RLPF: Reinforcement Learning from Prediction Feedback for User Summarization with LLMs

Jiaxing Wu, Lin Ning, Luyang Liu, Harrison Lee, Neo Wu, Chao Wang, Sushant Prakash, Shawn O'Banion, Bradley Green, Jun Xie

Google DeepMind
jxwu@google.com

Abstract

LLM-powered personalization agent systems employ Large Language Models (LLMs) to predict users' behavior from their past activities. However, their effectiveness often hinges on the ability to effectively leverage extensive, long user historical data due to its inherent noise and length of such data. Existing pretrained LLMs may generate summaries that are concise but lack the necessary context for downstream tasks, hindering their utility in personalization systems. To address these challenges, we introduce **R**einforcement **L**earning from **P**rediction **F**eedback (**RLPF**). RLPF fine-tunes LLMs to generate concise, human-readable user summaries that are optimized for downstream task performance. By maximizing the usefulness of the generated summaries, RLPF effectively distills extensive user history data while preserving essential information for downstream tasks. Our empirical evaluation demonstrates significant improvements in both extrinsic downstream task utility and intrinsic summary quality, surpassing baseline methods by up to 22% on downstream task performance and achieving an up to 84.59% win rate on Factuality, Abstractiveness, and Readability. RLPF also achieves a remarkable 74% reduction in context length while improving performance on 16 out of 19 unseen tasks and/or datasets, showcasing its generalizability. This approach offers a promising solution for enhancing LLM personalization by effectively transforming long, noisy user histories into informative and human-readable representations.

1 Introduction

Large Language Models (LLMs) have shown great promise for personalized prediction by leveraging historical activity data (Liu et al. 2023; Lyu et al. 2024; Li et al. 2023). However, the inherent noise and length of user data pose obstacles to their effective utilization in LLM-powered systems.

Natural language user summaries offer several advantages over using raw user activity data. First, they improve inference efficiency over using raw user data due to their compact nature. Second, they offer the potential to improve performance on downstream tasks by distilling user activities and reducing noise. Representing user context through natural language also offers several advantages over embedding-based representations. User representations in the natural language space are reusable across any LLM for downstream tasks without needing to re-train the LLM. In addition, natural language summaries are interpretable and editable, offering users more scrutability and control over their personalized experiences.

Generating user summaries is inherently challenging because user activities lack a ground-truth summary, and their quality is subjective and difficult to define. Existing techniques share a common shortfall: they offer no guarantee that generated summaries will support downstream personalization tasks—a critical function. Each approach also has unique drawbacks. Heuristic methods that extract subsets of activities fail to capture the breadth of user preferences and often produce less readable results. While prompt engineering is popular, pretrained models are not tailored to user data, and crafting effective prompts is both time-consuming and unscalable. Supervised fine-tuning is impractical due to nonexistent training datasets and the privacy concerns associated with collecting such data. Finally, RLHF or RLAIF methods rely on human or AI evaluators, but their judgments remain subjective without standardized criteria.

To overcome the challenges of generating natural language user summaries, we propose **RLPF**: **R**einforcement **L**earning from **P**rediction **F**eedback (illustrated in Figure 1), which includes three components:

- **Summarization Model**: A model is trained to generate succinct user summaries from raw activity data.
- **Prediction-based Reward Model**: To compute a reward, we measure the effectiveness of the generated summaries in downstream prediction tasks.
- **Feedback Loop**: The reward is then used to update the summarization model with RL, with an additional reward to encourage shorter lengths. This feedback loop guides the summarization model to continuously refine its ability to produce summaries that are not only concise but also highly effective for their intended applications.

RLPF offers a win-win solution: it enables the creation of high-quality user summaries without the need for resource-intensive and potentially privacy-compromising human intervention. By directly optimizing the summarization process for downstream prediction performance, we ensure that the generated summaries are both compact and directly relevant to the tasks they are meant to support. Furthermore,

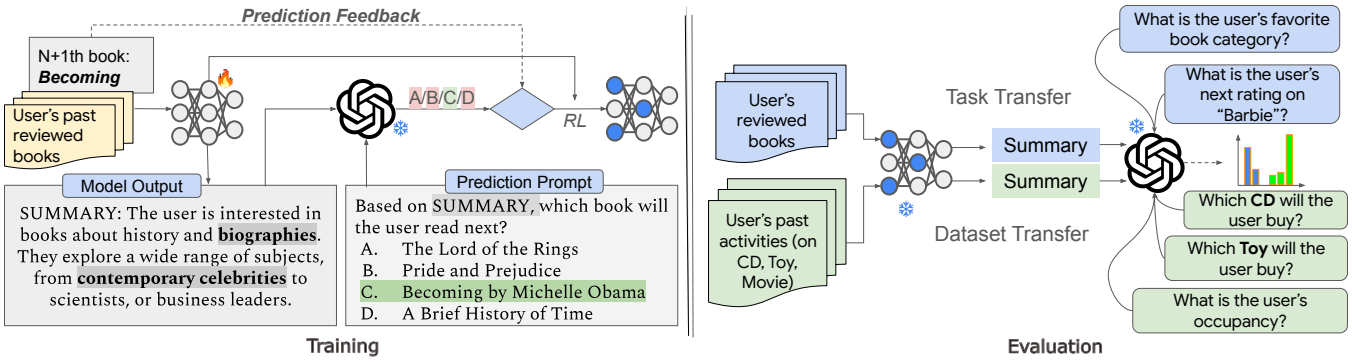

Figure 1: **Overview of RLPF**. Left: Training process of RLPF, in which future activity will be used towards reward computation. Right: We assess RLPF on unseen downstream prediction tasks to demonstrate its generalizability and adaptability.

compared to prevailing Reinforcement Learning (RL) approaches relying on feedback from a dedicated trained reward LLM (Ouyang et al. 2022; Bai et al. 2022; Lee et al. 2024; Yang et al. 2023), RLPF eliminates the overhead of training a separate reward model.

