Sparse Mixture-of-Experts Reward Models Learn Interpretable and Specialized Experts for Personalized Preference Modeling

Anonymous ACL submission

Abstract

Preference modeling plays a central role in reinforcement learning from human feedback (RLHF), enabling large language models (LLMs) to align with human values. However, most existing approaches assume a universal reward function, neglecting the diversity and heterogeneity of human preferences. To address this limitation without additional annotation costs, recent work has proposed learning multiple preference components from binary data and combining them to model individual preferences. Nevertheless, these components often fail to capture coherent and disentangled patterns, limiting their interpretability and effectiveness for personalization. In this work, we propose a sparse Mixture-of-Experts (MoE) reward model that encourages sparse routing and expert diversity during training on binary preference data. Across controlled and real-world experiments, sparse MoE learns interpretable routing patterns and specialized experts. It also improves test-time personalization, and post-adaptation shifts in expert weights provide a qualitative lens for analyzing how the model adapts to personalized preferences.

1 Introduction

Reinforcement learning from human feedback (RLHF) has become a central paradigm for training large language models (LLMs) (Bai et al., 2022; OpenAI, 2023; Kaufmann et al., 2025). It often relies on reward models to capture human preferences and align LLMs with human values (Ziegler et al., 2019). However, most existing reward models are trained as single, global functions, implicitly assuming a unified notion of human values (Yu et al., 2025; Wu, 2025). In reality, human preferences are inherently pluralistic and heterogeneous (Chakraborty et al., 2024; Siththaranjan et al., 2024). As a result, these models may fail to capture such diversity, limiting their effectiveness for personalized alignment (Zhang et al., 2025).

To better model heterogeneous preferences, prior work has explored incorporating additional supervision signals (Wu et al., 2023; Quan, 2024). These approaches often train multiple reward models, each corresponding to a different attribute or user group, and combine them with adaptive weights to model unseen preferences. While effective, they require costly curated annotations and rely on pre-defined preference dimensions, limiting scalability and flexibility. Synthetic data can alleviate these costs, but may suffer from reduced validity. To avoid annotation costs, recent data-driven approaches instead learn multiple components directly from binary data to infer latent preference structures (Lee et al., 2024; Luo et al., 2025; Shen et al., 2025). However, without explicit interpretability-oriented design, these components often capture entangled and incoherent patterns that are difficult for humans to understand. As a result, the learned preferences are harder to audit, and the adaptation process becomes less transparent. Moreover, such entanglement in the learned patterns can limit how effectively these components can be combined to model diverse individual preferences.

In this work, we aim to learn preference components that are both interpretable and useful for effective personalization. Specifically, we propose a sparse Mixture-of-Experts (MoE; Shazeer et al., 2017) reward model trained solely on binary preference data. As illustrated in Figure 1, we explicitly encourage sparse routing and expert diversity during training, so that different experts specialize in distinct and human-interpretable domains. These interpretable and specialized experts can then serve as disentangled building blocks for personalization, enabling efficient adaptation to individual preferences through lightweight router updates.

Controlled experiments show that sparse MoE recovers latent semantic structure in the data without direct supervision, and leverages its specialized experts for effective post-hoc preference steer-

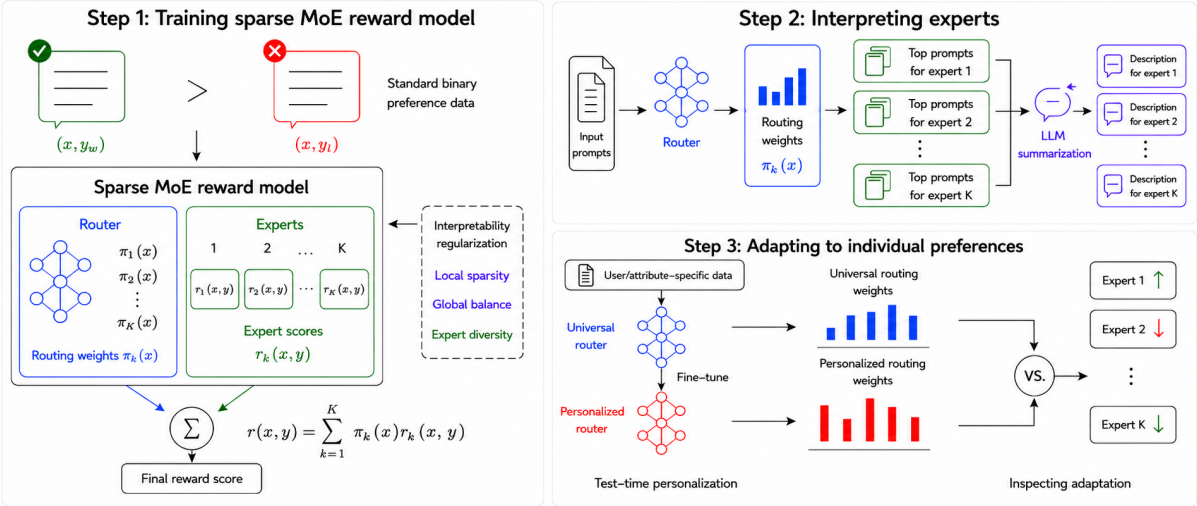

Figure 1: **Illustration of the pipeline for training a sparse MoE reward model, interpreting its experts, adapting it to individual preferences and inspecting the adaptation.** An MoE reward model is trained on standard binary preference data with interpretability regularization, and its experts can be interpreted by summarizing their top-activating examples. During test time, the router can be fine-tuned to fit personalized preferences, while changes in routing weights provide a qualitative view of how the model adapts to these preferences.

ing. On real-world preference datasets, experts remain interpretable when analyzed through their top-activating examples, and both automatic and human evaluations confirm the strong interpretability of sparse MoE. Furthermore, with only 50 adaptation examples, sparse MoE achieves a 25.81-point improvement in test-time personalization for individual preferences, substantially outperforming all baselines. Besides, post-adaptation shifts in expert weights often show plausible semantic associations with target preferences, offering a promising way to inspect model adaptation behavior.

2 Related Work

Personalization in Preference Modeling Most existing reward models are trained to reflect universal human values, yet they often fail to capture their heterogeneous nature (Yu et al., 2025; Wu, 2025). Consequently, they are not well-suited for aligning LLMs to individualized preferences (Chakraborty et al., 2024; Zhang et al., 2025). Many recent approaches address this limitation by incorporating additional supervision signals. Attribute-based approaches leverage multi-faceted human annotations over diverse attributes, such as humor and creativity, to train multiple reward models corresponding to different preference dimensions and reweight them for personalization (Wang et al., 2024a; Quan, 2024; Wang et al., 2024b; Zhou et al., 2024; Wu and Lu, 2026). In parallel, user-based approaches learn user prototypes from user-specific labels, enabling unseen users to be represented as mixtures over these prototypes with limited supervision at test time (Zhao et al., 2024; Choi et al., 2025; Liu et al., 2025; Balepur et al., 2025; Chen et al., 2025). While effective, these two types of approaches depend heavily on fine-grained human annotations or user-specific labels, both of which are costly to obtain. Synthetic data offers a partial alternative, but often suffers from limited reliability and fidelity (Ryan et al., 2025; Li et al., 2025). Besides, attribute-based methods are constrained by pre-defined attribute spaces, and thus may not capture the full diversity of real-world preferences.

To overcome these challenges, recent work has explored learning from binary preference datasets without additional supervision. Some approaches introduce latent user variables into preference modeling: Poddar et al. (2024) employ variational autoencoders, while Chidambaram et al. (2026) use expectation-maximization to infer user prototypes. Another line of work learns multiple reward components from binary data to capture latent preference patterns, and adapts their composition to reflect individual preferences. For instance, Lee et al. (2024) train an ensemble of reward models, while Luo et al. (2025) decompose response representations via principal component analysis. The work most closely related to ours is MiCRo (Shen et al., 2025), which trains an MoE reward model on standard preference data and fine-tunes the router to capture context-aware preferences. However, without explicit interpretability mechanisms, existing ap-

proaches generally fail to relate their learned components to disentangled and semantically meaningful patterns, making them suboptimal for transparent and personalized preference modeling.

Interpretability through Sparsity and MoE Despite the broad success of LLMs, understanding their internal behavior remains challenging (Bommasani et al., 2021). A key difficulty is superposition, where multiple features are compressed into a limited number of neurons, leading to entangled representations (Elhage et al., 2022). Prior work has shown that enforcing sparsity can help disentangle representations and recover more interpretable units (Bricken et al., 2023; Templeton et al., 2024). For example, sparse autoencoders (SAEs) project model representations into a higher-dimensional space under sparsity constraints, yielding more monosemantic features (Karvonen et al., 2025; Shu et al., 2025). These features can then be interpreted by using LLMs to summarize highly activating inputs. Recent studies have explored training reward functions on SAE features to improve interpretability (Movva et al., 2026; Liu et al., 2026; Zhang et al., 2026). However, these approaches often follow a two-stage pipeline, where features are first identified separately and then used for reward modeling. This separation may incur additional training costs and limit their alignment with target preferences. In contrast, our method directly builds interpretable modularity into reward models, enabling flexible preference modeling and adaptation.

In parallel, MoE architectures have emerged as an effective paradigm for scaling LLMs (Cai et al., 2025). Recent work suggests that MoE models offer a natural interpretability lens, as routing patterns often align with domain structures and experts exhibit emergent specialization. (Chaudhari et al., 2025a,b; Li et al., 2026; Fayyaz et al., 2026). Moreover, combining MoE with sparsity has been shown to further improve monosemanticity (Yang et al., 2025). However, beyond Wang et al. (2024b) and Shen et al. (2025), MoE-based preference modeling remains largely underexplored. Unlike token-level language modeling, it relies on pairwise comparisons and scalar supervision, posing distinct challenges for learning interpretable and specialized experts.

3 Sparse MoE Reward Model

In this section, we introduce the sparse MoE reward model architecture and the interpretability regularization used during training.

Preliminary In RLHF, a reward model $r_\theta(x, y) \in \mathbb{R}$ estimates human preferences over responses y given a prompt x. Training data usually consists of pairwise comparisons $\mathcal{D} = \{(x, y_w, y_l)\}$, where y_w is preferred over y_l. The reward model is typically implemented as a Bradley–Terry model (Bradley and Terry, 1952) which defines the preference probability as:

$$P(y_w \succ y_l \mid x) = \sigma\big(r_\theta(x, y_w) - r_\theta(x, y_l)\big), \quad (1)$$

where $\sigma(z) = (1 + e^{-z})^{-1}$. The reward model is trained by minimizing the negative log-likelihood of observed preferences.

