COLLABLLM: From Passive Responders to Active Collaborators (Extended Abstract)*

Shirley Wu[1], **Michel Galley**[2], **Baolin Peng**[2], **Hao Cheng**[2], **Gavin Li**[1], **Yao Dou**[3], **Weixin Cai**[4], **James Zou**[1], **Jure Leskovec**[1], **Jianfeng Gao**[2]
[1]Stanford University
[2]Microsoft Research
[3]Georgia Tech
[4]Atlassian
shirwu@cs.stanford.edu, mgalley@microsoft.com

Abstract

Large Language Models are typically trained with next-turn rewards, limiting their ability to optimize for long-term interaction. As a result, they often respond passively to ambiguous or open-ended user requests, failing to help users reach their ultimate intents and leading to inefficient conversations. To address these limitations, we introduce COLLABLLM, a novel and general training framework that enhances multiturn human-LLM collaboration. Its key innovation is a collaborative simulation that estimates the long-term contribution of responses using *Multiturn-aware Rewards*. By reinforcement fine-tuning these rewards, COLLABLLM goes beyond responding to user requests, and actively uncovers user intent and offers insightful suggestions—a key step towards more human-centered AI. We also devise a multiturn interaction benchmark with three challenging tasks such as document creation. COLLABLLM significantly outperforms our baselines with averages of 18.5% higher task performance and 44.3% improved interactivity as scored by LLM judges. Finally, we conduct a large user study with 201 judges, where COLLABLLM increases user satisfaction by 17.6% and reduces user spent time by 10.4%.

1 Introduction

Modern Large Language Models (LLMs) excel at generating high-quality single-turn responses when given well-specified inputs. However, real-world users often do not fully articulate their intents and sometimes initiate conversations with an imprecise understanding of their own needs [Taylor, 1968]. As a result, users routinely refine their requests post hoc through iterative corrections, which can increase frustration, hinder effective task completion, and reduce conversational efficiency [Amershi *et al.*, 2019; Zamfirescu-Pereira *et al.*, 2023; Wang *et al.*, 2024; Kim *et al.*, 2024]. Therefore, an open problem is to train models that guide users in clarifying their intents and achieving their goals.

Here we introduce **COLLABLLM**, a novel and general training framework that improves the ability of LLMs to effectively collaborate with humans in multiturn scenarios [Gao *et al.*, 2019; Balog and Zhai, 2023; Rahmani *et al.*, 2023]. The key innovation of COLLABLLM is to promote LLMs' forward-looking behavior that leads to long-term collaboration gains (Figure 1). We introduce a collaborative simulation module that samples future conversations with users to estimate the long-term impact of model responses across multiple turns, a measure we term the *Multiturn-aware Reward (MR)*. The MR function evaluates responses by incorporating both extrinsic metrics, such as task-specific success, and intrinsic metrics, such as efficiency, to holistically assess collaboration quality (*cf.* Section 3). By fine-tuning with RL algorithms [Rafailov *et al.*, 2023; Schulman *et al.*, 2017] on MRs, COLLABLLM promotes responses that lead to better task completion and efficiency in later conversation stages.

We also introduce three challenging multiturn tasks for training and evaluation in simulated environments: `MediumDocEdit-Chat` (document editing), `BigCodeBench-Chat` (code generation), and `MATH-Chat` (math problem solving). On the three test sets, our approach improves task accuracy metrics by 18.5% and interactivity by 44.3% over our best baselines. We further perform a large-scale and real-world user study with 201 Amazon Mechanical Turkers, who are asked to write documents with an AI assistant; COLLABLLM achieves impressive improvement with 17.6% increase in user satisfaction and yields user time savings of 10.4%.

2 Problem Formulation

In contrast to many existing tasks that are single-turn and require no human involvement beyond the initial query, our problem formulation reflects a real-world setting in which a user's underlying (implicit) goal is defined as g in a multiturn conversational task. The conversation unfolds over multiple turns $t_j = \{u_j, m_j\}$, where u_j is the user input and m_j is the model's response at each turn $j = 1, \ldots, K$, where K is the number of turns in the conversation.