Through extensive experiments on four public datasets grounded in real-world user interactions - MovieLens 2015 and 2003 (Harper and Konstan 2015), Amazon Review (He and McAuley 2016), and Google Local Review (Yan et al. 2022), we demonstrate that RLPF summaries outperform baselines in terms of predictive power on both seen and unseen tasks, as well as on intrinsic quality evaluations.

Our contributions are four-fold:

- We introduce the novel task of generating natural language user summaries for user modeling and personalization systems. This offers an interpretable alternative to traditional embedding-based representations and allows utilization by arbitrary LLMs without further training.
- We introduce RLPF, a novel and easy-to-implement method for training user summarizers. RLPF eliminates the need for reference summaries or hand-crafted prompts, while safeguarding user privacy.
- We demonstrate that RLPF summaries outperform baselines on both the training task and unseen tasks across four datasets and domains.
- We evaluate RLPF summaries intrinsically and find significant improvements in factuality, abstractiveness, and readability.

2 Methodology

Problem Statement

Consider a set of users $\mathcal{U} = \{u_i\}_{i=1}^{\mathcal{M}}$, where each user i has an associated chronologically ordered sequence of interactions, denoted as $\{v_i^1, v_i^2, ..., v_i^N\}$. Each v_i^j within this sequence (where $1 \leq j \leq N$) comprises one or more textual features that describe a specific item, such as the titles or ratings of movies watched by the user. For each user i, we concatenate all of their interactions $\{v_i^j\}_{j=1}^{N}$ into a single string to form the user context u_i.

A summarizer model π_θ takes as input the user context and generates a summary $s_i = \pi_\theta(u_i)$. The summary is then provided to off-the-shelf LLM to produce a prediction $\hat{y}_i = \mathcal{P}(s_i)$ for a specific downstream task. We optimize π_θ to generate summaries $\{s_i\}_{i=1}^{\mathcal{M}}$ that minimize the expected error between the predictions $\{\hat{y}_i\}_{i=1}^{\mathcal{M}}$ and the ground truth task labels $\{y_i\}_{i=1}^{\mathcal{M}}$.

Reinforcement Learning from Prediction Feedback

In the context of RL, we formulate summary generation as a Contextual Markov Decision Process (CMDP). In this framework, the state encompasses both the input text and the partially generated summary, while actions correspond to the selection of tokens from the entire vocabulary. At each step, the policy model maps these states to probability distributions over the vocabulary, facilitating autoregressive token selection. This selection process is guided by the current context and the overarching objective of maximizing cumulative rewards.

Within this RL framework, we formalize RLPF in the context of user summarization as follows:

- **State:** The set of user contexts $\mathcal{U} = \{u_i\}_{i=1}^{\mathcal{M}}$, where each u_i is a single string representing the textual features of a user's N past activities.
- **Action:** The set of user summaries $S = \{s_i\}_{i=1}^{\mathcal{M}}$ generated based on the corresponding user contexts.
- **Policy Model:** The summarizer model, denoted by π_θ, which maps user contexts (states) to user summaries (actions): $\pi(u_i; \theta) \rightarrow s_i$.
- **Reward:** We leverage a frozen, pre-trained LLM to generate predictions $\mathcal{P}(s_i)$ for one or more specified tasks based on user summaries s_i. Then a scalar reward value is computed by comparing the prediction $\mathcal{P}(s_i)$ with its corresponding ground truth label y_i of the specific task.

The objective of RLPF is to learn a policy π^* that maximizes the expected cumulative reward:

$$\pi^* = \arg\max_{\pi} \mathbb{E}_{u_i \sim \mathcal{U}}[r(\pi(u_i; \theta))]$$

Reward Computation RLPF provides the flexibility to leverage any task for reward derivation, tailored to specific

downstream application requirements. Moreover, it seamlessly accommodates the combination of rewards from multiple tasks if needed. Our implementation leveraged future activity prediction as the sole task for generating reward signals. This approach demonstrated strong generalization and transferability to unseen tasks, as detailed in the Results section. This underscores the convenience and efficiency of RLPF by eliminating the need for extensive, complex model training and overhead. Further results using alternative reward tasks, along with guidelines for task selection, are provided in the Appendix F.

For each user i, summary reward $r(s_i)$ is as follows:

$$r(s_i) = r^{pred}(s_i,\ y_i) + w \cdot r^{len}(s_i)$$

where $r^{pred}(.)$ is the prediction feedback reward, $r^{len}(.)$ is the length reward, and w is a weight that controls the balance between the two terms.

Prediction Feedback Reward: Recall that each user context u_i consists of the textual features of N past user activities. We employ the subsequent $(N+1)$-th activity (e.g., watched movie title etc.) as the ground truth label y_i for predicting the future activity. Given the user summary s_i, we calculate a binary reward by comparing the LLM's prediction based on s_i to the actual future activity v_i^{N+1}:

$$r^{pred}(s_i,\ y_i) = \mathbb{1}(\mathcal{P}(s_i) = y_i),\ \textit{where}\ y_i = v_i^{N+1}$$

However, since the reward model operates in a zero-shot setting, predicting item names with exact matches without any additional context is challenging due to the vast number of possibilities. This hinders the policy model's ability to receive positive feedback and learn effectively. To tackle this issue, we adopt a multiple-choice approach, providing four answer choices for each summary based prediction, including the ground truth. The reward model is then prompted to select the correct option from the given choices. Notably, our method is adaptable to any closed-ended question formats. See Appendix K for full prompts.

