough of these models in a set, we assume that each hidden context group will have its utilities mapped by at least one of the models in the set.

## E Definitions and Proofs

**Definition 1** (Policy passing preference). A policy $\pi(a|s) : \mathcal{S} \times \mathcal{A} \mapsto [0, 1]$ passes a preference $(\sigma_i \prec \sigma_j)$ if the probability of the preferred segment $\prod_{(s,a)\in\sigma_j} \pi(a|s)$ is higher than the probability of the other segment $\prod_{(s,a)\in\sigma_i} \pi(a|s)$. Or, equivalently, if $\sum_{(s,a)\in\sigma_j} \log \pi(s, a) > \sum_{(s,a)\in\sigma_i} \log \pi(s, a)$.

**Definition 2** (Policy-set-relative Pareto-optimality). A policy $\pi(a|s) : \mathcal{S} \times \mathcal{A} \mapsto [0, 1]$ is Pareto optimal with respect to a set of preferences $\mathcal{P}$ relative to a set of other policies $\Pi = \{\pi_1, \pi_2, \cdots, \pi_n\}$ if and only if there exists a preference $(\sigma_i \succ \sigma_j) \in \mathcal{P}$ that $\pi$ passes but all other policies in $\Pi$ do not pass.

**Definition 3** (Hidden context group). A hidden context group is a group of $m$ annotators, each with their own reward function $r_1, r_2, \ldots, r_m$ that all monotonically rank the segments $\sigma \in \mathcal{D}$ the same.

**Definition 4** (Optimal policy for hidden context group). An optimal policy $\pi^*_g$ for a hidden context group $g$ is the optimal policy in a given environment (MDP/R) using the group's implicit ground truth reward function $r_g$ as the reward function.

**Definition 5** (Contradictory preferences). A pair of preferences $(\sigma_i \succ \sigma_j)$, $(\sigma_x \succ \sigma_y)$ are contradictory under a specific policy regularization scheme if the likelihood of any policy that satisfies both is lower than the likelihood of a policy that satisfies either.

With no regularization, a policy that satisfies two preferences should have a higher likelihood than one that satisfies one but not the other. This is because the likelihood of a certain policy is related to the total number of preferences passed by it. Passing two preferences would therefore elicit a higher likelihood than passing just one of them. However, if by satisfying both preferences, the models would need to incur a much greater regularization loss, it is likely that these two preferences came from individuals with differing hidden context.

**Theorem 1**. In a completely noiseless setting, all policies that are optimal for specific HC groups are Pareto optimal with respect to the set of all preferences $\mathcal{P}$ generated from all the groups, and the space of all possible policies $\Pi = \{\pi | \pi \in \mathcal{S} \times \mathcal{A} \mapsto [0, 1]\}$

A proof of Theorem 1 can be found in the appendix. In essence, a set of Pareto-optimal policies must each satisfy a unique set of mutually satisfiable preferences (ones that do not contain a contradictory preference). As such, the optimal policies for a group with hidden context would also be Pareto-optimal.

**Proof of Theorem 1 (Contradiction)**. Let us assume that there is a policy $\pi_z^*$ that is the optimal policy according to a hidden context group $z$. This means $\pi_z^*$ passes all the preferences compatible with the values of group $z$ and fails only the preferences that are not compatible with preferences given by group $z$. For sake of contradiction, we assert that this reward function is not Pareto optimal with respect to all other reward function candidates. This means there exists another reward function $\pi'$ that performs better than or equal to $\pi_z^*$ across every preference in $\mathcal{P}$, including those generated by group $z$. This is a clear contradiction as that would imply $\pi'$ is the optimal policy for group $z$ instead of $\pi_z^*$.

# F Experimental Details

In this section, we include more implementation details of our experiments.

**Baselines** Throughout our experiments, we will use 3 main baselines. In the experiments on reward function inference, we use Bayesian Reward Extrapolation (B-REx) [12] as a baseline, as it generates a large set of reward function hypotheses based on a Boltzmann-rational likelihood function, and has demonstrated efficacy in RL domains. For our policy inference experiments, we compare to Contrastive Preference Learning [31] as it is a leading RLHF algorithm for arbitrary sequential tasks. We also use a naive method of learning a *set* of policies based on CPL that we call Multi-CPL. In this approach, after pretraining, we fine-tune the last layer using the CPL objective multiple times to generate a large set of reward functions.

For our language model (contextual bandit) experiments, we compare to both B-REx and Distributional Preference Learning (DPL) [49], as well as standard the standard RLHF paradigm [17], as these present a variety of approaches for generating reward models that can be used to ensure fairness across groups.