At the j-th turn, the model generates its response based

*This paper is an extended abstract of Wu *et al.* [2025]; the full paper additionally provides ablations, a case study, out-of-domain generalization results, and qualitative user feedback.

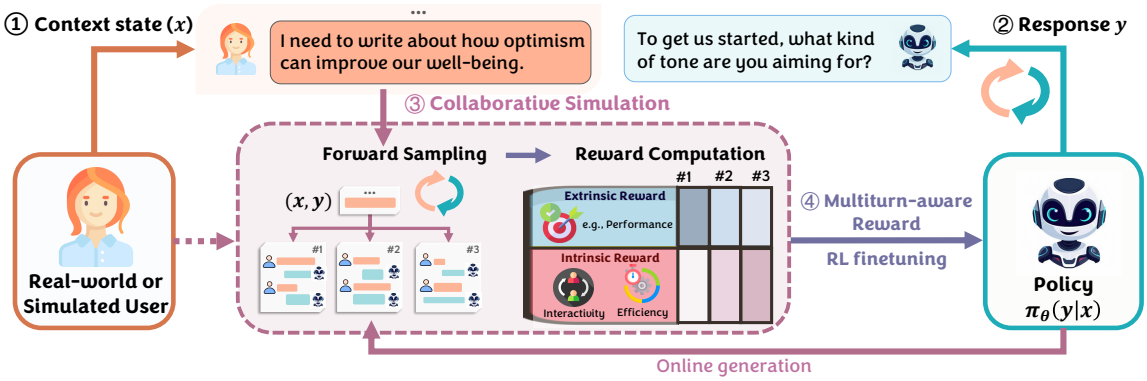

Figure 1: CollabLLM Framework: Given a context ①, the model generates a response ② to maximize long-term collaboration gains, termed *Multiturn-aware Rewards* (MR). During training, MRs are estimated via ③ collaborative simulation, which forward-samples conversations with simulated users. Finally, ④ reinforcement fine-tuning is applied using the MRs.

on the previous conversation turns $t_{1:j-1} = \{t_1, \ldots, t_{j-1}\}$ and the current user response u_j. For simplicity, we define historical conversation at j-th turn as $t_j^h = t_{1:j-1} \cup \{u_j\}$, therefore, $m_j = M(t_j^h)$. The objective is to generate a sequence of model responses $\{m_j\}_{j=1}^K$ that effectively and efficiently achieve goal g, where goal achievement is assessed based on user satisfaction or an external evaluation function. Formally, we define the objective as $R^*(t_{1:K} \mid g)$, where R^* incorporates the achievement of task success and user experience factors such as time cost.

3 Unified Collaborative LLM Training

Key Motivations. Established LLM training frameworks, such as Reinforcement Learning from Human Feedback (RLHF) [Ouyang *et al.*, 2022], focus on maximizing immediate rewards for single-turn tasks. This causes a misalignment between their single-turn objective and real-world multiturn objective $R^*(t_{1:K} \mid g)$. Precisely, the model's accumulative single-turn reward $\sum_{j=1}^{j=K} R(m_j \mid t_j^h)$ may not imply a higher final reward $R^*(t_{1:K} \mid g)$. For example, if the user's goal g is to write an engaging article, an RLHF model may produce locally helpful sections (e.g., an introduction or a conclusion) that fail to cohere across turns.

Instead, effective multiturn collaboration requires model responses that optimally contribute to the final reward. The model should align its responses with the user's goal g by considering their impact on the trajectory $t_{1:K}$—e.g., asking "*Should I maintain an engaging tone in the conclusion like the introduction?*" rather than drafting in isolation.