Length Reward: Furthermore, to promote concise summary generation, we incorporate a length reward:

$$r^{len}(s_i) = \min[\mathcal{C},\ \beta * (\mathcal{L} - l_i)]$$

where l_i represents the token length of summary s_i, and the hyperparameters $\mathcal{M}$, β, and $\mathcal{L}$ denote the upper bound, magnitude, and target length of the summary, respectively. We set the target length to the average length of Zero Shot summaries in our experiments. See variable values in Appendix D.

Training Process The absence of reference summaries prevents the application of supervised fine-tuning to either the policy or reward model. Unlike the standard RLHF pipeline, which sequentially involves supervised fine-tuning, reward modeling, and policy optimization, RLPF directly optimizes the policy in a single RL training step. By leveraging LLMs' inherent zero-shot summarization and prediction capabilities, RLPF eliminates the need for intricate prompt engineering, generating feedback for the RL process based on predicted future activities. While RLPF is not tied to any specific RL algorithm, we utilize REINFORCE (Williams 1992) with a baseline to update the policy model given that it is simpler yet still effective for our tasks. Both policy and value models are initialized from a frozen model.

To preserve the LLM's original summarization capability and mitigate reward hacking, we introduce a KL divergence term between the current policy π_θ and the initial policy π_{init}. Consequently, the policy parameters are updated according to the following rule:

$$\theta \leftarrow \theta + [(1-\alpha)\nabla_\theta \mathbb{E}[r_i] - \alpha \mathbb{E}[\nabla_\theta KL(\pi_\theta || \pi_{init})]]$$

where α is a hyperparameter controlling the balance between the reward maximization and policy regularization.

3 Experimental Details

Dataset

We conduct experiments on four public datasets grounded in real-world user interactions, encompassing product reviews, movie watching behavior, and location data. We perform training on *Amazon Books* (He and McAuley 2016), *Google Local Review* (Yan et al. 2022), *MovieLens 2015*(Harper and Konstan 2015). Additionally, we utilized another four *Amazon Review* datasets with different product categories, as well as *MovieLens 2003*, which features distinct users and movie catalogs compared to *MovieLens 2015*). See Appendix C for dataset details.

Data Generation For each user's interaction data, presented as a chronologically ordered list of activities $u_i \in \mathcal{U}$, we randomly select one item as the target for future activity prediction, denoted as y_i. We utilize the N activities preceding this target as the past activities $\{v_i^j\}_{j=1}^N$. v_i^j represents an item name and rating pair, where item name correspond to movie title for MovieLens, product name for Amazon Review, and place name + city name for Google Local Review, respectively. As previously mentioned, we concatenate $\{v_i^j\}_{j=1}^N$ to construct the user context u_i. To prevent label leakage, the last item in each user's data is reserved as the target item in the test set. Unless otherwise specified, we set $N = 50$ in our experiments.

Evaluation Metrics

Extrinsic Utility We gauge the predictiveness of the summaries based on their prediction performance in various downstream tasks. Extending beyond *Future Activity Prediction* which is used as feedback during training, we incorporated additional tasks of various types to gauge the transferability and generalization capabilities of the generated summaries. These included **19** tasks include user interest reasoning, history activity retrieval, rating prediction, user demographic prediction and open text review generation. Please refer to Appendix I for detailed task definitions as well as their abbreviation used in the paper.

A frozen instruction tuned Gemini 1.0 Pro model was employed to generate predictions for all downstream tasks. Each summary s_i was fed into the model, and the resulting predictions were evaluated against ground truth labels.

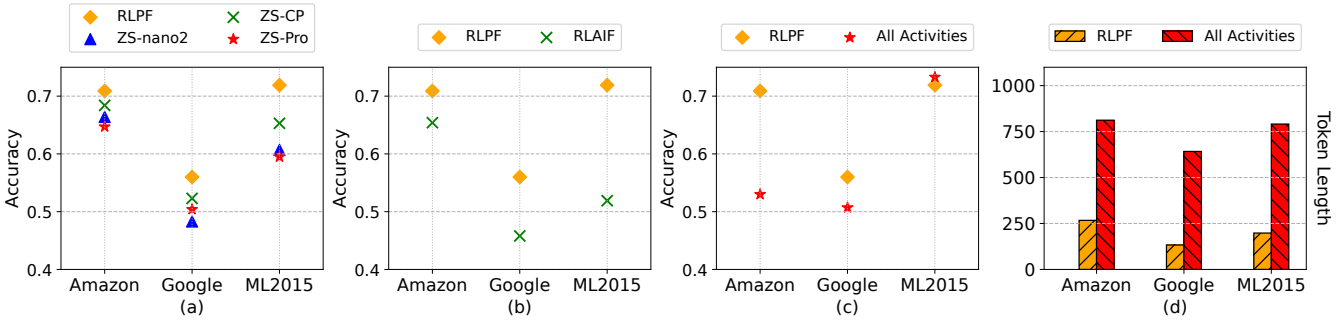

Figure 2: RLPF summaries consistently demonstrate superior performance in Future Activity Prediction, surpassing both other summarization techniques and the full user context ("All Activities"), while significantly reducing the required context length. ZS-nano2: Gemini Nano-2 Zero-Shot; ZS-CP: Gemini Nano-2 with Crafted Prompts; ZS-Pro: Gemini Pro Zero-Shot.