MoE Reward Model Conventional models assume a single reward function underlying all preferences, which is insufficient for modeling heterogeneous preferences. Therefore, inspired by Shen et al. (2025), we adopt an MoE architecture consisting of K experts $\{r_{\theta_k}\}_{k=1}^{K}$. A router $\pi_\phi(x) \in \Delta^{K-1}$ produces a softmax distribution over experts based on the input x, where $\pi_{\phi,k}(x)$ denotes the weight assigned to expert k. In practice, we implement experts as linear heads and the router as a one-hidden-layer MLP over backbone model representations. The final reward is computed as the weighted average of expert scores:

$$r(x, y) = \sum_{k=1}^{K} \pi_{\phi,k}(x) \cdot r_{\theta_k}(x, y). \quad (2)$$

Following Shen et al. (2025), we define the reward modeling loss by marginalizing the pairwise preference probability over experts:

$$\mathcal{L}_{\mathrm{RM}} = -\frac{1}{|\mathcal{D}|} \sum_{(x,y_w,y_l)\in\mathcal{D}} \log \sum_{k=1}^{K} \big(\pi_{\phi,k}(x) \cdot \sigma(r_{\theta_k}(x, y_w) - r_{\theta_k}(x, y_l))\big). \quad (3)$$

Interpretability Regularization In addition to the reward modeling loss, we introduce regularization terms to explicitly encourage interpretable and well-structured router behavior and emergence of expert specialization:

(1) Local Sparsity: We encourage routers to output sparse weight distributions by minimizing the entropy of routing weights:

$$\mathcal{L}_{\mathrm{ls}} = \mathbb{E}_{x\sim\mathcal{D}} \left[\frac{H(\pi_\phi(x))}{\log K} \right], \qquad H(\pi) = -\sum_{k=1}^{K} \pi_k \log \pi_k. \quad (4)$$

(2) Global Balance: To prevent model collapse, we encourage balanced utilization across experts by maximizing the entropy of the average routing distribution within a batch $\mathcal{B}$:

$$\mathcal{L}_{\text{gb}} = \mathbb{E}_{\mathcal{B}\sim\mathcal{D}}\left[-\frac{H(\bar{\pi}_\phi)}{\log K}\right], \bar{\pi}_\phi = \mathbb{E}_{x\sim\mathcal{B}}[\pi_\phi(x)]. \tag{5}$$

(3) Expert Diversity: We encourage experts to capture distinct preference signals by penalizing correlations between their reward differences. Let $\Delta_k = r_{\theta_k}(x, y_w) - r_{\theta_k}(x, y_l)$, and we minimize the average pairwise Pearson correlation:

$$\mathcal{L}_{\text{div}} = \frac{2}{K(K-1)}\sum_{i<j}\text{corr}(\Delta_i, \Delta_j)^2, \tag{6}$$

where the Pearson correlations are computed over a batch $\mathcal{B}$ of preference pairs.

The overall **training objective** is defined as:

$$\mathcal{L} = \mathcal{L}_{\text{RM}} + \lambda_{\text{ls}}\mathcal{L}_{\text{ls}} + \lambda_{\text{gb}}\mathcal{L}_{\text{gb}} + \lambda_{\text{div}}\mathcal{L}_{\text{div}}, \tag{7}$$

where λ_{ls}, λ_{gb}, and λ_{div} control the strengths of local sparsity, global balance, and expert diversity regularization, respectively.

4 Experimental Setup

Models Following Shen et al. (2025), we use a strong open-source reward model GRM-Llama3.2-3B (Yang et al., 2024) as the frozen backbone and train K linear heads as experts. The router is implemented as a one-hidden-layer MLP with 128 hidden units. Unless otherwise specified, we set λ_{ls}, λ_{gb}, and λ_{div} to 0.5, 0.5, and 2.0 throughout all experiments. The number of experts is set separately for each experiment according to the complexity of the target data. Details on data processing, evaluation metrics, training configurations, and experimental setups are provided in Appendix A. Appendices B and C provide ablations on the number of experts K and each regularization term, along with practical guidelines for selecting hyperparameters.

Baselines We compare our sparse MoE reward model against a single-head reward model and two MoE baselines with the same architecture: vanilla MoE reward model and MiCRo (Shen et al., 2025). In contrast to our sparsity regularization, MiCRo encourages a more uniform routing distribution.

5 Controlled Experiments

We first evaluate sparse MoE in two controlled settings where the latent structure is known. Importantly, we focus on two complementary aspects of MoE interpretability: routing pattern interpretability and expert specialization. Without interpretable routing patterns, expert behavior is difficult to characterize; without expert specialization, router-based interpretations derived from routing behavior may reflect only input assignment patterns rather than functional expert roles, making them less faithful and less useful for model adaptation. Finally, we test whether these properties enable effective steering through routing weight interventions.

5.1 Category Recovery

We first investigate whether sparse MoE can discover latent semantic structure in the data without supervision. We use the Stanford Human Preferences (SHP; Ethayarajh et al., 2022) dataset, which includes annotations of the prompt domains. As these domains vary in granularity, we group the 16 subsets into four broader categories, namely *academic*, *professional*, *cooking*, and *fiction*.

We train a sparse MoE reward model with $K = 5$ experts on the dataset without using the domain annotations. Once the MoE model is trained, we assign each expert a category label based on the majority category among the 50 validation examples that receive the highest routing weights for that expert. We then evaluate whether the router automatically learns to assign inputs according to category structure without supervision and whether the experts specialize in their assigned categories.

Evaluation Setting In addition to accuracy, we report two model behavior metrics: average routing entropy (Eq. 4) and average expert correlation (Eq. 6). Lower values indicate sparser routing and more distinct expert behavior, respectively, which are desirable for routing pattern interpretability and expert specialization. Since the latent categories are known in this experiment, we further evaluate interpretability and specialization directly using the following metrics adapted from prior work (Badjie et al., 2025; Yang et al., 2025; Herbst et al., 2026):

- *Category Coverage*: The proportion of categories that are assigned to at least one expert. Higher scores indicate that the router better captures the underlying category structure.
- *Expert Purity*: For each expert, the proportion of its 50 top-activating test examples that belong to its assigned category. Higher scores indicate more coherent routing patterns.
- *Expert Specialization*: For each expert, the difference between its performance on its labeled

Method	Model Behavior			Interpretability		
	Accuracy (↑)	Routing Entropy (↓)	Expert Correlation (↓)	Category Coverage (↑)	Expert Purity (↑)	Expert Specialization (↑)
Single Reward	70.52	-	-	-	-	-
Vanilla MoE	**71.85**	0.998	0.523	0.50	48.40	-0.51
MiCRo	71.70	1.000	0.527	0.50	51.60	0.08
Sparse MoE	69.62	**0.200**	**0.011**	**1.00**	**85.60**	**5.77**

Table 1: **Model behavior metrics and interpretability results on SHP.** Sparse MoE reward model substantially outperforms baseline models in learning coherent and semantically meaningful routing patterns, while its experts exhibit clear behavioral specialization toward their assigned categories.

category and the average performance of all experts on the same category. Higher scores indicate stronger expert specialization in the router-assigned categories, and thus better alignment between routing patterns and expert behavior.

Results Table 1 presents the results of our sparse MoE reward model on the SHP dataset. Compared to baselines, sparse MoE shows a slight decrease in overall performance (~2%) under the structural constraints imposed by interpretability regularization. In return, the resulting routing sparsity and expert diversity yield substantially stronger expert-level interpretability and specialization.

First, sparse MoE is the only one that successfully recovers all four categories, indicating that it automatically discovers semantically meaningful routing patterns aligned with the known category structure. By contrast, vanilla MoE and MiCRo recover only two categories because of their near-uniform routing, with routing entropy close to 1 and examples assigned almost equally across experts. Moreover, sparse MoE exhibits highly coherent and monosemantic routing behavior, with 85.60% of each expert's top-activating examples coming from its assigned category. Finally, experts in sparse MoE show diverse behavior and outperform the average expert performance on their assigned categories by 5.77%, suggesting strong functional specialization. Appendix D provides the full results and more detailed analyses.

Qualitative Analysis Table 2 presents representative input prompts from SHP, together with their expert distributions produced by the sparse MoE router. We find that the router outputs highly sparse distributions, typically concentrating most probability mass on a single expert. The dominant experts are also consistent with the category labels identified above. Besides, when an input involves multiple semantic aspects, as in the last two examples, the router also assigns non-negligible weight to a secondary expert, demonstrating that it can combine expertise for more accurate predictions.

We further observe that sparse MoE discovers more fine-grained concepts than the coarse category labels. For example, among the two academic prompts, the science question is routed primarily to Expert 1, while the humanities question is routed mainly to Expert 0. This suggests that sparse MoE can capture nuanced semantic distinctions.

Takeaway: Sparse MoE reward models learn semantically meaningful routing patterns and induce expert specialization.

5.2 Attribute Steering

We next examine whether the interpretable routing patterns and expert specialization are merely correlational, or whether they can be used to steer model behavior through intervention. To this end, we train reward models with preference attributes and steer them post hoc by adjusting routing weights.

Specifically, we train MoE models on the Reasonable Preference Reversal (RPR; Pitis et al., 2024) dataset, in which each example contains two responses, each preferred under a different attribute (e.g., clarity vs. creativity). Given that RPR spans 40 diverse attributes, we train a model with $K = 10$ experts. Following Shen et al. (2025), we append a user specification of the desired attribute to each prompt during training, allowing the model to learn attribute-conditioned preferences.

At test time, we remove the attribute information and evaluate whether manipulating the routing weights can steer MoE models toward target preference attributes. For each attribute, we select a target expert whose top 100 activating validation examples contain the highest proportion of that attribute. We then steer the model by setting the target expert's weight to $w \in \{0.25, 0.5, 0.75, 1.0\}$, and rescaling the remaining weights to preserve a normalized routing distribution. We also report a random baseline that randomly selects target experts from the sparse MoE model.