3.1 Multiturn-aware Rewards

Our key insight is that effective multiturn collaboration relies on forward-looking strategies. Given a context ①, the model should consider how its response ② influences the subsequent turns of the conversation. To capture this, we design a ③ collaborative simulation module to estimate this impact. By ④ fine-tuning to distinguish between future conversations from different responses, the model better aligns with the overarching goal g. This aligns with causal effect estimation [Pearl, 2009; Pearl *et al.*, 2016]: MR is an interventional estimate of a response's causal contribution to long-term reward, unlike trajectory-level methods that learn from observed conversations. More specifically, we define:

Multiturn-aware Reward (MR): *The multiturn-aware reward for model response m_j at the j-th turn is given by:*

$$\begin{aligned} &\mathrm{MR}(m_j \mid t_j^h, g) \\ &= \mathbb{E}_{t_j^f \sim P(t_{j+1:K} \mid t_j^h \cup \{m_j\})} R^*(t_j^h \cup \{m_j\} \cup t_j^f \mid g) \\ &= \mathbb{E}_{t_j^f \sim P(t_j^f \mid t_{1:j})} R^*(t_{1:j} \cup t_j^f \mid g), \end{aligned} \tag{1}$$

where $t_{1:j}$ denotes the conversation history up to and including the j-th turn (equivalent to $t_j^h \cup \{m_j\}$), and $t_j^f = t_{j+1:K}$ represents the forward trajectory of turns following the j-th turn. The distribution $P(t_j^f \mid t_{1:j})$ models the possible forward conversations conditioned on the prior conversation history.

Computing Equation 1 requires two components, detailed below: **(a) a conversation-level reward function** $R^*(t \mid g)$ to evaluate an arbitrary multiturn conversation t, and **(b) a sampling strategy** for forward conversations $P(t_j^f \mid t_{1:j})$.

Conversation-level Reward Function

We approximate the conversation-level reward $R^*(t \mid g)$ with a combination of extrinsic (goal-specific) and intrinsic (goal-agnostic) metrics:

$$R^*(t \mid g) \simeq R_{\text{ext}}(t, g) + R_{\text{int}}(t), \tag{2}$$

where $R_{\text{ext}}(t, g)$ focuses on task success, and $R_{\text{int}}(t)$ evaluates user experience including efficiency and engagement.

- **Extrinsic Reward** $R_{\text{ext}}(t, g)$ measures how well the conversation achieves the user's goal g. Formally:

$$R_{\text{ext}}(t, g) = S(\mathrm{Extract}(t), y_g), \tag{3}$$

where $\mathrm{Extract}(t)$ extracts the final solution or response from the conversation t, especially for tasks requiring revisions or multi-step answers. y_g is the reference solution for the goal g, *e.g.,* the ground truth solution for a math problem. And $S(\cdot, \cdot)$ evaluates task-specific metrics like accuracy or similarity. This ensures the conversation contributes directly to achieving the desired goal.

- **Intrinsic Reward** $R_{\text{int}}(t)$ prioritizes conversations that enhance user experience, defined as:

$$R_{\text{int}}(t) = -\min[\lambda \cdot \text{TokenCount}(t), 1] + R_{\text{LLM}}(t), \quad (4)$$

where we encourage conversational efficiency by penalizing excessive tokens that users read and write, with λ controlling the penalty severity. This efficiency measure is bounded by 1 to maintain balance with other metrics. The second term, $R_{\text{LLM}}(t)$, is assigned by an LLM-based judge [Zheng *et al.*, 2023] on a 0–1 scale, evaluating user-valued objectives such as engagement / interactivity. Notably, additional conversational aspects, such as clarity, can be further integrated into the objective.

Forward Sampling

To compute Eq. 1, we require samples from $P(t_j^f \mid t_{1:j})$, the distribution of forward conversation conditioned on the conversation history. A simple approach is to use Monte Carlo sampling, where the conversation is extended turn-by-turn until it concludes. However, this can be computationally expensive for computing reward for every model response. For a scalable approximation, we introduce a window size w as a hyperparameter to limit the maximum number of forward turns considered in t_j^f (we use $w = 2$ in all experiments). This reduces the computational cost while maintaining sufficient context.