Intrinsic Quality To further assess the intrinsic quality of the generated summaries, we utilize automated evaluation to compare summaries before and after training. This assessment focuses on aspects not explicitly covered by downstream task performance, including *Factuality*, *Abstractiveness*, *Readability* and *Overall* quality. For each criterion and overall quality, the Auto Rater compares a pair of summaries, with their relative positions randomly assigned to eliminate potential bias in the evaluation. We harnessed the most powerful model in the Gemini family, Gemini 1.5 Pro (GeminiTeam et al. 2024), as the Auto Rater. See Appendix A for the full prompt.

In addition to using Auto Rater, the Appendix G provides further results and discussions on employing grounded evaluation metrics to assess factuality and readability.

Training Details

The summarizer model, or policy model π_θ, is initialized from Gemini 1.0 Nano-2(instruction tuned) and fine-tuned using RLPF. During training, reward computation ($\mathcal{P}(s_i)$) is performed by a frozen, instruction-tuned Gemini 1.0 Pro model, which predicts future activity based on the generated summary s_i. Gemini 1.0 Pro was selected for its optimal balance of performance and inference efficiency. In additiona, we also employed PaLM-2 XS (Anil et al. 2023) to showcase RLPF's applicability across diverse policy models.

For each of the three training datasets (Amazon Books, Google Local Review, and MovieLens 2015), we trained the policy model with a batch size of 64 for 15,000 steps, and evaluation was performed on the final checkpoint. More hyper-parameter values are listed in Appendix D.

Baselines

We compare the performance of user summary generated by RLPF against two categories of baselines: summary-based and activity-based. As the evaluator model makes zero-shot predictions for all inputs, any performance differences are attributed to the informativeness of the input, assuming consistent prediction capability.

- **Summary-Based Baselines**: We employ frozen instruction tuned or fine-tuned models to generate summaries and assess their downstream performance.
  - **Gemini 1.0 Nano-2 Zero-Shot:** Uses summaries generated by Gemini 1.0 Nano-2 in a zero-shot manner. This represents the anchor model before training.
  - **Gemini 1.0 Pro Zero-Shot:** Uses summaries generated by Gemini 1.0 Pro in a zero-shot manner, a larger and more powerful model than the anchor model.
  - **Gemini 1.0 Nano-2 Few-Shot:** Uses summaries generated by Gemini 1.0 Nano-2 in a few-shot manner. We provided two examplars in context, where the example summaries are generated by Gemini 1.5 Pro. See full prompts in Appendix K.
  - **Gemini 1.0 Nano-2 with Crafted Prompt:** Uses summaries from Gemini 1.0 Nano-2, but with custom-designed prompts optimized for downstream tasks. We show the prompt in Appendix K.
  - **RLAIF:** User summaries trained with Direct RLAIF (Lee et al. 2024), using Gemini 1.0 Nano-2 as the policy model. The reward score is provided by an LLM (Gemini 1.0 Pro). Further details on the prompting technique are available in the Appendix K.
- **Activity-Based Baselines**: The user context $u_i$is directly fed as input to a frozen instruction tuned model (Gemini 1.0 Pro) to generate predictions:
  - **First X Activities:** Uses only the earliest X activities ($X < N$) for downstream task predictions, ensuring comparable token length to RLPF summaries.
  - **Random X Activities:** Similar to the above, but selects X activities randomly.
  - **Last X Activities:** Uses the most recent X activities.
  - **All Activities:** Uses the full user context N activities.

4 Results

Target Task Performance

Figure 2 compares RLPF performance on the Future Activity Prediction task. Across all three datasets, RLPF demonstrates superior or comparable performance to various summarizers, including crafted prompting, a larger summarizer model, and RLAIF. Overall, RLPF outperforms Nano-2

	Training Dataset	Evaluation Dataset	Evaluation Task	0-Shot	RLAIF	RLPF	vs 0-Shot	vs RLAIF
Task Transfer	MovieLens 2015	MovieLens 2015	Fav Genre	0.774	0.776	**0.818**	5.68%	5.48%
	MovieLens 2015	MovieLens 2015	Rating	0.225	0.229	**0.232**	3.11%	1.31%
	Amazon Books	Amazon Books	Fav Category	0.594	**0.613**	0.605	1.85%	-1.27%
	Amazon Books	Amazon Books	Rating	0.244	0.147	**0.255**	4.51%	73.47%
	Amazon Books	Amazon Books	Review Gen	13.52	**13.68**	13.46	-0.41%	-1.58%
	Google Local	Google Local	Fav Category	0.487	0.513	**0.559**	14.78%	8.90%
	Google Local	Google Local	Rating	**0.118**	**0.118**	0.111	-5.93%	-5.93%
	Google Local	Google Local	Common City	0.765	0.791	**0.901**	17.73%	13.93%
Dataset Transfer	MovieLens 2015	MovieLens 2003	Future Act	0.468	0.447	**0.509**	8.82%	13.93%
	MovieLens 2015	Amazon Movies	Future Act	0.572	0.579	**0.606**	5.94%	4.66%
	Amazon Books	Amazon Movies	Future Act	0.645	0.573	**0.663**	2.73%	15.68%
	Amazon Books	Amazon CDs	Future Act	0.397	0.447	**0.573**	44.33%	28.22%
	Amazon Books	Amazon Toys	Future Act	0.620	0.585	**0.644**	3.94%	10.14%
	Amazon Books	Amazon Games	Future Act	0.688	0.631	**0.713**	3.60%	12.90%
Task & Dataset Transfer	MovieLens 2015	MovieLens 2003	Fav Genre	0.808	0.801	**0.843**	4.35%	5.26%
	MovieLens 2015	MovieLens 2003	User Age	0.274	**0.341**	0.246	-10.22%	-27.86%
	MovieLens 2015	MovieLens 2003	User Gender	0.723	**0.738**	0.729	0.90%	-1.15%
	MovieLens 2015	MovieLens 2003	User Occupancy	0.146	0.130	**0.162**	11.20%	24.89%
	MovieLens 2015	MovieLens 2003	Rating	0.228	0.224	**0.245**	7.50%	9.38%