Evaluation Setting We evaluate steering effectiveness using the *flip rate*. For each attribute, we

Input Prompt	Category	Expert 0 (Academic)	Expert 1 (Academic)	Expert 2 (Professional)	Expert 3 (Cooking)	Expert 4 (Fiction)
Why is 18 the maximum amount of electrons an atomic shell can hold?	Academic	0.025	**0.967**	0.000	0.007	0.000
Why is it that Eastern and Western philosophy don't appear to have interacted much, even in the ancient world, until the 19th century? Is there any evidence of cross pollination of ideas?	Academic	**0.777**	0.222	0.000	0.000	0.001
I own a condo and my downstairs neighbors have called police on me over 300 times in the last 365 days claiming frivolous noise complaints, HOA is now involved and threatening legal action. Can they force me out of my unit? No mortgage, bought in cash...I am sole occupant. (MN)	Professional	0.000	0.000	**0.989**	0.000	0.011
A friend said I should crack my garlic 10 minutes before cooking, is there a real reason for this? She said that it causes some chemical reaction to make the garlic taste better, can I just cut the garlic up and wait 10 minutes or is this pointless to begin with?	Cooking	0.000	0.123	0.000	**0.877**	0.000
[Marvel Comics] Why did Norse paganism fade away, when Thor was actually present on Earth, fighting monsters and performing miracles?	Fiction	0.178	0.000	0.000	0.000	**0.822**

Table 2: **Input prompts from SHP together with the routing weights assigned to each expert.** Parentheses under each expert indicate the category associated with that expert. The router produces sparse and semantically meaningful routing distributions based on the input semantics.

first identify examples that are misclassified when the target expert is removed. The flip rate is then computed as the proportion of these examples that switch to the preferred response after assigning a specified weight to the target expert. Finally, we average the flip rates across all target weights and attributes to obtain an overall score.

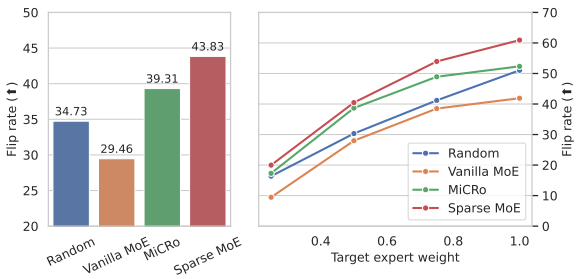

Figure 2: **Attribute steering results on RPR.** Sparse MoE achieves the strongest steering performance. The improvements are statistically significant ($\alpha < 0.05$).

Results Figure 2 presents the average flip rate of each model (left) and the flip rate as a function of the target expert weight (right). Benefiting from improved interpretability and expert specialization, sparse MoE outperforms the strongest baseline by 4.52% in average flip rate. Specifically, behaviorally distinct experts make routing weight interventions more influential on model preferences, as reflected by the high flip rate of sparse MoE even under random expert steering. Moreover, the enhanced interpretability helps identify suitable target experts and enables more targeted steering. Notably, sparse MoE reaches a flip rate of over 60% with $w = 1.0$, further highlighting its effectiveness.

We additionally analyze undesired flips in Appendix E and show that steering the target experts in sparse MoE does not increase unintended changes to originally correct predictions. These results further confirm the effectiveness and robustness of attribute steering in the sparse MoE model.

Takeaway: Interpretable routing patterns and expert specialization in sparse MoE reward models support targeted interventional control over model behavior by tuning routing weights.

6 Real-World Binary Preference Data

In this section, we evaluate the interpretability and personalization capability of sparse MoE reward models in a real-world setting using standard binary preference data. In particular, we train MoE models on the preference-700K dataset (700K; Dong et al., 2024) and use the dataset for expert interpretation as well. To capture diverse patterns in the real-world preference data, we use $K = 20$ experts and set $\lambda_{\text{gb}} = 1.0$ to prevent routing collapse.

6.1 Interpretability Evaluation

Following prior work on automated feature interpretation (Shu et al., 2025), we interpret experts by summarizing their top-activating examples. Concretely, for each expert, we collect 20 validation examples with the largest routing weights and prompt an LLM (Qwen3.6-27B) to summarize their shared features as a natural language description. This yields a router-based interpretation of each expert. Table 3 presents some examples of the descriptions. Results for the random baseline are computed by shuffling the examples while keeping the sparse MoE router weights fixed.

Evaluation Setting As in the controlled experiments, we evaluate MoE interpretability from two complementary perspectives: whether routing patterns are interpretable, and whether these interpretations align with expert specialization.

ID	Natural Language Description	Description Fidelity (↑)	Expert Specialization (↑)
0	asks for advice, feedback, or discussion on a personal, professional, or academic challenge	0.43	22.08
6	asks to solve a procedural task involving numerical or list manipulation	0.51	25.98
8	asks for a summary or extraction of information from a provided text	0.45	19.35
10	asks for a direct answer to a multiple-choice or short-answer question	0.36	20.56
13	requests assistance with illegal, harmful, or sexually explicit activities	0.62	3.35
15	asks to solve a competitive mathematics problem	0.42	32.00

Table 3: **Natural language descriptions of selected experts in the sparse MoE model.** Most experts specialize in a particular task, topic, or input format. Descriptions of all experts are provided in Appendix D.

For routing pattern interpretability, we measure *description fidelity* following Movva et al. (2026). For each expert, we sample 200 test examples and instruct an LLM (Qwen3.6-35B-A3B) to judge whether each example matches the expert's natural language description. We then compute the Pearson correlation between these binary relevance judgments and the corresponding router activations. A higher description fidelity score indicates that the model learns coherent and interpretable routing patterns that can be described in natural language.

For expert specialization, we report the *expert specialization* score defined in Section 5.1. Since no ground truth clustering is available, we compute each expert's accuracy on its top 10% highest-activating examples and compare it with the average accuracy of all experts on the same examples. The resulting performance gap is reported as the specialization score. A higher expert specialization score indicates that the router-based interpretation faithfully reflects the expert's functional role, rather than merely describing an assignment pattern.

Method	Accuracy (↑)	Description Fidelity (↑)	Expert Specialization (↑)
Single Reward	83.52	-	-
Vanilla MoE	**84.82**	0.29	0.09
MiCRo	84.68	0.15	0.04
Random	-	0.01	-0.04
Sparse MoE	83.78	**0.38**	**15.13**

Table 4: **Interpretability results on 700K.** Sparse MoE exhibits the most interpretable routing patterns, which are also consistent with its expert specialization.

Results Table 4 shows that our sparse MoE model achieves substantially better interpretability than baselines, while maintaining competitive predictive accuracy. It obtains the highest description fidelity, indicating that the routing patterns are coherent and interpretable. It also achieves a considerably higher expert specialization score, showing that the experts perform better on examples matching their natural language descriptions. By contrast, vanilla MoE and MiCRo exhibit near-random expert specialization. Together, these results suggest that sparse MoE learns interpretable and functionally specialized experts whose strengths align with the routing patterns. Examples in Table 3 further show that inputs are grouped by highly human-interpretable features, such as topics or tasks.

Human Evaluation We additionally conduct a human study to compare expert interpretation quality across models. For each model, we sample 5 experts and present their 10 highest-activating examples to 5 annotators. Annotators first rate whether the examples exhibit coherent patterns (*pattern coherence*), and then rate how well the LLM-generated description captures the shared pattern (*description quality*), both on a 1–5 scale. Additional details are provided in Appendix F.

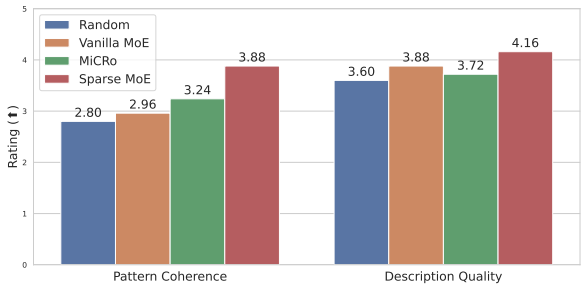

Figure 3: **Human evaluation of model interpretability.** Human annotators find that sparse MoE has the most interpretable routing patterns. The improvements in pattern coherence are statistically significant ($\alpha < 0.05$).

Figure 3 further shows that the sparse MoE model is more interpretable than baselines. This improvement mainly stems from the ability of sparse MoE to route inputs based on more coherent patterns (i.e., higher pattern coherence score), as a result of our explicit interpretability regularization. It also facilitates the generation of better natural language descriptions of the experts by LLMs.

Takeaway: Sparse MoE reward models learn interpretable and specialized experts on real-world preference data as well.

6.2 Personalization Evaluation

We conjecture that the disentangled experts in sparse MoE models support effective and interpretable personalization. To evaluate this, we

Method	Linguistic Creativity	User Friendliness	Humor & Entertainment	Scientific Rigor	Narrative & Storytelling	Creativity & Originality	Factual Accuracy	Overall
Single Reward	31.73	39.33	30.16	76.19	34.17	18.52	85.92	45.15
HyRe	37.50 (+6.73)	46.44 (+9.36)	32.14 (+3.57)	78.17 (+0.79)	40.0 (+7.50)	31.02 (+14.35)	81.22 (-0.47)	49.5 (+5.98)
Vanilla MoE	42.31 (+11.54)	44.19 (+8.23)	36.51 (+6.75)	81.74 (+3.17)	51.67 (+19.17)	28.70 (+10.64)	**89.20 (+7.51)**	53.47 (+9.57)
MiCRo	40.39 (+9.62)	42.32 (+6.36)	40.48 (+10.72)	81.35 (+2.78)	51.25 (+18.75)	28.24 (+10.18)	88.73 (+7.04)	53.25 (+9.35)
Sparse MoE	**74.04 (+43.27)**	**55.06 (+14.61)**	**80.56 (+50.80)**	**82.54 (+7.54)**	**68.75 (+28.75)**	**60.19 (+31.02)**	83.57 (+4.70)	**72.1 (+25.81)**

Table 5: **Personalization results on RPR.** Scores in parentheses denote the change after adaptation. Higher scores are better. Sparse MoE demonstrates the strongest personalization performance among all compared models.

conduct test-time personalization experiments by adapting the routers of MoE models to new attributes or users using only a small adaptation set.

Evaluation Setting Following Luo et al. (2025), we use RPR for attribute-level personalization. For each attribute, we use $n = 50$ training examples for adaptation and evaluate on all test examples (see Appendix G for results under different adaptation set sizes). During adaptation, we freeze the experts of MoE models trained on 700K and fine-tune the router using the Hedge algorithm proposed by Shen et al. (2025). As an additional baseline, we include HyRe (Lee et al., 2024), an ensemble method that trains 20 reward models and reweights them at test time. Unlike MoE models, HyRe uses input-independent ensemble weights. We report post-adaptation performance and improvements over the corresponding non-adapted models, averaged over three independently sampled adaptation sets.

Results Table 5 presents the attribute-level personalization results on RPR. Our sparse MoE reward model achieves the strongest performance in modeling target attributes, improving overall accuracy by 25.81 points, while no other model achieves an improvement greater than 10. The gains are especially large when the original model performs poorly under the target attribute, suggesting that sparse MoE can effectively adapt to preferences that substantially deviate from universal preference patterns. Results for all tested attributes are provided in Appendix D.