Gathering forward conversations from human participants at scale is impractical, so we introduce a user simulator U.

User Simulator: *A user simulator $U : \mathcal{T} \to \mathcal{U}$ maps a conversation history t to a user response u, generating a distribution $P(u \mid t)$ that simulates realistic user behavior.*

Specifically, we prompt an LLM to role-play as users (each with an implicit goal g), following the same language style as previous user turns and injecting typical user behaviors such as evolving needs, limited background knowledge, or clarification requests [Park *et al.*, 2024]. Unlike prior work that uses simulators mainly for synthetic dialogue data generation [Shi *et al.*, 2019; Tseng *et al.*, 2021; Hong *et al.*, 2023; Hu *et al.*, 2023; Faltings *et al.*, 2023], CollabLLM leverages them in forward sampling to estimate long-term response effects.

3.2 Optimization & Synthetic Datasets

With the conversation-level reward function and forward sampling strategy, we can compute MR for any model response without requiring an additional reward model, which is often costly and slow to train. Further, we employ reinforcement learning (RL) methods such as PPO [Schulman *et al.*, 2017] and DPO [Rafailov *et al.*, 2023] to guide the model in multiturn conversations. Moreover, MR can generate high-quality synthetic conversations for both supervised fine-tuning (SFT) and DPO: for SFT, it iteratively selects top-ranked responses to build realistic, goal-directed conversation histories; for DPO, it constructs pairwise comparisons by ranking responses at each turn, distinguishing "chosen" and "rejected" pairs based on MR scores.

4 Experimental Setup

For fine-tuning and evaluation, we create three multiturn datasets using publicly available data across diverse domains [Hendrycks *et al.*, 2021; Zhuo *et al.*, 2024; Chiusano, 2024]: collaborative document editing, coding problem assistance, and multiturn mathematics problem solving.

To build a multiturn environment, we employ GPT-4o-mini as a user simulator LLM to role-play realistic user behaviors, given the target problem and conversation history. The three datasets are:

`MediumDocEdit-Chat`: Document editing requires iterative feedback and refinements across multiple turns to ensure coherence and alignment with user intent. We sample 100 Medium articles as goal documents, which are summarized into target problems to guide the user simulator. After each interaction, task performance is evaluated using the BLEU score, measuring similarity between the extracted document and the original articles.

`BigCodeBench-Chat`: Coding tasks inherently require multiturn interactions, such as clarifying requirements and debugging. We sample 600 coding problems from BigCodeBench [Zhuo *et al.*, 2024] as the target problems given to the user simulator. For evaluation, we compute the average Pass Rate (PR) of code at the end of the interactions.

`MATH-Chat`: Math problem solving often requires addressing implicit assumptions, verifying intermediate steps, and clarifying reasoning. We sample 200 level-5 math problems from MATH [Hendrycks *et al.*, 2021] to prompt the user simulator, which interacts with the LLMs. Task success is measured by the accuracy (ACC) of the final solution.

In addition to the above task-specific metrics, we incorporate two task-agnostic scores across all datasets: **1) Average Token Count**, which quantifies the average number of tokens generated by the LLM per conversation, reflecting interaction efficiency. **2) Interactivity (ITR)**, which evaluates engagement levels using an LLM judge (Claude-3.5-Sonnet), with scores rescaled to an upper bound of 1.

Fine-tuning CollabLLMs. CollabLLMs are based on Llama-3.1-8B-Instruct [Llama Team, 2024] with LoRA fine-tuning [Hu *et al.*, 2022]. We train four model variants: **1) Offline models**: SFT and Offline DPO are fine-tuned on pre-generated multiturn conversational datasets (500 conversations per task) guided by Multiturn-aware Rewards (MR) (*cf.* Section 3.2). **2) Online models**: PPO and Online DPO are further trained from the SFT and Offline DPO models, respectively. During online fine-tuning, the model joins the collaborative simulation to compute MRs, which dynamically adjust model preference.