Table 1: RLPF, trained exclusively on future activity prediction, exhibits remarkable transferability and generalization across diverse unseen tasks and datasets. Evaluation metrics: recall@3 for Favorite Genre/Category, Common City, and User Occupancy; ROUGE-Lsum for Review Gen; and accuracy for the remaining tasks.

zero-shot summaries by +13.4% improvement, and outperforms RLAIF by +22% on average. Compared to utilizing the full user context (all activities), RLPF achieves an average context length compression of -73.8% while still exhibiting a +12.4% performance gain. Further comparisons with other baselines are provided in the Appendix H, underscoring exceptional capability of RLPF summaries to capture both short-term and long-term user context information.

For comparison, we conducted supervised fine-tuning of a Gemini 1.0 Pro model on the same task, reaching 94% accuracy. However, this fine-tuned model exhibited zero performance on other tasks, highlighting its overfitting to the specific training task. Conversely, RLPF showcased remarkable transferability and generalization capabilities, as demonstrated in the subsequent section.

Transferability and Generalization

To evaluate the generalizability and adaptability of RLPF for various personalization agent systems, we conducted a comprehensive transferability assessment across a diverse set of unseen tasks and datasets. As shown in Table 1, RLPF summaries consistently exhibited superior transferability compared to zero-shot and RLAIF baselines, demonstrating improvements in 16 and 14 out of 19 total evaluation cases, respectively. These results highlight RLPF's exceptional transferability and its potential to be effectively applied to a wide range of personalization scenarios, particularly when training data is scarce.

Task Transfer RLPF summaries demonstrated a slight improvement on an unseen retrieval task, common city retrieval on Google Local Review, and performed on par with zero-shot summary on an unseen personalized text generation task, review generation on Amazon Books.

Dataset and Domain Transfer We also evaluated whether an RLPF trained model can generalize to an unseen dataset, either in same domain or a different domain. We used the policy model trained with MovieLens 2015 to generate summaries on MovieLens 2003 and Amazon Movies&TVs dataset and evaluated future movie prediction with the generated summaries. From the results, RLPF model trained on MovieLens 2015, showed improvements on both unseen datasets. Furthermore, the model trained on Amazon Books achieved significant performance gains on Amazon CDs&Vinyl data, highlighting its strong domain adaptation abilities.

Task and Dataset Transfer Furthermore, we evaluated RLPF model performance on unseen tasks from unseen datasets. RLPF model trained with MovieLens 2015 with future activity prediction showed improvement on MovieLens 2003 dataset in favorite genre prediction and user demographic reasoning.

Intrinsic Evaluation

Table 2 demonstrates that RLPF summaries consistently outperform zero-shot summaries on all three datasets, as evaluated by the automated rater across all criteria: *Factuality*, *Abstractiveness*, *Readability*, and *Overall* evaluation.

This finding is noteworthy given that RLPF was trained solely on reward signals from future activity prediction. Despite this focused training, RLPF summaries not only avoid degradation or overfitting to a single goal but also exhibit

Dataset	Criteria	RLPF Win Rate vs Zero-Shot	RLPF Win Rate vs RLAIF
MovieLens 2015	Factuality	61.32%	62.53%
	Abstractiveness	62.54%	56.09%
	Readability	62.42%	56.36%
	Overall	62.47%	56.10%
Amazon Books	Factuality	72.93%	40.09%
	Abstractiveness	70.14%	39.20%
	Readability	71.28%	35.47%
	Overall	70.08%	39.17%
Google Local Review	Factuality	77.58%	49.97%
	Abstractiveness	84.59%	54.56%
	Readability	83.73%	46.02%
	Overall	84.46%	54.22%

Table 2: Intrinsic Evaluation with Auto Rater.

significant improvements in other crucial aspects. This suggests that when employing RLPF for user summarization, designing explicit reward signals for each criterion, which can be challenging to obtain, may not be necessary. Instead, future activity prediction performance appears to provide correlated and implicit signals for these criteria. Intuitively, to make accurate future activity predictions, a summary needs to be factually consistent and distill key user information. While readability might not be a strict prerequisite for future activity prediction, it's noteworthy that this criterion also correlates with this downstream task.

Interestingly, RLPF's performance on par with RLAIF in this evaluation, even though RLAIF was specifically trained with reward signals more aligned with the intrinsic evaluation criteria, highlights the effectiveness of RLPF.

Analysis

Alternative Policy Model Additionally, we applied RLPF to a policy model initialized from the PaLM-2 XS model, with results presented in Table 3. Mirroring the observations with Gemini 1.0 Nano-2, RLPF summaries based on PaLM-2 XS also exhibited improvements in both the training task (future activity prediction) and the unseen task (favorite genre/category prediction) across all three datasets. A slight drop in performance was noted for favorite genre prediction on the MovieLens 2015 dataset.

Robustness to Model that Uses Summaries To further ensure that RLPF summaries are not overly tailored to the specific reward model used during training, we employed an additional evaluator model PaLM-2 S to assess their performance. As in previous experiments, RLPF summaries were trained using reward signals derived from Gemini 1.0 Pro. Table 4 demonstrates that the improvements achieved with RLPF summaries transfer effectively to these different evaluator models, highlighting the generalizability of RLPF summaries across various LLM-powered systems.