For completeness, we evaluate user-level adaptation on PersonalLLM (Zollo et al., 2025), where sparse MoE outperforms baseline models in adapting to individualized preferences as well (Appendix H). We also conduct a post-training experiment with the adapted reward models and show that they can effectively guide model alignment toward personalized values (Appendix I).

Inspecting Preference Adaptation Leveraging the interpretability and specialization of experts in sparse MoE, we further analyze the model's adaptation to individual preferences by examining changes in expert weights after adaptation. Table 6 reports, for a subset of attributes, the expert whose weight increases the most after adaptation. We observe that the upweighted experts often show plausible semantic associations with the target attributes. Overall, post-adaptation expert weight shifts provide a useful diagnostic signal for inspecting preference adaptation.

Attribute	Most Upweighted Expert
Factual Accuracy	asks for a summary or extraction of information from a provided text
Pedagogical Effectiveness	asks for advice, feedback, or discussion on a personal, professional, or academic challenge
Innovativeness	asks a question containing a factual error, logical fallacy, or absurd premise

Table 6: **Experts most strongly upweighted after adaptation for selected attributes.** The experts align semantically with the adapted attributes in an interpretable way.

Takeaway: The interpretable and specialized experts in sparse MoE reward models facilitate effective test-time personalization and provide insight into how models adapt to individual preferences.

7 Conclusion

In this work, we address the challenge of personalized preference modeling by proposing a sparse MoE reward model, which encourages sparse routing and expert diversity during training on binary preference data. Experiments in both controlled and real-world settings demonstrate that sparse MoE reward models learn interpretable and specialized experts. These disentangled experts also support effective test-time personalization in low-data settings, and their post-adaptation weight shifts offer a qualitative way to inspect how the model adjusts to individual preferences.

Limitations

Our study has certain limitations that should be acknowledged. First, although we demonstrate improved interpretability and personalization capability in sparse MoE reward models, effective training still depends on the selection of multiple hyperparameters. This could introduce additional complexity and tuning effort in practice. To mitigate this issue, we provide detailed ablation studies on the effect of the number of experts and each regularization term in Appendix B and Appendix C, and offer practical guidelines for hyperparameter selection. Second, as our method is largely data-driven, the learned expert patterns are inherently dependent on the latent structure of the training data and are difficult to predict beforehand. As a result, the emergence of specific patterns or expert behavior of interest is usually not guaranteed. Finally, the interpretability and personalization benefits of sparse MoE models come with a minor drop in universal preference modeling performance. Therefore, they may be less suitable for settings where accurately capturing global preferences is the primary objective.

Ethics Statement

While our work aims to address the challenge of modeling heterogeneous human values, the proposed models also introduce potential risks of dual use, as they can be adapted to pverfit to individual preferences. As a result, universal principles such as fairness and safety constraints could potentially be bypassed, and harmful behaviors or addictive usage patterns may be reinforced. In addition, our framework makes certain preference patterns more interpretable, which may increase the risk of privacy leakage or misuse of sensitive user information. Therefore, we strongly recommend that any real-world deployment of such systems undergo careful ethical review and oversight by appropriate ethics committees.

References

Bakary Badjie, José Cecílio, and António Casimiro. 2025. Double-stage feature-level clustering-based mixture of experts framework. *CoRR*, abs/2503.09504.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, and 12 others. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *CoRR*, abs/2204.05862.

Nishant Balepur, Vishakh Padmakumar, Fumeng Yang, Shi Feng, Rachel Rudinger, and Jordan Lee Boyd-Graber. 2025. Whose boat does it float? improving personalization in preference tuning via inferred user personas. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3371–3393, Vienna, Austria. Association for Computational Linguistics.

Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ B. Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, Shyamal Buch, Dallas Card, Rodrigo Castellon, Niladri S. Chatterji, Annie S. Chen, Kathleen Creel, Jared Quincy Davis, Dorottya Demszky, and 34 others. 2021. On the opportunities and risks of foundation models. *CoRR*, abs/2108.07258.

Avinandan Bose, Zhihan Xiong, Yuejie Chi, Simon Shaolei Du, Lin Xiao, and Maryam Fazel. 2025. Lore: Personalizing LLMs via low-rank reward modeling. In *Second Conference on Language Modeling*.

Ralph Allan Bradley and Milton E Terry. 1952. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, and 6 others. 2023. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*.

Weilin Cai, Juyong Jiang, Fan Wang, Jing Tang, Sunghun Kim, and Jiayi Huang. 2025. A survey on mixture of experts in large language models. *IEEE Transactions on Knowledge and Data Engineering*, 37(7):3896–3915.

Souradip Chakraborty, Jiahao Qiu, Hui Yuan, Alec Koppel, Dinesh Manocha, Furong Huang, Amrit Bedi, and Mengdi Wang. 2024. MaxMin-RLHF: Alignment with diverse human preferences. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 6116–6135. PMLR.

Marmik Chaudhari, Idhant Gulati, Nishkal Hundia, Pranav Karra, and Shivam Raval. 2025a. Moe lens - an expert is all you need. In *Sparsity in LLMs (SLLM): Deep Dive into Mixture of Experts, Quantization, Hardware, and Inference*.

Marmik Chaudhari, Jeremi Nuer, and Rome Thorstenson. 2025b. Superposition in mixture of experts. In *Mechanistic Interpretability Workshop at NeurIPS 2025*.

Daiwei Chen, Yi Chen, Aniket Rege, Zhi Wang, and Ramya Korlakai Vinayak. 2025. PAL: Sample-efficient personalized reward modeling for pluralistic alignment. In *The Thirteenth International Conference on Learning Representations*.

Keertana Chidambaram, Karthik Vinay Seetharaman, and Vasilis Syrgkanis. 2026. Direct preference optimization with unobserved preference heterogeneity: The necessity of ternary preferences. In *The 29th International Conference on Artificial Intelligence and Statistics*.

Youngbin Choi, Seunghyuk Cho, Minjong Lee, Moonjeong Park, Yesong Ko, Jungseul Ok, and Dongwoo Kim. 2025. CoPL: Collaborative preference learning for personalizing LLMs. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 12875–12893, Suzhou, China. Association for Computational Linguistics.

Hanze Dong, Wei Xiong, Bo Pang, Haoxiang Wang, Han Zhao, Yingbo Zhou, Nan Jiang, Doyen Sahoo, Caiming Xiong, and Tong Zhang. 2024. RLHF workflow: From reward modeling to online RLHF. *Transactions on Machine Learning Research*.

Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, Roger Grosse, Sam McCandlish, Jared Kaplan, Dario Amodei, Martin Wattenberg, and Christopher Olah. 2022. Toy models of superposition. *Transformer Circuits Thread*.

Kawin Ethayarajh, Yejin Choi, and Swabha Swayamdipta. 2022. Understanding dataset difficulty with $\mathcal{V}$-usable information. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 5988–6008. PMLR.

Mohsen Fayyaz, Ali Modarressi, Hanieh Deilamsalehy, Franck Dernoncourt, Ryan A. Rossi, Trung Bui, Hinrich Schuetze, and Nanyun Peng. 2026. Steering moe LLMs via expert (de)activation. In *The Fourteenth International Conference on Learning Representations*.

Jeremy Herbst, Jae Hee Lee, and Stefan Wermter. 2026. The expert strikes back: Interpreting mixture-of-experts language models at expert level. *CoRR*, abs/2604.02178.

Adam Karvonen, Can Rager, Johnny Lin, Curt Tigges, Joseph Isaac Bloom, David Chanin, Yeu-Tong Lau, Eoin Farrell, Callum Stuart McDougall, Kola Ayonrinde, Demian Till, Matthew Wearden, Arthur Conmy, Samuel Marks, and Neel Nanda. 2025. SAEBench: A comprehensive benchmark for sparse autoencoders in language model interpretability. In *Forty-second International Conference on Machine Learning*.

Timo Kaufmann, Paul Weng, Viktor Bengs, and Eyke Hüllermeier. 2025. A survey of reinforcement learning from human feedback. *Transactions on Machine Learning Research*. Survey Certification.

Yoonho Lee, Jonathan Williams, Henrik Marklund, Archit Sharma, Eric Mitchell, Anikait Singh, and Chelsea Finn. 2024. Test-time alignment via hypothesis reweighting. *CoRR*, abs/2412.08812.

Shuyue Stella Li, Melanie Sclar, Hunter Lang, Ansong Ni, Jacqueline He, Puxin Xu, Andrew Cohen, Chan Young Park, Yulia Tsvetkov, and Asli Celikyilmaz. 2025. Prefpalette: Personalized preference modeling with latent attributes. In *Second Conference on Language Modeling*.

Wengang Li, Lingqi Zhang, Toshio Endo, and Mohamed Wahib. 2026. Understanding cross-layer contributions to mixture-of-experts routing in LLMs. In *The Fourteenth International Conference on Learning Representations*.

Dengcan Liu, Jiahao Li, Zheren Fu, Yi Tu, Jiajun Li, Zhendong Mao, and Yongdong Zhang. 2026. Sparserm: A lightweight preference modeling with sparse autoencoder. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(38):32051–32059.

Jiongnan Liu, Yutao Zhu, Shuting Wang, Xiaochi Wei, Erxue Min, Yu Lu, Shuaiqiang Wang, Dawei Yin, and Zhicheng Dou. 2025. LLMs + persona-plug = personalized LLMs. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9373–9385, Vienna, Austria. Association for Computational Linguistics.

Feng Luo, Rui Yang, Hao Sun, Chunyuan Deng, Jiarui Yao, Jingyan Shen, Huan Zhang, and Hanjie Chen. 2025. Rethinking diverse human preference learning through principal component analysis. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 19857–19870, Vienna, Austria. Association for Computational Linguistics.

Rajiv Movva, Smitha Milli, Sewon Min, and Emma Pierson. 2026. What's in my human feedback? learning interpretable descriptions of preference data. In *The Fourteenth International Conference on Learning Representations*.

OpenAI. 2023. GPT-4 technical report. *CoRR*, abs/2303.08774.

Silviu Pitis, Ziang Xiao, Nicolas Le Roux, and Alessandro Sordoni. 2024. Improving context-aware preference modeling for language models. In *Advances in Neural Information Processing Systems*, volume 37, pages 70793–70827. Curran Associates, Inc.

Sriyash Poddar, Yanming Wan, Hamish Ivison, Abhishek Gupta, and Natasha Jaques. 2024. Personalizing reinforcement learning from human feedback with variational preference learning. In *Advances in Neural Information Processing Systems*, volume 37, pages 52516–52544. Curran Associates, Inc.

Shanghaoran Quan. 2024. DMoERM: Recipes of mixture-of-experts for effective reward modeling. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 7006–7028, Bangkok, Thailand. Association for Computational Linguistics.