Baselines. We compare CollabLLMs against (1) the pre-trained Llama-3.1-8B-Instruct (*Base*), (2) the base model with proactive prompt engineering (*Proactive Base*), which encourages follow-up and clarification questions.

5 Results of Simulated Experiments

We present the results in Table 1 and the takeaways are:

Prompt engineering is helpful, but limited in terms of performance gains and flexibility. Proactive Base improves

	Method	MediumDocEdit-Chat			BigCodeBench-Chat			MATH-Chat		
		BLEU ↑	#Tok(k) ↓	ITR ↑	PR ↑	#Tok(k) ↓	ITR ↑	ACC ↑	#Tok(k) ↓	ITR ↑
	Base	32.2	2.49	46.0	9.3	1.59	22.0	11.0	3.40	44.0
	Proactive Base	35.0	2.18	62.0	11.0	1.51	33.7	12.5	2.90	46.0
CollabLLM	SFT	35.2	2.21	68.0	11.7	1.35	42.0	13.5	2.88	58.0
CollabLLM	PPO	**38.5**	**2.00**	78.0	**14.0**	1.35	40.7	13.0	2.59	52.0
CollabLLM	Offline DPO	36.4	2.15	82.0	12.3	1.35	46.7	15.5	2.40	50.0
CollabLLM	Online DPO	36.8	**2.00**	**92.0**	13.0	**1.31**	**52.0**	**16.5**	**2.37**	**60.0**
Rel. Improv.		5.14%	8.25%	48.3%	18.2%	13.2%	54.3%	32.0%	18.3%	30.4%

Table 1: Evaluation results on our multiturn datasets. The best in each column is **bolded**. *Rel. Improv.* reports the relative improvement of CollabLLM (Online DPO) over Proactive Base.

base model performance by encouraging follow-up questions and clarifications. For example, it increases BLEU on `MediumDocEdit-Chat` from 32.2% to 35.0% and reduces read tokens by 0.31k compared to the base model. However, these gains are modest and do not fully address the challenges of multiturn collaboration. We observe that prompting strategies remain rigid, relying on predefined instructions rather than adapting dynamically to user needs. For instance, the model sometimes asks unnecessary clarification questions that disrupt the conversation flow.

CollabLLM improves task performance, efficiency, and engagement. Averaged across the three datasets, CollabLLM (Online DPO) achieves 18.5% higher task-specific performance, 13.3% more efficient conversations, and 44.3% enhanced interactivity over Proactive Base (Table 1, last row). For `MediumDocEdit-Chat`, the Online DPO model increases ITR from 0.46 to 0.92. For `MATH-Chat`, Online DPO decreases token count per conversation from 3.40k to 2.37k.

6 Real-world User Study

Setup. We conduct a large-scale user study using Amazon Mechanical Turk with 201 participants. Each participant is assigned a document type—randomly selected to be either blog post, creative writing, or personal statement—and chooses a topic from a predefined set. Participants then engage in at least eight turns of conversation with an anonymized AI assistant, which can be Base, Proactive Base, or CollabLLM. We also record the total interaction duration to assess efficiency. Participants rate the final document quality and overall interaction on a 1–10 scale.

Results. CollabLLM achieves an average document quality score of 8.50, a 17.6% relative gain over Base. Specifically, 91.4% of participants rate CollabLLM's document quality as "good" (score 8–9), and 56.9% as "very good" (score 9–10), compared to 88.5% and 39.3% for Base (Llama-3.1-8B-Instruct), respectively. Similarly, 63.8% of participants find CollabLLM highly engaging, while only 42.6% report the same for Base. Moreover, CollabLLM improves task efficiency, reducing time spent by 10.4% compared to the Base model and by 15.6% relative to Proactive Base. Overall, CollabLLM enhances collaboration through iterative refinement; future work should improve personalization, creativity, and real-time knowledge integration.