Impact of Summary Length Figure 3 illustrates our experiments on MovieLens 2015, where we varied the target length($\mathcal{L}$) in the length reward term. Generally, longer summaries led to improved task performance but decreased

Dataset	Task	PaLM-2 XS zero-shot	PaLM-2 XS RLPF
MovieLens 2015	Future Act	0.638	**0.741**
	Fav Category	0.860	**0.849**
Amazon Books	Future Act	0.626	**0.675**
	Fav Category	0.557	**0.565**
Google Local Review	Future Act	0.502	**0.532**
	Fav Category	0.454	**0.477**

Table 3: RLPF with PaLM-2 XS as the policy model.

Dataset	Task	zero-shot	RLPF
MovieLens 2015	Future Act	0.578	**0.674**
	Fav Category	0.822	**0.840**
Amazon Books	Future Act	0.689	**0.734**
	Fav Category	0.543	**0.567**

Table 4: Evaluated using PaLM-2 S, with reward signals derived from Gemini 1.0 Pro during training.

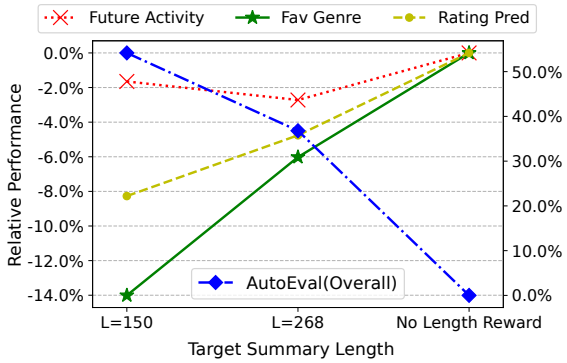

Figure 3: Impact of Different Target Lengths on MovieLens 2015. Percentage changes are calculated relative to "No Length Reward" condition (no maximum length constraint). Data on the right axis pertains to AutoEval, while the left axis corresponds to the remaining tasks.

scores in automated evaluation metrics, suggesting a trade-off between extrinsic utility and intrinsic qualities.

Robustness to Prompts We investigated the impact of varying prompts for summary generation and prediction during reward computation. As illustrated in Figure 4, task returns converge to a similar level despite initial differences in zero-shot performance, demonstrating the robustness of RLPF to diverse prompts. See full prompts in Appendix K.

Qualitative Observation In general, zero-shot summaries tend to mimic the structure of the input, which may either be directly copied from the input activities or represent hallucinations (e.g., mentioning popular movies like "The Godfather" despite their absence in the user history). After RLPF training, summaries become more coherent and

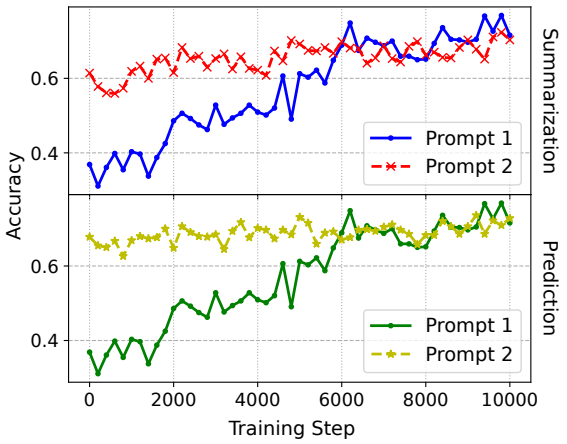

Figure 4: **RLPF is robust with various prompts.** Top: Evaluation metric with different prompts for *Summarization*, Bottom: Evaluation metric with different prompts for *Prediction* during reward computation. Prediction task: Future activity prediction on MovieLens 2015.

distill user information effectively, though some repetition or hallucination may still occur occasionally. This also explains the better Factuality and Abstractiveness scores from the Automated Evaluation. Although, we noticed both RLPF and RLAIF summaries sometimes exhibit repetitive patterns (e.g., "I am always up for a good recommendation"), while the core content remains user-specific. See the Appendix J for example summaries.

5 Discussion

Responsible Deployment While RLPF shows promise for enhancing personalization, its use of user data raises privacy and data leakage concerns. Offline training of user summaries and employing a frozen LLM for online serving can mitigate some risks. However, a thorough analysis of potential vulnerabilities is crucial before real-world deployment.

6 Related Work

Text Summarization

Leveraging language models for summarizing long documents has gained prominence. Unlike text summarization, which condenses texts while retaining key information, our approach focuses on distilling implicit user insights and preferences beyond merely extracting user history.

User summarization poses distinct challenges in model training and evaluation due to the absence of ground truth user summaries. In text summarization, widely-used datasets with reference summaries (Hermann et al. 2015; Narayan, Cohen, and Lapata 2018; Napoles, Gormley, and Van Durme 2012) enable supervised fine-tuning (Cohan et al. 2018; He et al. 2022, 2023; Kryściński et al. 2021; Roit et al. 2023) or RL with reward signals comparing generated and reference summaries (Gunasekara et al. 2021), as well as evaluation metrics with lexical matching (Lin 2004) or embedding similarity (Zhang et al. 2020). These methods and metrics are inapplicable to user summarization due to the lack of datasets. Human evaluation has been used in text summarization (Goyal, Li, and Durrett 2023), but privacy concerns make it impractical for user summarization.