Michael J. Ryan, Omar Shaikh, Aditri Bhagirath, Daniel Frees, William Held, and Diyi Yang. 2025. SynthesizeMe! inducing persona-guided prompts for personalized reward models in LLMs. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8045–8078, Vienna, Austria. Association for Computational Linguistics.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *CoRR*, abs/2402.03300.

Noam Shazeer, *Azalia Mirhoseini, *Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations*.

Jingyan Shen, Jiarui Yao, Rui Yang, Yifan Sun, Feng Luo, Rui Pan, Tong Zhang, and Han Zhao. 2025. MiCRo: Mixture modeling and context-aware routing for personalized preference learning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 17447–17463, Suzhou, China. Association for Computational Linguistics.

Dong Shu, Xuansheng Wu, Haiyan Zhao, Daking Rai, Ziyu Yao, Ninghao Liu, and Mengnan Du. 2025. A survey on sparse autoencoders: Interpreting the internal mechanisms of large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 1690–1712, Suzhou, China. Association for Computational Linguistics.

Anand Siththaranjan, Cassidy Laidlaw, and Dylan Hadfield-Menell. 2024. Distributional preference learning: Understanding and accounting for hidden context in RLHF. In *The Twelfth International Conference on Learning Representations*.

Adly Templeton, Tom Conerly, Jonathan Marcus, Jack Lindsey, Trenton Bricken, Brian Chen, Adam Pearce, Craig Citro, Emmanuel Ameisen, Andy Jones, Hoagy Cunningham, Nicholas L Turner, Callum McDougall, Monte MacDiarmid, C. Daniel Freeman, Theodore R. Sumers, Edward Rees, Joshua Batson, Adam Jermyn, and 3 others. 2024. Scaling monosemanticity: Extracting interpretable features from claude 3 sonnet. *Transformer Circuits Thread*.

Haoxiang Wang, Yong Lin, Wei Xiong, Rui Yang, Shizhe Diao, Shuang Qiu, Han Zhao, and Tong Zhang. 2024a. Arithmetic control of LLMs for diverse user preferences: Directional preference alignment with multi-objective rewards. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8642–8655, Bangkok, Thailand. Association for Computational Linguistics.

Haoxiang Wang, Wei Xiong, Tengyang Xie, Han Zhao, and Tong Zhang. 2024b. Interpretable preferences via multi-objective reward modeling and mixture-of-experts. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 10582–10592, Miami, Florida, USA. Association for Computational Linguistics.

Xiaobao Wu. 2025. A comprehensive survey on learning from rewards for large language models: Reward models and learning strategies. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 17847–17875, Suzhou, China. Association for Computational Linguistics.

Xinle Wu and Yao Lu. 2026. Reward model routing in alignment. In *The Fourteenth International Conference on Learning Representations*.

Zeqiu Wu, Yushi Hu, Weijia Shi, Nouha Dziri, Alane Suhr, Prithviraj Ammanabrolu, Noah A. Smith, Mari Ostendorf, and Hannaneh Hajishirzi. 2023. Fine-grained human feedback gives better rewards for language model training. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Rui Yang, Ruomeng Ding, Yong Lin, Huan Zhang, and Tong Zhang. 2024. Regularizing hidden states enables learning generalizable reward model for llms. In *Advances in Neural Information Processing Systems*.

Xingyi Yang, Constantin Venhoff, Ashkan Khakzar, Christian Schroeder de Witt, Puneet K. Dokania, Adel Bibi, and Philip Torr. 2025. Mixture of experts made intrinsically interpretable. In *Forty-second International Conference on Machine Learning*.

Rui Yu, Shenghua Wan, Yucen Wang, Chen-Xiao Gao, Le Gan, Zongzhang Zhang, and De-Chuan Zhan. 2025. Reward models in deep reinforcement learning: A survey. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence, IJCAI-25*, pages 10807–10816. International Joint Conferences on Artificial Intelligence Organization. Survey Track.

Michael JQ Zhang, Zhilin Wang, Jena D. Hwang, Yi Dong, Olivier Delalleau, Yejin Choi, Eunsol Choi, Xiang Ren, and Valentina Pyatkin. 2025. Diverging preferences: When do annotators disagree and do models know? In *Forty-second International Conference on Machine Learning*.

Shuyi Zhang, Wei Shi, Sihang Li, Jiayi Liao, Tao Liang, Hengxing Cai, and Xiang Wang. 2026. Interpretable reward model via sparse autoencoder. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(41):34808–34816.

Siyan Zhao, John Dang, and Aditya Grover. 2024. Group preference optimization: Few-shot alignment of large language models. In *The Twelfth International Conference on Learning Representations*.

Zhanhui Zhou, Jie Liu, Jing Shao, Xiangyu Yue, Chao Yang, Wanli Ouyang, and Yu Qiao. 2024. Beyond one-preference-fits-all alignment: Multi-objective direct preference optimization. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 10586–10613, Bangkok, Thailand. Association for Computational Linguistics.

Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul F. Christiano, and Geoffrey Irving. 2019. Fine-tuning language models from human preferences. *CoRR*, abs/1909.08593.

Thomas P Zollo, Andrew Wei Tung Siah, Naimeng Ye, Ang Li, and Hongseok Namkoong. 2025. PersonalLLM: Tailoring LLMs to individual preferences. In *The Thirteenth International Conference on Learning Representations*.

A Details on Experimental Setup

SHP The SHP dataset is curated from Reddit posts and contains 18 subsets featuring different domains. Following the recommended practice from the dataset curator, we use only examples where both chosen and rejected responses receive more than 3 likes. For training and evaluation, we exclude the *explainlikeimfive* and *changemyview* subsets, as their examples contain special prefixes that can easily leak the domain information. For examples replying to the same input, we only sample one pair of responses for evaluation. In the end, we use 36000/4000/1497 examples for train/val/test sets, respectively. The domains, their groupings, and numbers in the test set can be found in Table 7.

RPR Since each example in RPR is paired with two attributes, under which the preference is swapped, we convert each example to two attribute-conditioned examples with context during training. Specifically, following Shen et al. (2025), we append the criteria corresponding to target attributes to the inputs and assign labels accordingly. We use the complete training set with all 40 attributes for reward model training, resulting in 18300/2034 train/validation examples. Some examples of attributes and their corresponding context-dependent criteria are provided below:

- Linguistic Creativity: The response should demonstrate linguistic creativity by incorporating poetry or quotations attributed to the Sufi figure, enhancing the cultural and emotional understanding of their teachings.
- Detail and Elaboration: Provides a detailed step-by-step guide on how to build the app.
- Clarity and Conciseness: Provides a precise and succinct explanation of the bug, focusing on its technical aspects and why it violates design guidelines.

For evaluation robustness, we conduct attribute steering and personalization experiments on the 17 attributes that appear over 50 times in the test set. These attributes are: *Linguistic Creativity, User Friendliness, Humor and Entertainment Value, Scientific Rigor, Narrative and Storytelling Quality, Creativity and Originality, Factual Accuracy, Innovativeness, Pedagogical Effectiveness, Economic Feasibility, Technical Complexity, Interdisciplinary Approach, Empathy and Emotional Intelligence, Contextual Relevance, Clarity and Conciseness, User Experience, Practical Application*.

We perform paired t-tests on flip rates across all tested attributes to assess the statistical significance of the improvements.

700K We randomly sample 36000/4000/5000 examples to create the train/validation/test sets for 700K.

License We use publicly available datasets and models for research purposes and cite the original sources. SHP is derived from Reddit data under the Reddit API Terms of Use and does not provide a standard open-source license. RPR is released under CDLA-Permissive-2.0. For preference-700K, we could not identify an explicit license on the dataset card; therefore, we report the license of the official accompanying RLHF Workflow repository, which is released under Apache-2.0. For PersonalLLM, although the Hugging Face metadata lists CC BY 4.0, the paper states that the dataset is released under CC BY-NC 4.0; we therefore conservatively treat it as non-commercial research data. The GRM-Llama3.2-3B reward model and Qwen3.6 models used for expert interpretation and evaluation are released under Apache-2.0. We do not redistribute the original datasets or model weights.

Category	Domain	# Examples in the Test Set
Academic	askphysics, askscience, askengineers, askhistorians, askphilosophy, asksocialscience, askanthropology	617
Professional	askdocs, askvet, askhr, legaladvice, askacademia, askcarguys	384
Cooking	askbaking, askculinary	256
Fiction	askscience fiction	240

Table 7: **Domain composition and number of test examples for each category in SHP.**

Training Details For a fair comparison, we use the same training configuration for all models and baselines. We set the learning rate to 0.002/0.0005 for experts and router parameters, respectively, with a default AdamW optimizer and 5% warmup steps. A dropout rate of 0.2 and jitter noise of 0.25 are applied to the router in MoE models to avoid routing collapse.

We train the models for 3 epochs with a batch size of 16 on one H100 GPU, as we find sufficiently long training is necessary for MoE to develop reasonable specialization.

Evaluation Metrics We provide mathematical descriptions of the evaluation metrics we used in the experiments.

• **Category Coverage** Let $\mathcal{C}$ denote the set of ground truth categories. For each expert k, we assign a category label $\hat{c}_k$ based on the majority category among its top-activating validation examples. Category coverage is defined as:

$$\text{Coverage} = \frac{|\{c \in \mathcal{C} : \exists k, \hat{c}_k = c\}|}{|\mathcal{C}|}. \quad (8)$$

• **Expert Purity** Let $\ell(x) \in \mathcal{C}$ denote the ground truth category of input x. For each expert k, let $\mathcal{T}_k^{(n)}$ be the set of n test examples with the largest routing weights $\pi_{\phi,k}(x)$. Given the category label $\hat{c}_k$ assigned to expert k, expert purity is defined as:

$$\text{Purity}_k = \frac{1}{|\mathcal{T}_k^{(n)}|} \sum_{x \in \mathcal{T}_k^{(n)}} \mathbf{1}\{\ell(x) = \hat{c}_k\}. \quad (9)$$

The overall expert purity score is averaged across experts:

$$\text{Purity} = \frac{1}{K} \sum_{k=1}^{K} \text{Purity}_k. \quad (10)$$

• **Expert Specialization** Let $\mathcal{D}_c$ denote the set of test examples belonging to category c, and let $\text{Acc}_j(c)$ denote the accuracy of expert j on $\mathcal{D}_c$. For each expert k with assigned category $\hat{c}_k$, we define its specialization score as the difference between its accuracy on $\hat{c}_k$ and the average accuracy of all experts on the same category:

$$\text{Spec}_k = \text{Acc}_k(\hat{c}_k) - \frac{1}{K} \sum_{j=1}^{K} \text{Acc}_j(\hat{c}_k). \quad (11)$$

In settings without ground truth categories, we instead evaluate specialization on the examples most strongly associated with each expert. For each expert k, let $\mathcal{T}_k^{(m\%)}$ denote the top m% test examples with the largest routing weights $\pi_{\phi,k}(x)$. Let $\text{Acc}_j(\mathcal{S})$ denote the accuracy of expert j on a set of examples $\mathcal{S}$. We define the specialization score of expert k as:

$$\text{Spec}_k = \text{Acc}_k(\mathcal{T}_k^{(m\%)}) - \frac{1}{K} \sum_{j=1}^{K} \text{Acc}_j(\mathcal{T}_k^{(m\%)}). \quad (12)$$

The overall expert specialization score is averaged across experts:

$$\text{Spec} = \frac{1}{K} \sum_{k=1}^{K} \text{Spec}_k. \quad (13)$$

• **Flip Rate** Let $\mathcal{A}$ denote the set of evaluated attributes and $\mathcal{W}$ denote the set of target expert weights. For each attribute $a \in \mathcal{A}$, let $\mathcal{D}_a$ be its test set and let k_a be the target expert selected for this attribute. Given a target weight $w \in \mathcal{W}$, we define the intervened routing distribution $\tilde{\pi}_\phi^{(k_a,w)}(x)$ by setting the weight of expert k_a to w and rescaling the remaining expert weights to sum to $1 - w$.