7 Related Work

Non-collaborative LLM training. Existing LLM training frameworks, including pre-training, supervised fine-tuning (SFT), and reinforcement learning (RL) [Rafailov *et al.*, 2023; Schulman *et al.*, 2017; Ouyang *et al.*, 2022; Lee *et al.*, 2024], primarily optimize for next-turn response quality. While effective for single-turn objectives, these approaches fail to capture how responses influence user intent discovery and task success [Amershi *et al.*, 2019; Zamfirescu-Pereira *et al.*, 2023; Wang *et al.*, 2024; Kim *et al.*, 2024].

Learning-based methods for multiturn interaction. Beyond prompting, prior studies have explored supervised fine-tuning [Andukuri *et al.*, 2024], RL fine-tuning [Chen *et al.*, 2024; Zamani *et al.*, 2020; Erbacher and Soulier, 2023], and active learning [Pang *et al.*, 2024] to train models to ask clarification questions. Several studies extend RLHF to multiturn settings by optimizing trajectory-level rewards [Shani *et al.*, 2024; Zhou *et al.*, 2024; Gao *et al.*, 2024; Shi *et al.*, 2024; Zhang *et al.*, 2025]. Other works [Xu *et al.*, 2023; Deng *et al.*, 2024] leverage self-chat or self-play to enhance model adaptation. However, these methods primarily rely on post-hoc trajectory-level data, learning from observed conversations rather than explicitly modeling the causal effect of individual responses on task success. Additionally, they often overlook open-ended tasks such as document generation [Faltings *et al.*, 2023; Jiang *et al.*, 2024], where user responses can be highly diverse, and users may have limited capacity to read and refine lengthy model outputs.

8 Conclusion

Multiturn human-LLM collaborations are increasingly prevalent in real-world applications. Foundation models should act as collaborators rather than passive responders, actively uncovering user intents in open-ended and complex tasks—an area where current LLMs fall short. The key insight of CollabLLM is making LLMs more multiturn-aware by using forward sampling to estimate the long-term impact of responses. Through simulated and real-world evaluations, we show CollabLLM is effective, efficient, and engaging across three diverse tasks; the full paper [Wu *et al.*, 2025] further demonstrates zero-shot generalization to out-of-domain tasks (Abg-CoQA), advancing human-centered LLMs.

References

[Amershi *et al.*, 2019] Saleema Amershi, Daniel S. Weld, Mihaela Vorvoreanu, Adam Fourney, Besmira Nushi, Penny Collisson, Jina Suh, Shamsi T. Iqbal, Paul N. Bennett, Kori Inkpen, Jaime Teevan, Ruth Kikin-Gil, and Eric Horvitz. Guidelines for human-ai interaction. In *CHI*. ACM, 2019.

[Andukuri *et al.*, 2024] Chinmaya Andukuri, Jan-Philipp Fränken, Tobias Gerstenberg, and Noah D. Goodman. Star-gate: Teaching language models to ask clarifying questions. *CoLM*, 2024.

[Balog and Zhai, 2023] Krisztian Balog and ChengXiang Zhai. Rethinking conversational agents in the era of llms: Proactivity, non-collaborativity, and beyond. In *SIGIR-AP*, 2023.

[Chen *et al.*, 2024] Maximillian Chen, Ruoxi Sun, Sercan Ö. Arik, and Tomas Pfister. Learning to clarify: Multi-turn conversations with action-based contrastive self-training. *CoRR*, abs/2406.00222, 2024.

[Chiusano, 2024] Fabio Chiusano. Medium articles dataset. https://huggingface.co/datasets/fabiochiu/medium-articles, 2024. A dataset of Medium articles collected through web scraping, including metadata such as titles, authors, publication dates, tags, and content.

[Deng *et al.*, 2024] Yang Deng, Wenxuan Zhang, Wai Lam, See-Kiong Ng, and Tat-Seng Chua. Plug-and-play policy planner for large language model powered dialogue agents. In *ICLR*, 2024.