Previous summarization work without reference summaries aligns more with ours. These methods often leverage question-answering (QA) (Durmus, He, and Diab 2020; Fabbri et al. 2022; Deutsch, Bedrax-Weiss, and Roth 2021; Fabbri et al. 2021) or pre-trained models (Kryscinski et al. 2020; Goyal and Durrett 2020), relying on the capabilities of QA generation or entailment models. However, no datasets exist for training these models on user activity data. Our work also employs QA and pre-trained LLMs for reward computation, but takes a practical approach by grounding reward signals in real-world personalization questions with answers derived directly from user data, avoiding the need to train additional QA models.

User Modeling

User modeling has benefited significantly from LLM advancements. While existing methods often represent user activity with embeddings (Ning et al. 2024; Doddapaneni et al. 2024), our work generates natural language-based user summaries, a more human-readable and reusable alternative.

Previous work on natural language-based user modeling has primarily relied on prompting or fine-tuning for specific downstream tasks (Bao et al. 2023; Wu et al. 2024b; Liu et al. 2023; Lyu et al. 2024; Li et al. 2023; Salemi et al. 2024; Wang et al. 2024) or pre-defined user attributes (Rao, Leung, and Miao 2023; Ji et al. 2023; Wu et al. 2024a). In contrast, our approach introduces a novel end-to-end training framework for generating user summaries. This method focuses on comprehensive user profiling to support a wide range of downstream tasks, rather than focusing on a single user attribute or characteristic.

Reinforcement Learning from AI Feedback

RL from Human Feedback (RLHF) (Ouyang et al. 2022) aligns language models with human values but relies heavily on high-quality human labels. To mitigate this dependency, RL from AI Feedback (RLAIF)(Bai et al. 2022; Yang et al. 2023) utilizes off-the-shelf LLMs to replace human annotations, achieving superior performance on tasks like summarization (Lee et al. 2024). RLAIF scores summaries directly using an LLM, which introduces subjectivity due to the lack of standardized criteria. In contrast, our approach RLPF uses downstream task performance as the reward signal, enabling direct optimization for improved personalization.

7 Conclusions

We introduced RLPF, a novel method to generate human-readable user summaries from raw activity data. RLPF leverages readily available LLMs and downstream task performance as reward signals, overcoming challenges in traditional summarization approaches. Our experiments demonstrate superior performance, context compression, and generalizability across unseen tasks. Future work will extend RLPF to more complex scenarios, additional feedback mechanisms, and broader applications.

References

Anil, R.; Dai, A. M.; Firat, O.; Johnson, M.; Lepikhin, D.; Passos, A.; Shakeri, S.; Taropa, E.; Bailey, P.; Chen, Z.; et al. 2023. PaLM 2 Technical Report. *arXiv preprint arXiv:2305.10403*.

Bai, Y.; Kadavath, S.; Kundu, S.; Askell, A.; Kernion, J.; Jones, A.; Chen, A.; Goldie, A.; Mirhoseini, A.; McKinnon, C.; et al. 2022. Constitutional AI: Harmlessness from AI Feedback. *arXiv preprint arXiv:2212.08073*.

Bao, K.; Zhang, J.; Zhang, Y.; Wang, W.; Feng, F.; and He, X. 2023. TALLRec: An Effective and Efficient Tuning Framework to Align Large Language Model with Recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems*. ACM.

Cohan, A.; Dernoncourt, F.; Kim, D. S.; Bui, T.; Kim, S.; Chang, W.; and Goharian, N. 2018. A Discourse-Aware Attention Model for Abstractive Summarization of Long Documents. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, 615–621.

Deutsch, D.; Bedrax-Weiss, T.; and Roth, D. 2021. Towards Question-Answering as an Automatic Metric for Evaluating the Content Quality of a Summary. *Transactions of the Association for Computational Linguistics*.

Doddapaneni, S.; Sayana, K.; Jash, A.; Sodhi, S.; and Kuzmin, D. 2024. User Embedding Model for Personalized Language Prompting. In *Proceedings of the 1st Workshop on Personalization of Generative AI Systems (PERSONALIZE 2024)*.

Durmus, E.; He, H.; and Diab, M. 2020. FEQA: A Question Answering Evaluation Framework for Faithfulness Assessment in Abstractive Summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.

Fabbri, A.; Wu, C.-S.; Liu, W.; and Xiong, C. 2022. QAFactEval: Improved QA-Based Factual Consistency Evaluation for Summarization. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Fabbri, A. R.; Kryściński, W.; McCann, B.; Xiong, C.; Socher, R.; and Radev, D. 2021. SummEval: Re-evaluating Summarization Evaluation. *arXiv preprint arXiv:2007.12626*.

GeminiTeam; et al. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*.

Goyal, T.; and Durrett, G. 2020. Evaluating Factuality in Generation with Dependency-level Entailment. In *Findings of the Association for Computational Linguistics: EMNLP*.

Goyal, T.; Li, J. J.; and Durrett, G. 2023. News Summarization and Evaluation in the Era of GPT-3. *arXiv preprint arXiv:2209.12356*.

Gunasekara, C.; Feigenblat, G.; Sznajder, B.; Aharonov, R.; and Joshi, S. 2021. Using question answering rewards to improve abstractive summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, 518–526.

Harper, F. M.; and Konstan, J. A. 2015. The MovieLens Datasets: History and Context. *ACM Trans. Interact. Intell. Syst.*, 5(4).

He, J.; Kryscinski, W.; McCann, B.; Rajani, N.; and Xiong, C. 2022. CTRLsum: Towards Generic Controllable Text Summarization. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 5879–5915.