## F.1 Synthetic Experiments

We follow the experimental procedure of Siththaranjan et al. [49] in generating preferences, except modify their code such that we ensure that annotator identity is held constant for each preference. We use last layer finetuning on a neural network that is randomly initialized. We did not include any pre-training here to ensure that we are not pushing our reward models towards any modes before starting to train. We use a batch size of 2048 preferences, a step size of 0.1 and 10000 steps of MCMC for B-Rex. For POPL, we use a population size of 100 and a generation count of 100. We use a $\beta$ (confidence) value of 10, although have found that changing this value does not significantly affect B-REx's performance.

## F.2 Gridworld Model Experiments

For the Gridworld model experiments, we base our environment on the Minigrid package [16]. Demonstrations were generated by rolling out many checkpointed policies at different levels of performance, trained using Proximal Policy Optimization (PPO). Then, these demonstrations were annotated based on a high performing policy's action selection probabilities.

For MultiCPL and POPL, we use behavior cloning directly on the demonstrations for 1000 iterations with a batch size of 64 and a learning rate of 0.001 with the Adam optimizer [36] as pretraining. The model architecture was a simple convolutional neural network that takes input from the agent's view window, and has a single fully connected layer with 128 nodes to output the 7 actions from the environment. For both MultiCPL and POPL, we use last layer fine-tuning. For MultiCPL, we use the CPL objective, a learning rate of 0.001, where each model in a population of 500 models is trained for 20 iterations. For POPL, we use a learning rate of 0.2, and 1000 total steps. We sample 640 preferences every 10 iterations (as we can cache the last layer features for this examples for improved performance), and sub-sample a batch of size 64 for each step of lexicase selection. For a fair comparison between these two approaches, we approximately hold constant total wall clock time on the same hardware. Given a final population of policies generated by POPL or MultiCPL, we select the top 10 models for each hidden context class as the catered policy for that group.

### F.3 Language Model Experiments

In the LLM experiments, we assess the performance of reward learning by examining preference accuracy on the test set. To investigate vulnerabilities to jailbreak, we analyze pairs of responses to jailbreak prompts designed by Wei et al. [52] to deceive the model into giving a harmful response. We calculate the percentage of prompts where it assigns a higher reward to the jailbroken response ("jailbreak rate"). Additionally, we evaluate the reward function's ability to assess helpfulness on non-harmful prompts, *i.e.*, the reward function predicts higher rewards on the more helpful response. We compare our method to normal RLHF with an LLM-based preference model, Bayesian Reward Extrapolation (B-REx), and distributional preference learning (DPL). DPL methods predict parameters of the distribution of reward values for each response, rather than a single reward value, in order to better account for hidden context in human preferences.

For standard RLHF, we use the pre-trained LLAMA-2-7b [51] preference model by Siththaranjan et al. [49], which is fine-tuned on the HH-RLHF dataset using LoRA [33]. We implement B-REx by performing linear reward extrapolation on the last layer of the pre-trained LLAMA-2-7b preference model. Following the B-REx implementation in [11], we run 200,000 steps of MCMC with a step size of 0.05. We use a burn-in of 5000 and a skip every 20 samples to reduce auto-correlation. For POPL, we run lexicase selection for 100 generations with a population size of 1000, and randomly sample 100 reward functions in the last generation. Default settings use the mean reward across the entire set. For fairness optimization, we use the $10^{\text{th}}$ percentile of reward values across all the reward functions in the set.

Because the ranking likelihood is invariant to affine transformations of the rewards, we normalize the rewards by subtracting the median reward calculated on the training set over all the responses. This ensures that the reward values are comparable when calculating the lower quantile of rewards in risk-averse optimization.

## G Broader Societal Impacts

The proposed work on Pareto Optimal Preference Learning (POPL) aims to enhance the alignment of AI systems with diverse human values, thereby addressing critical issues of fairness and representation. By focusing on learning from human preferences with hidden context, our method seeks to ensure that AI models do not disproportionately favor or disadvantage specific groups, making them more equitable and just. This has the potential to significantly improve the societal acceptance and trust in AI systems, particularly in sensitive applications such as healthcare, education, and law enforcement, where fairness and inclusivity are critical.

However, there are potential negative societal impacts to consider. The deployment of AI systems that can cater to specific groups might inadvertently reinforce existing biases if the hidden context reflects social prejudices or discriminatory practices. Therefore, it is crucial to incorporate safeguards and robust validation mechanisms to detect and mitigate any biased outcomes. As researchers and developers, we must be vigilant about the sources of our training data and continually audit AI systems for unintended consequences.

Moreover, the computational work required for training these models can have environmental impacts, given the high-energy consumption associated with large-scale AI computations. Researchers

should consider optimizing algorithms to be more efficient and exploring the use of renewable energy sources to mitigate this impact.

By considering these factors, we aim to advance AI technologies in a direction that promotes fairness, inclusiveness, and sustainability, ensuring that they serve the broader interests of society.