[Erbacher and Soulier, 2023] Pierre Erbacher and Laure Soulier. CIRCLE: multi-turn query clarifications with reinforcement learning. *CoRR*, abs/2311.02737, 2023.

[Faltings *et al.*, 2023] Felix Faltings, Michel Galley, Kianté Brantley, Baolin Peng, Weixin Cai, Yizhe Zhang, Jianfeng Gao, and Bill Dolan. Interactive text generation. In *EMNLP*, 2023.

[Gao *et al.*, 2019] Jianfeng Gao, Michel Galley, and Lihong Li. Neural approaches to conversational AI. *Found. Trends Inf. Retr.*, 13, 2019.

[Gao *et al.*, 2024] Zhaolin Gao, Wenhao Zhan, Jonathan D. Chang, Gokul Swamy, Kianté Brantley, Jason D. Lee, and Wen Sun. Regressing the relative future: Efficient policy optimization for multi-turn rlhf, 2024.

[Hendrycks *et al.*, 2021] Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *NeurIPS*, 2021.

[Hong *et al.*, 2023] Joey Hong, Sergey Levine, and Anca D. Dragan. Zero-shot goal-directed dialogue via RL on imagined conversations. *CoRR*, 2023.

[Hu *et al.*, 2022] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *ICLR*, 2022.

[Hu *et al.*, 2023] Zhiyuan Hu, Yue Feng, Anh Tuan Luu, Bryan Hooi, and Aldo Lipani. Unlocking the potential of user feedback: Leveraging large language model as user simulators to enhance dialogue system. In *CIKM*. ACM, 2023.

[Jiang *et al.*, 2024] Yucheng Jiang, Yijia Shao, Dekun Ma, Sina J. Semnani, and Monica S. Lam. Into the unknown unknowns: Engaged human learning through participation in language model agent conversations. *arXiv*, abs/2408.15232, 2024.

[Kim *et al.*, 2024] Yoonsu Kim, Jueon Lee, Seoyoung Kim, Jaehyuk Park, and Juho Kim. Understanding users' dissatisfaction with chatgpt responses: Types, resolving tactics, and the effect of knowledge level. In *IUI*. ACM, 2024.

[Lee *et al.*, 2024] Harrison Lee, Samrat Phatale, Hassan Mansoor, Thomas Mesnard, Johan Ferret, Kellie Lu, Colton Bishop, Ethan Hall, Victor Carbune, Abhinav Rastogi, and Sushant Prakash. RLAIF vs. RLHF: scaling reinforcement learning from human feedback with AI feedback. In *ICML*, 2024.

[Llama Team, 2024] AI @ Meta Llama Team. The llama 3 herd of models, 2024.

[Ouyang *et al.*, 2022] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022.

[Pang *et al.*, 2024] Jing-Cheng Pang, Heng-Bo Fan, Pengyuan Wang, Jiahao Xiao, Nan Tang, Si-Hang Yang, Chengxing Jia, Sheng-Jun Huang, and Yang Yu. Empowering language models with active inquiry for deeper understanding. *CoRR*, abs/2402.03719, 2024.

[Park *et al.*, 2024] Joon Sung Park, Carolyn Q. Zou, Aaron Shaw, Benjamin Mako Hill, Carrie Cai, Meredith Ringel Morris, Robb Willer, Percy Liang, and Michael S. Bernstein. Generative agent simulations of 1,000 people, 2024.

[Pearl *et al.*, 2016] Judea Pearl, Madelyn Glymour, and Nicholas P. Jewell. *Causal Inference in Statistics: A Primer*. John Wiley & Sons, 2016.

[Pearl, 2009] Judea Pearl. *Causality: Models, Reasoning and Inference*. Cambridge University Press, 2nd edition, 2009.

[Rafailov *et al.*, 2023] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *NeurIPS*, 2023.