He, P.; Peng, B.; Wang, S.; Liu, Y.; Xu, R.; Hassan, H.; Shi, Y.; Zhu, C.; Xiong, W.; Zeng, M.; Gao, J.; and Huang, X. 2023. Z-Code++: A Pre-trained Language Model Optimized for Abstractive Summarization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

He, R.; and McAuley, J. 2016. Ups and Downs: Modeling the Visual Evolution of Fashion Trends with One-Class Collaborative Filtering. In *Proceedings of the 25th International Conference on World Wide Web*.

Hermann, K. M.; Kočiskỳ, T.; Grefenstette, E.; Espeholt, L.; Kay, W.; Suleyman, M.; and Blunsom, P. 2015. Teaching Machines to Read and Comprehend. *Advances in neural information processing systems*, 28.

Ji, Y.; Wu, W.; Zheng, H.; Hu, Y.; Chen, X.; and He, L. 2023. Is chatgpt a good personality recognizer? a preliminary study. *arXiv preprint arXiv:2307.03952*.

Kryscinski, W.; McCann, B.; Xiong, C.; and Socher, R. 2020. Evaluating the Factual Consistency of Abstractive Text Summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Kryściński, W.; Rajani, N.; Agarwal, D.; Xiong, C.; and Radev, D. 2021. BookSum: A Collection of Datasets for Long-form Narrative Summarization. *arXiv preprint arXiv:2105.08209*.

Lee, H.; Phatale, S.; Mansoor, H.; Mesnard, T.; Ferret, J.; Lu, K.; Bishop, C.; Hall, E.; Carbune, V.; Rastogi, A.; and Prakash, S. 2024. RLAIF vs. RLHF: Scaling Reinforcement Learning from Human Feedback with AI Feedback. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*.

Li, R.; Deng, W.; Cheng, Y.; Yuan, Z.; Zhang, J.; and Yuan, F. 2023. Exploring the Upper Limits of Text-Based Collaborative Filtering Using Large Language Models: Discoveries and Insights. *arXiv preprint arXiv:2305.11700*.

Lin, C.-Y. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out*, 74–81. Barcelona, Spain: Association for Computational Linguistics.

Liu, J.; Liu, C.; Zhou, P.; Lv, R.; Zhou, K.; and Zhang, Y. 2023. Is ChatGPT a Good Recommender? A Preliminary Study. *arXiv preprint arXiv:2304.10149*.

Lyu, H.; Jiang, S.; Zeng, H.; Xia, Y.; Wang, Q.; Zhang, S.; Chen, R.; Leung, C.; Tang, J.; and Luo, J. 2024. LLM-Rec: Personalized Recommendation via Prompting Large Language Models. *arXiv preprint arXiv:2307.15780*.

Napoles, C.; Gormley, M.; and Van Durme, B. 2012. Annotated Gigaword. *arXiv preprint arXiv:1204.4134*.

Narayan, S.; Cohen, S. B.; and Lapata, M. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 1797–1807.

Ning, L.; Liu, L.; Wu, J.; Wu, N.; Berlowitz, D.; Prakash, S.; Green, B.; O'Banion, S.; and Xie, J. 2024. User-LLM: Efficient LLM Contextualization with User Embeddings. *arXiv preprint arXiv:2402.13598*.

Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C. L.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray, A.; Schulman, J.; Hilton, J.; Kelton, F.; Miller, L.; Simens, M.; Askell, A.; Welinder, P.; Christiano, P.; Leike, J.; and Lowe, R. 2022. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:2203.02155*.

Rao, H.; Leung, C.; and Miao, C. 2023. Can chatgpt assess human personalities? a general evaluation framework. *arXiv preprint arXiv:2303.01248*.

Roit, P.; Ferret, J.; Shani, L.; Aharoni, R.; Cideron, G.; Dadashi, R.; Geist, M.; Girgin, S.; Hussenot, L.; Keller, O.; et al. 2023. Factually consistent summarization via reinforcement learning with textual entailment feedback. *arXiv preprint arXiv:2306.00186*.

Salemi, A.; Mysore, S.; Bendersky, M.; and Zamani, H. 2024. LaMP: When Large Language Models Meet Personalization. *arXiv preprint arXiv:2304.11406*.

Wang, C.; Wu, N.; Ning, L.; Wu, J.; Liu, L.; Xie, J.; O'Banion, S.; and Green, B. 2024. UserSumBench: A Benchmark Framework for Evaluating User Summarization Approaches. *arXiv preprint arXiv:2408.16966*.

Williams, R. J. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4): 229–256.

Wu, B.; Shi, Z.; Rahmani, H. A.; Ramineni, V.; and Yilmaz, E. 2024a. Understanding the Role of User Profile in the Personalization of Large Language Models. *arXiv preprint arXiv:2406.17803*.

Wu, L.; Zheng, Z.; Qiu, Z.; Wang, H.; Gu, H.; Shen, T.; Qin, C.; Zhu, C.; Zhu, H.; Liu, Q.; Xiong, H.; and Chen, E. 2024b. A Survey on Large Language Models for Recommendation. *arXiv preprint arXiv:2305.19860*.

Yan, A.; He, Z.; Li, J.; Zhang, T.; and McAuley, J. 2022. Personalized Showcases: Generating Multi-Modal Explanations for Recommendations. *arXiv preprint arXiv:2207.00422*.

Yang, K.; Klein, D.; Celikyilmaz, A.; Peng, N.; and Tian, Y. 2023. Rlcd: Reinforcement learning from contrast distillation for language model alignment. *arXiv preprint arXiv:2307.12950*.

Zhang, T.; Kishore, V.; Wu, F.; Weinberger, K. Q.; and Artzi, Y. 2020. BERTScore: Evaluating Text Generation with BERT. In *International Conference on Learning Representations*.