Let $\hat{y}^{(k_a,w)}(x)$ denote the response selected by the model under this intervention, and let $y_a^+(x)$ denote the response preferred under attribute a. We first identify the examples that are misclassified when the target expert is removed:

$$\mathcal{M}_a = \left\{ x \in \mathcal{D}_a : \hat{y}^{(k_a,0)}(x) \neq y_a^+(x) \right\}. \quad (14)$$

The flip rate for attribute a under target expert weight w is:

$$\text{FR}(a, w) = \frac{\sum_{x \in \mathcal{M}_a} \mathbf{1}\left\{\hat{y}^{(k_a,w)}(x) = y_a^+(x)\right\}}{|\mathcal{M}_a|}. \quad (15)$$

The overall flip rate is averaged over all evaluated attributes and target expert weights:

$$\mathrm{FR} = \frac{1}{|\mathcal{A}|\,|\mathcal{W}|} \sum_{a \in \mathcal{A}} \sum_{w \in \mathcal{W}} \mathrm{FR}(a, w). \tag{16}$$

• **Undesired Flip Rate** For each attribute a, we define the set of initially correct examples under target expert removal as:

$$\mathcal{C}_a = \left\{ x \in \mathcal{D}_a : \hat{y}^{(k_a,0)}(x) = y_a^+(x) \right\}. \tag{17}$$

The undesired flip rate under target weight w is then:

$$\mathrm{UFR}(a, w) = \frac{\sum_{x \in \mathcal{C}_a} \mathbf{1}\left\{ \hat{y}^{(k_a,w)}(x) \neq y_a^+(x) \right\}}{|\mathcal{C}_a|}. \tag{18}$$

• **Description Fidelity** For each expert k, let d_k denote its natural language description, and let $\mathcal{S}_k$ be the set of sampled test examples used for evaluation. For each example $x \in \mathcal{S}_k$, we ask an LLM to judge whether x matches d_k, producing a binary relevance judgment $z_k(x) \in \{0, 1\}$. We then compute the Pearson correlation between these judgments and the corresponding router activations $\pi_{\phi,k}(x)$:

$$\mathrm{DF}_k = \mathrm{corr}_{x \in \mathcal{S}_k} \left(z_k(x), \pi_{\phi,k}(x) \right). \tag{19}$$

The overall description fidelity score is averaged across experts:

$$\mathrm{DF} = \frac{1}{K} \sum_{k=1}^{K} \mathrm{DF}_k. \tag{20}$$

Expert Interpretation When computing the description fidelity of expert interpretations, since only a small fraction of examples strongly activate any given expert, we stratify the sample by drawing 100 examples from the top 10% of activations and 100 examples from the remaining 90%.

The prompts that we use for interpreting experts and evaluating interpretability are adapted from Movva et al. (2026) and are provided in Figure 4 and Figure 5.

Test-Time Personalization We use the test-time weight adjustment method from Lee et al. (2024) for HyRe and implement the Hedge algorithm from Shen et al. (2025) for fine-tuning routers in MoE models. For Hedge, we use $\tau = 0.001$, following the recommendation of the original paper. All models are trained for 10 epochs using a batch size of 32 and a learning rate of 0.01 for adaptation.

B Effect of the Number of Experts

We study the effect of the number of experts by training sparse MoE models with $K \in \{10, 20, 30, 40, 50\}$ on 700K and evaluating their interpretability. We conduct this analysis on the real-world dataset rather than the controlled settings, because the pre-defined labels in the controlled experiments capture only coarse latent structures and may not fully reflect the finer-grained patterns discovered by the model.

Figure 6 shows how description fidelity and expert specialization change with the number of experts. Description fidelity increases from $K = 10$ to $K = 30$, where it reaches its peak, and then remains relatively stable. This suggests that increasing the number of experts initially helps the router discover more semantically coherent input clusters, but adding experts beyond this point provides limited additional benefit. We interpret this peak as reflecting the effective granularity of input patterns that can be reliably supported by the data, training objective, and interpretation procedure.

By contrast, expert specialization decreases steadily as K increases. This suggests that larger expert counts may fragment the preference learning signal and weaken the behavioral specialization of individual expert heads. Together, these trends indicate a trade-off between semantic interpretability and functional specialization: router assignments can remain semantically interpretable at larger K, while the corresponding expert heads become less behaviorally specialized.

C Ablation Study of Interpretability Regularization

To better understand the effect of each interpretability regularization term on model behavior, we conduct a comprehensive ablation study using a grid search over $\lambda_{ls} \in [0.0, 0.5, 1.0]$, $\lambda_{gb} \in [0.0, 0.5, 1.0]$, and $\lambda_{div} \in [0.0, 1.0, 2.0]$ on SHP with $K = 5$ experts. The resulting task performance, routing pattern, expert behavior, and interpretability metrics are reported in Table 8.

For ease of visualization, we highlight in **bold** the models that do not suffer from collapse. These models exhibit low local entropy and high global entropy, indicating that the MoE does not rely on a single expert nor distribute inputs uniformly across experts. We observe that the interaction between λ_{ls} and λ_{gb} largely determines whether the trained MoE model collapses. These two hyperparame-

ters influence routing patterns in the intended manner: when the model uniformly utilizes all experts (i.e., high local and global entropy), increasing λ_{ls} encourages more confident routing decisions; conversely, when the model collapses to a single expert (i.e., low local and global entropy), increasing λ_{gb} encourages the model to utilize a broader set of experts.

While λ_{ls} and λ_{gb} primarily control router behavior, λ_{div} directly affects model interpretability and expert specialization. Increasing λ_{div} encourages more diverse expert outputs, leading to stronger expert specialization. At the same time, it imposes additional pressure on the router to distribute examples in a more semantically meaningful manner, resulting in higher expert purity.

Overall, although our method introduces three regularization terms and corresponding hyperparameters, the ablation study shows that they serve distinct roles, allowing users to tune them based on the observed model behavior. In practice, we recommend starting with smaller and similar values for λ_{ls} and λ_{gb} to ensure stable training, and selecting λ_{div} according to the desired degree of interpretability.

D Full Results

Category Recovery Results Table 9 and Table 10 present the overall performance of our sparse MoE model on SHP, together with the accuracy and routing weights of each expert. We observe that the global entropy within each category is lower than that on the full dataset, indicating semantically reasonable routing patterns at the category level. Examining the expert distributions further, we find that most categories assign more than 50% of their routing weight to a single expert. Moreover, these dominant experts typically achieve strong performance within their corresponding categories. Together, these results suggest that the sparse MoE reward model learns meaningful routing patterns and behaviorally specialized experts.

We further analyze the performance drop of the sparse MoE reward model compared to the single-head baseline. The largest degradation occurs in the academic and professional categories, where the best-performing experts do not receive the highest routing weights. We therefore attribute the accuracy drop of sparse MoE mainly to a mismatch between routing weights and expert abilities. This mismatch likely arises because these two categories are broad and heterogeneous, making it more difficult for the router to assign examples to the most suitable experts.

Expert Descriptions Table 11 presents the natural language descriptions of all experts in the sparse MoE model trained on 700K examples. Most experts specialize in a concrete type of question or task, exhibiting high description fidelity, while the corresponding experts also specialize behaviorally in the assigned examples. Despite nuanced differences between experts, many focus on recurring categories such as feedback-seeking, safety-related queries, mathematics, code generation, and question answering. These expert patterns partially reflect the underlying structure and distribution of the training data.

We also observe that experts with similar activation patterns, measured by high correlations in their activations across examples, tend to receive similar interpretations. For instance, expert 4 exhibits strong activation correlations with expert 7 (0.95) and expert 16 (0.93), and all three are identified as advice-seeking experts. Similarly, safety-related experts also display highly aligned activation patterns, as illustrated by the 0.96 correlation between experts 5 and 18.

These findings suggest the potential for hierarchical expert interpretation and for reducing expert redundancy through additional regularization on router activation correlations, which we leave for future work.

Personalization Results Test-time adaptation results for all attributes on RPR in Table 12 further demonstrate the strong personalization capability of the sparse MoE reward model, consistent with the findings in Table 5. Our model achieves both the largest improvements over the non-adapted model and the best post-adaptation performance after adapting to individual preferences using only a small set of examples at test time.