[Rahmani *et al.*, 2023] Hossein A. Rahmani, Xi Wang, Yue Feng, Qiang Zhang, Emine Yilmaz, and Aldo Lipani. A survey on asking clarification questions datasets in conversational systems. In *ACL*, 2023.

[Schulman *et al.*, 2017] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *CoRR*, abs/1707.06347, 2017.

[Shani *et al.*, 2024] Lior Shani, Aviv Rosenberg, Asaf B. Cassel, Oran Lang, Daniele Calandriello, Avital Zipori, Hila Noga, Orgad Keller, Bilal Piot, Idan Szpektor, Avinatan Hassidim, Yossi Matias, and Rémi Munos. Multi-turn reinforcement learning from preference human feedback. *CoRR*, abs/2405.14655, 2024.

[Shi *et al.*, 2019] Weiyan Shi, Kun Qian, Xuewei Wang, and Zhou Yu. How to build user simulators to train rl-based dialog systems. In *EMNLP-IJCNLP*. Association for Computational Linguistics, 2019.

[Shi *et al.*, 2024] Wentao Shi, Mengqi Yuan, Junkang Wu, Qifan Wang, and Fuli Feng. Direct multi-turn preference optimization for language agents. In *EMNLP*. Association for Computational Linguistics, 2024.

[Taylor, 1968] Robert S Taylor. Question-negotiation and information seeking in libraries. *College & research libraries*, 29(3):178–194, 1968.

[Tseng *et al.*, 2021] Bo-Hsiang Tseng, Yinpei Dai, Florian Kreyssig, and Bill Byrne. Transferable dialogue systems and user simulators. In *ACL/IJCNLP*. Association for Computational Linguistics, 2021.

[Wang *et al.*, 2024] Jiayin Wang, Weizhi Ma, Peijie Sun, Min Zhang, and Jian-Yun Nie. Understanding user experience in large language model interactions. *CoRR*, abs/2401.08329, 2024.

[Wu *et al.*, 2025] Shirley Wu, Michel Galley, Baolin Peng, Hao Cheng, Gavin Li, Yao Dou, Weixin Cai, James Zou, Jure Leskovec, and Jianfeng Gao. Collabllm: From passive responders to active collaborators. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, 2025.

[Xu *et al.*, 2023] Canwen Xu, Daya Guo, Nan Duan, and Julian J. McAuley. Baize: An open-source chat model with parameter-efficient tuning on self-chat data. In *EMNLP*, 2023.

[Zamani *et al.*, 2020] Hamed Zamani, Susan T. Dumais, Nick Craswell, Paul N. Bennett, and Gord Lueck. Generating clarifying questions for information retrieval. In *WWW*, 2020.

[Zamfirescu-Pereira *et al.*, 2023] J. D. Zamfirescu-Pereira, Richmond Y. Wong, Bjoern Hartmann, and Qian Yang. Why johnny can't prompt: How non-ai experts try (and fail) to design LLM prompts. In *CHI*. ACM, 2023.

[Zhang *et al.*, 2025] Chen Zhang, Xinyi Dai, Yaxiong Wu, Qu Yang, Yasheng Wang, Ruiming Tang, and Yong Liu. A survey on multi-turn interaction capabilities of large language models. *arXiv preprint arXiv:2501.09959*, 2025.

[Zheng *et al.*, 2023] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena. In *NeurIPS*, 2023.

[Zhou *et al.*, 2024] Yifei Zhou, Andrea Zanette, Jiayi Pan, Sergey Levine, and Aviral Kumar. Archer: Training language model agents via hierarchical multi-turn RL. In *ICML*, 2024.

[Zhuo *et al.*, 2024] Terry Yue Zhuo, Minh Chien Vu, Jenny Chim, Han Hu, Wenhao Yu, Ratnadira Widyasari, Imam Nur Bani Yusuf, Haolan Zhan, Junda He, Indraneil Paul, et al. Bigcodebench: Benchmarking code generation with diverse function calls and complex instructions. *arXiv preprint arXiv:2406.15877*, 2024.