E Undesired Flips in Attribute Steering

In addition to measuring successful attribute control (Section 5.2), we also evaluate whether routing weight interventions introduce undesired changes to originally correct predictions. Specifically, we define the undesired flip rate as the proportion of initially correct predictions, under the setting where the target expert is removed, that are flipped after increasing the target expert weight. Lower undesired

λ_{ls}	λ_{gb}	λ_{div}	Accuracy (↑)	Routing Entropy (↓)	Global Entropy (↑)	Expert Correlation (↓)	Expert Purity (↑)	Expert Specialization (↑)
0.00	0.00	0.00	70.67	0.991	0.993	0.556	74.40	-1.00
0.00	0.00	1.00	70.70	0.133	0.139	0.024	80.00	1.12
0.00	0.00	2.00	71.10	0.014	0.015	0.003	70.00	0.47
0.00	0.50	0.00	71.17	1.000	1.000	0.493	43.20	-0.27
0.00	0.50	1.00	71.54	1.000	1.000	0.141	46.40	0.62
0.00	0.50	2.00	71.49	1.000	1.000	0.126	49.20	-0.07
0.00	1.00	0.00	71.24	1.000	1.000	0.520	56.80	-0.20
0.00	1.00	1.00	70.79	1.000	1.000	0.153	47.60	-0.91
0.00	1.00	2.00	71.92	1.000	1.000	0.110	47.20	0.84
0.50	0.00	0.00	70.24	0.000	0.000	0.743	54.80	-0.57
0.50	0.00	1.00	71.44	0.000	0.000	-0.028	46.80	-0.71
0.50	0.00	2.00	71.19	0.000	0.000	-0.027	44.00	-0.93
0.50	**0.50**	**0.00**	**71.39**	**0.087**	**0.984**	**0.782**	**43.20**	**0.06**
0.50	**0.50**	**1.00**	**68.55**	**0.131**	**0.983**	**0.016**	**75.20**	**4.66**
0.50	**0.50**	**2.00**	**69.62**	**0.200**	**0.986**	**0.011**	**85.60**	**5.77**
0.50	**1.00**	**0.00**	**71.40**	**0.084**	**0.989**	**0.746**	**46.80**	**0.34**
0.50	**1.00**	**1.00**	**66.96**	**0.293**	**0.960**	**0.067**	**54.00**	**4.12**
0.50	**1.00**	**2.00**	**69.45**	**0.333**	**0.917**	**0.033**	**67.20**	**3.34**
1.00	0.00	0.00	70.52	0.000	0.000	0.673	46.40	-0.69
1.00	0.00	1.00	70.64	0.000	0.000	-0.026	54.00	-1.23
1.00	0.00	2.00	70.76	0.000	0.000	0.006	34.00	-0.76
1.00	**0.50**	**0.00**	**70.31**	**0.214**	**0.909**	**0.783**	**49.60**	**-0.06**
1.00	0.50	1.00	70.05	0.142	0.148	0.044	80.40	1.10
1.00	0.50	2.00	70.54	0.161	0.166	0.007	92.00	1.35
1.00	**1.00**	**0.00**	**71.17**	**0.083**	**0.983**	**0.746**	**47.20**	**0.38**
1.00	**1.00**	**1.00**	**69.31**	**0.088**	**0.987**	**0.072**	**47.20**	**3.71**
1.00	**1.00**	**2.00**	**66.15**	**0.104**	**0.997**	**0.009**	**54.80**	**3.98**

Table 8: **Ablation study of interpretability regularization terms on SHP.** Models that do not suffer from collapse are marked in **bold**. λ_{ls} and λ_{gb} mainly determine whether the MoE model collapses, while λ_{div} affects model interpretability and expert specialization.

Domain	# Examples	Accuracy (↑)	Routing Entropy (↓)	Global Entropy (↑)	Expert Correlation (↓)
Academic	617	70.50 (-7.06)	0.252	0.883	0.079
Professional	384	70.31 (-4.43)	0.149	0.733	0.038
Cooking	256	66.41 (-0.39)	0.156	0.568	-0.006
Fiction	240	71.25 (+0.84)	0.198	0.704	0.046
Overall	1497	69.62 (-0.90)	0.200	0.986	0.011

Table 9: **Category-level evaluation results on SHP.** Scores in parentheses denote performance comparison against the single-head reward model.

flip rates indicate more robust steering behavior.

Figure 7 reports the undesired flip rates of sparse MoE and the baselines. As individual experts are not perfectly accurate for every attribute, all models exhibit a modest undesired flip rate. Considering the results together with Figure 2, vanilla MoE shows low flip rates in both the desired and undesired directions, suggesting that its experts are highly correlated and have limited influence on model behavior. In contrast, random expert steering in sparse MoE leads to high flip rates in both directions. This indicates that sparse MoE learns more specialized experts, whose weights have a stronger effect on model preferences. Building on this specialization, the improved interpretability of sparse MoE helps identify appropriate target experts for steering. As a result, sparse MoE achieves a high desired flip rate while maintaining a low undesired flip rate. These results suggest that sparse MoE enables more effective post-hoc attribute steering without introducing additional instability.

F Details on Human Study

We recruited five annotators with at least a master's degree in computer science, computational linguistics, or linguistics to evaluate the interpretability of sparse MoE and the baseline models. Annotators are first asked to rate the pattern coherence after seeing the top-activating examples for each expert. They are then presented with the LLM-generated natural language description and asked to rate the description quality.

The ratings show fair to moderate inter-annotator agreement. Sparse MoE achieves statistically significant improvements in pattern coherence over all baselines ($\alpha < 0.05$), while the improvements in description quality are not statistically significant. The statistical significance results are computed

Domain	Expert 0	Expert 1	Expert 2	Expert 3	Expert 4
Academic	65.80 / **35.56**	67.59 / 25.93	62.40 / 10.46	55.92 / 3.76	**69.53** / 24.28
Professional	58.85 / 10.98	52.34 / 6.91	67.45 / **55.40**	52.34 / 2.06	**68.75** / 24.66
Cooking	61.33 / 7.18	61.72 / 21.17	49.61 / 1.19	**68.36** / **68.28**	59.77 / 2.18
Fiction	63.33 / 15.81	56.25 / 4.33	66.25 / 28.57	46.25 / 0.08	**68.33** / **51.21**
Overall	62.86 / 21.23	60.86 / 16.78	62.12 / 23.31	55.58 / 13.77	**67.47** / **24.92**

Table 10: **Expert accuracy and average routing weights on SHP.** Each cell reports expert accuracy / routing weight.

ID	Natural Language Description	Description Fidelity (↑)	Expert Specialization (↑)
0	asks for advice, feedback, or discussion on a personal, professional, or academic challenge	0.43	22.08
1	asks for natural language inference or textual entailment classification	0.26	22.48
2	requests the generation of computer code or technical scripts	0.45	21.53
3	asks for troubleshooting advice regarding a specific problem in cooking, baking, or home maintenance	0.32	15.84
4	asks for personal advice or recommendations	0.50	6.80
5	contains a request for harmful, illegal, or policy-violating content	0.43	7.97
6	asks to solve a procedural task involving numerical or list manipulation	0.51	25.98
7	asks for an opinion, consensus, or general advice on a subjective or non-factual topic	0.41	6.20
8	asks for a summary or extraction of information from a provided text	0.45	19.35
9	asks for general information, advice, or explanation on a broad topic	0.62	8.59
10	asks for a direct answer to a multiple-choice or short-answer question	0.36	20.56
11	asks for technical advice, architectural design, or problem-solving in software engineering and IT	-0.19	19.49
12	requests technical implementation details, system design, or professional strategy plans	0.39	18.69
13	requests assistance with illegal, harmful, or sexually explicit activities	0.62	3.35
14	asks for legal or procedural advice regarding a specific real-world situation	0.33	24.32
15	asks to solve a competitive mathematics problem	0.42	32.00
16	asks for a list of examples, suggestions, or items	0.30	8.60
17	asks a question containing a factual error, category mistake, or logical impossibility	0.30	3.92
18	requests assistance with illegal, harmful, or unethical activities	0.36	6.76
19	asks a question containing a factual error, logical fallacy, or absurd premise	0.35	8.12

Table 11: **Natural language descriptions of all experts in the sparse MoE model.**

from paired t-test.

The annotation instructions and rating scales for pattern coherence and description quality are provided below.

Pattern Coherence Pattern coherence measures whether the input prompts in the same group share a consistent and recognizable pattern. The shared pattern could be a specific task, shared topic, semantic feature, input format, writing style, user intent, domain, or another interpretable property. A high score means that the prompts appear meaningfully related and could plausibly be summarized by a common feature. A low score means that the prompts appear random, unrelated, or too diverse to be summarized by a single coherent pattern.

1. **Not Coherent:** The prompts appear unrelated, random, or inconsistent. No clear shared task, topic, feature, format, style, intent, or domain can be identified.

2. **Weakly Coherent:** A few prompts share some similarities, but the overall group is noisy and difficult to summarize with a single pattern.

3. **Moderately Coherent:** The prompts share a partially recognizable pattern, but there are also several exceptions or mixed themes.

4. **Mostly Coherent:** Most prompts share a clear common task, topic, feature, format, style, intent, domain, or other interpretable property, with only minor exceptions.

5. **Highly Coherent:** The prompts are strongly and consistently related. They clearly reflect a common interpretable concept.

Description Quality Description quality measures how accurately the provided natural language description captures the common pattern shared by the input prompts. This pattern may involve a specific task, shared topic, semantic feature, input format, writing style, user intent, domain, or another interpretable property. A high score means that the description correctly summarizes the main shared pattern among the prompts without adding misleading or unsupported claims. A low score means that the description is inaccurate, too vague, inconsistent with the prompts, or misses the main pattern.

Attribute	Single Reward	HyRe	Vanilla MoE	MiCRo	Sparse MoE
Linguistic Creativity	31.73	37.50 (+6.73)	42.31 (+11.54)	40.39 (+9.62)	**74.04 +(43.27)**
User Friendliness	39.33	46.44 (+9.36)	44.19 (+8.23)	42.32 (+6.36)	**55.06 +(14.61)**
Humor and Entertainment Value	30.16	32.14 (+3.57)	36.51 (+6.75)	40.48 (+10.72)	**80.56 +(50.80)**
Scientific Rigor	76.19	78.17 (+0.79)	81.74 (+3.17)	81.35 (+2.78)	**82.54 +(7.54)**
Narrative and Storytelling Quality	34.17	40.00 (+7.50)	51.67 (+19.17)	51.25 (+18.75)	**68.75 +(28.75)**
Creativity and Originality	18.52	31.02 (+14.35)	28.70 (+10.64)	28.24 (+10.18)	**60.19 +(31.02)**
Factual Accuracy	85.92	81.22 (-0.47)	**89.20 +(7.51)**	88.73 (+7.04)	83.57 (+4.70)
Innovativeness	27.27	38.38 (+9.59)	38.89 (+8.59)	39.39 (+9.09)	**60.10 +(26.77)**
Pedagogical Effectiveness	58.06	58.06 (-1.62)	**60.21** (+0.53)	**60.21** (+0.53)	58.06 **+(4.83)**
Economic Feasibility	45.90	49.18 (-1.64)	50.27 (+1.09)	50.82 (+1.64)	**54.65 +(10.39)**
Technical Complexity	57.38	63.93 (+3.27)	61.75 (+1.09)	61.75 (+1.09)	**68.30 +(10.92)**
Interdisciplinary Approach	50.00	**56.32** (+2.87)	54.02 (+4.02)	54.60 (+4.60)	54.60 **+(11.50)**
Empathy and Emotional Intelligence	54.55	56.36 (0.00)	60.61 (+4.25)	60.00 (+3.64)	**63.64 +(16.37)**
Contextual Relevance	**64.81**	56.79 (-6.17)	62.96 **(-1.85)**	62.96 **(-1.85)**	59.88 (-3.08)
Clarity and Conciseness	75.47	78.62 (+5.04)	77.36 (+3.78)	77.36 (+3.78)	**89.31 +(25.16)**
User Experience	38.46	41.66 (+1.28)	41.67 (+1.29)	41.67 (+1.29)	**43.59 +(7.05)**
Practical Application	44.45	**50.33 +(5.23)**	48.37 (+3.27)	49.02 (+3.92)	45.10 (+1.96)
Overall	48.96	52.71 (+3.51)	54.73 (+5.47)	54.74 (+5.48)	**64.82 +(17.21)**

Table 12: **Personalization results for all attributes on RPR.** Scores in parentheses denote the change after adaptation. Higher scores are better. The sparse MoE model consistently demonstrates stronger personalization performance than baselines.

1. **Poor:** The description does not match the prompts, misses the main pattern, or describes a feature that is mostly absent.

2. **Limited:** The description captures only a small part of the pattern, or is mostly vague, incomplete, or partially incorrect.

3. **Adequate:** The description captures some important shared features, but it is still vague, misses other key aspects or includes noticeable inaccuracies.

4. **Good:** The description accurately captures the main pattern of the prompts, with only minor omissions or minor overgeneralizations.

5. **Excellent:** The description clearly and accurately summarizes the shared features of the prompts, with no meaningful inaccuracies.

Human Study Ethics and Compensation Before participating in the study, annotators were informed about how their data would be used and were told that their annotations would be deleted upon request. The study took approximately 20 minutes on average, and annotators were compensated at an hourly rate comparable to the local average hourly salary for researchers. The human study protocol has been approved by the institutional ethics board.

G Effect of Adaptation Set Size

Figure 8 shows the attribute-level personalization performance of the sparse MoE model under different adaptation set sizes. Results are averaged over 17 attributes and three independently sampled adaptation sets. We observe that, even with only 5 adaptation examples, the model improves accuracy by approximately 13%. As expected, adaptation performance further improves with larger adaptation sets and begins to saturate at around 50 examples.

H User-Level Personalization

We use PersonalLLM for user-level adaptation. Following prior work (Bose et al., 2025; Choi et al., 2025), we model each user as a weight vector over ten reward models. Adaptation and evaluation examples are then constructed by the highest and lowest ranked responses according to user-specific weights. We sample 5 users from a Dirichlet distribution with $\alpha = 0.3$, and use $n = 50$ examples for adaptation. Evaluation is conducted on 200 synthetic examples. Results are averaged over three independently sampled adaptation sets.

Table 13 shows that our sparse MoE reward model also achieves the strongest performance in user-level adaptation. Consistent with the attribute-level adaptation results in Table 5, sparse MoE attains both the best post-adaptation performance and the largest improvements over the non-adapted model. By contrast, the two MoE baselines, vanilla

Method	User1	User2	User3	User4	User5	Overall
Single Reward	93.83	94.83	91.50	94.67	87.17	92.40
HyRe	95.50 (+0.83)	95.17 (-0.33)	**93.00** (-0.17)	95.33 (-0.5)	**89.50 (+1.0)**	93.70 (+0.17)
Vanilla MoE	94.83 (+0.00)	94.67 (-0.83)	90.67 (-2.00)	94.67 (-1.33)	88.83 (+0.50)	92.73 (-0.74)
MiCRo	94.50 (-0.33)	94.67 (-0.83)	90.67 (-2.00)	94.83 (-1.17)	88.17 (-0.16)	92.57 (-0.90)
Sparse MoE	**95.83 (+4.50)**	**96.17 (+3.50)**	92.67 **(+1.34)**	**97.33 (+3.33)**	88.50 (-0.33)	**94.10 (+2.47)**

Table 13: **Personalization results on PersonalLLM.** Scores in parentheses denote the change after adaptation. Higher scores are better. Sparse MoE also demonstrates strong performance in user-level adaptation.

MoE and MiCRo, fail to adapt effectively to new users, with performance degrading after adaptation. The static-weight ensemble model HyRe performs better than the MoE baselines, likely because its simpler adaptation mechanism is more robust in low-data settings.

I Personalized Alignment with Adapted Preferences

To evaluate whether the adapted preferences can support personalized alignment of LLMs, we post-train policies using adapted sparse MoE reward models corresponding to different attributes from RPR. Specifically, we select two attributes, creativity and scientific rigor, which exhibit improved performance after personalization.

For simplicity, we perform GRPO (Shao et al., 2024) post-training on prompts from the RPR training set for 1000 optimization steps. We use an effective batch size of 8 and sample 4 rollouts per prompt. Policies are initialized from Llama3.2-3B-Instruct.

During evaluation, we generate responses for 200 examples from the RPR test set and compare them against a baseline policy trained using the non-adapted reward model. An LLM judge (Qwen3.6-27B) evaluates which response better reflects the target attribute while maintaining higher overall response quality.

Results in Table 14 show that post-training policies with personalized reward models better align the policies with the target attributes compared to using a universal preference model. This demonstrates that the personalization improvements of our sparse MoE reward models also generalize to the post-training phase.

J LLM Usage

We used LLMs for coding assistance, text polishing, and figure generation. All scientific content, experimental decisions, analyses, and conclusions were developed and verified by the authors.

Alignment Type	Creativity		Scientific Rigor	
	Win Rate (Target Attribute, ↑)	Win Rate (Quality, ↑)	Win Rate (Target Attribute, ↑)	Win Rate (Quality, ↑)
Personalized	**53.0**	**52.5**	**52.0**	**52.5**
Universal	45.5	45.5	44.5	45.0

Table 14: **Personalized alignment results compared against alignment using a universal reward model.** After personalization, the sparse MoE reward model can effectively guide policies toward the target attributes without negatively affecting overall generation quality.

Prompt: Describing experts in natural language given top-activating examples

Task Overview
You are a machine learning researcher analyzing a sparse mixture-of-experts reward model.
The model contains a router that assigns each input prompt to one or more experts. You are trying to understand what feature of the input prompt causes a specific expert to receive high routing weight.
Each dataset example consists only of a user prompt. These prompts were selected because this expert was strongly activated for them, meaning the router assigned this expert a high weight.
Your goal is to identify the common feature shared by these prompts, and to describe this feature as a concise, natural language concept.

Instructions
First, inspect each prompt and consider what concrete property it has.

Then, synthesize the single feature that is most consistently shared across the prompts.
The feature should describe the input prompt itself, not the model, the router, or the expert.
For example, the common feature might be:

- "asks for emotionally supportive advice"
- "requests factual explanation of a technical concept"
- "asks for creative writing or brainstorming"
- "contains a safety-sensitive or potentially harmful request"
- "asks for evaluation of competing options"

If the prompts do not share a clear topic, focus on the type of user intent, task format, reasoning style, domain, constraint, or interaction pattern that best explains why the expert fires.

Additional Rules
Rules about the feature you identify:

- The feature should be objective and concrete.
- Err on the side of being concise.
- The feature should be an attribute that could be present in an individual prompt.
- Do not refer to "the expert", "the router", "activation", or "these examples" in the final answer.
- Do not use vague descriptions such as "is helpful", "is complex", or "is high quality".

- Avoid overly broad concepts such as "general question" unless no more specific feature is apparent.
- If the shared feature is a negation, describe it directly, e.g., "does not specify a concrete task or goal".

Prompts
———-
{examples}
———-

Output
Do not output anything besides the feature.
Output exactly one description, starting with "-" and surrounded by quotes.

Your response is:-"

Figure 4: **Prompt used to summarize the common feature of top-activating prompts and describe it in natural language.**

Prompt: Judging whether an example matches expert description

```
You are a binary classifier.

Task:
Determine whether the input text expresses the feature
described below.

Feature description:
{description}

Input text:

{input_text}

Decision rule:
Answer YES if the input text clearly matches or expresses
the feature description.
Answer NO otherwise.

Output format:
Return exactly one token: YES or NO.

Answer:
```

Figure 5: **Prompt used to judge whether an input example matches a given natural language feature description.**

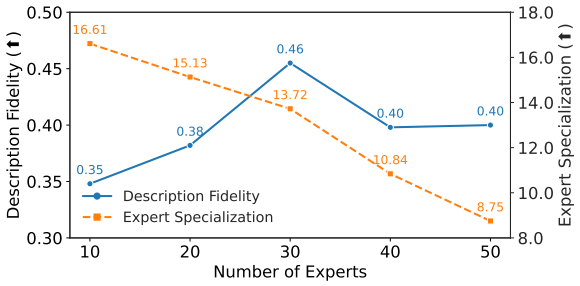

Figure 6: **Effect of the number of experts on interpretability.** Description fidelity peaks at a certain number of experts, while expert specialization consistently decreases as the number of experts increases.

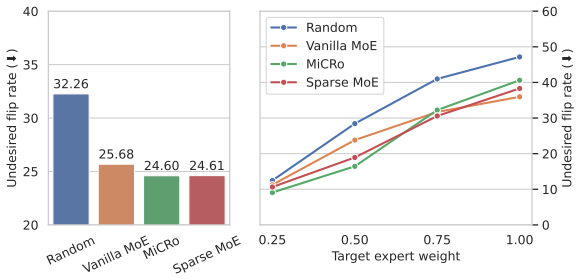

Figure 7: **Undesired flip rate during attribute steering on the RPR dataset.** Lower scores are better. In addition to its strong attribute steering capability, sparse MoE maintains a low undesired flip rate.

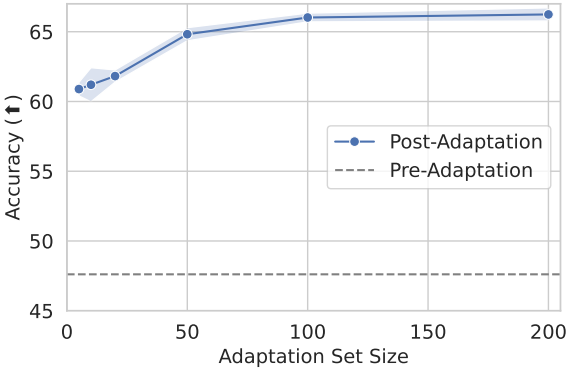

Figure 8: **Personalization results of sparse MoE with different adaptation set sizes on RPR.** Even with only 5 adaptation examples, sparse MoE achieves strong personalization performance. Performance further improves with larger adaptation sets before saturating at around 50 examples.