RecLLMSim: A Comprehensive Task-based Recommendation Conversation Dataset Generated by Large Language Models

Anonymous ACL submission

Abstract

Conversational systems have garnered significant attention and importance in recent years. However, collecting conversational datasets has traditionally been a time-consuming and labor-intensive process. With the advent of large language models (LLMs), there is a growing interest in using them to generate synthetic datasets. Given LLMs' strong role-playing capabilities, they hold the potential to simulate users effectively. This capability allows for the automated generation of conversations, with LLMs acting as both users and assistants across various scenarios.

Our study proposes a framework, designed to generate task-based recommendation conversation datasets across multiple scenarios with LLMs. We have created a comprehensive conversational dataset using this framework, and the dataset is named *RecLLMSim*[1]. We conducted extensive experiments to measure the quality of the user simulator and the assistant, and annotated user intent and hallucinations to improve its usability. Experimental results demonstrate that using LLMs as user simulators is a promising approach. Besides, the generated *RecLLMSim* dataset can be adapted for various tasks such as user profiling and simulation, offering a rich resource for further advancements in conversational systems.

1 Introduction

Recently, conversational systems have surged in significance and widespread attention as they play a pivotal role in enhancing human-computer interactions. As these conversation systems have been applied to various scenarios in our daily lives and work, there is a growing demand for high-quality conversational datasets to help improve conversational systems.

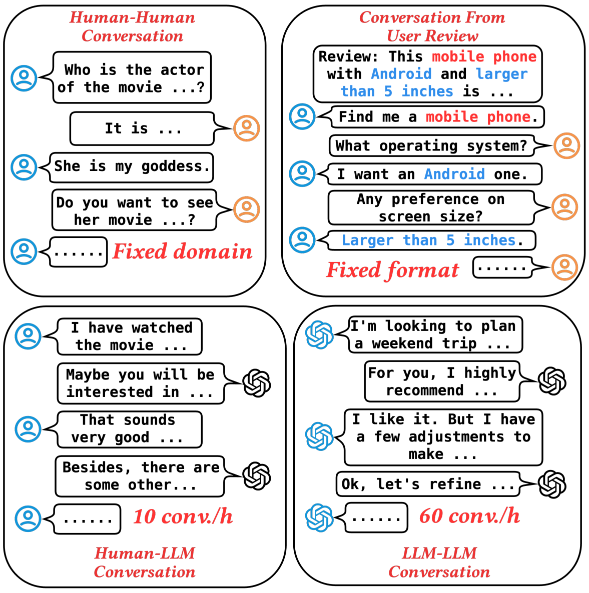
Human-Human Conversation
Who is the actor of the movie ...?
It is ...
She is my goddess.
Do you want to see her movie ...?
......
Fixed domain
Conversation From User Review
Review: This mobile phone with Android and larger than 5 inches is ...
Find me a mobile phone.
What operating system?
I want an Android one.
Any preference on screen size?
Larger than 5 inches.
Fixed format
......
I have watched the movie ...
Maybe you will be interested in ...
That sounds very good ...
Besides, there are some other...
......
10 conv./h
Human-LLM Conversation
I'm looking to plan a weekend trip ...
For you, I highly recommend ...
I like it. But I have a few adjustments to make ...
Ok, let's refine ...
......
60 conv./h
LLM-LLM Conversation

Figure 1: Comparison of different types of conversation dataset construction methods. The method of generating conversations between LLMs (LLM-LLM conversation) is more diverse and efficient.

As shown in the upper part of Figure 1, traditionally, there are mainly two types of strategies to construct conversational datasets. The first type of method involves collecting conversations between humans, such as customer service interactions. Although this approach provides rich and diverse conversational data, it is either inefficient or limited to a single domain. The second type of method entails extracting keywords from user reviews or other static texts to construct synthetic *System Ask - User Respond* conversations. Although this technique can construct conversations more swiftly, it tends to generate rigid and formulaic texts.

Large language models (LLMs) are widely integrated into everyday life, encouraging researchers to leverage them to construct conversational datasets. Collecting user-LLM interactions offers a more realistic approach, but it requires

[1] The code of the framework and the dataset can be found in https://anonymous.4open.science/r/RecLLMSim-EF85.

hiring suitable users and is difficult to control conversation scenarios well. In contrast, LLMs have garnered significant attention for their use in synthetic data generation(Liu et al., 2024). Therefore, we propose simulating both sides of the conversations with LLMs, using predefined scenarios to ensure diversity.

However, generating LLM-LLM conversations presents its own challenges, such as ensuring LLMs understand human-defined scenarios, setting comprehensive user profiles for simulators, and verifying the quality of the generated utterances. To address these issues, we designed a novel conversation dataset generation framework, and conducted manual quality validation on the generated dataset *RecLLMSim* to confirm its reliability. In summary, the main contributions of this work can be summarized as follows:

- We propose an automated data collection framework, using LLMs as user simulators. Based on a three-step approach, it can automatically generate conversations across various customized task scenarios.
- Based on the framework, we generate a task-based recommendation conversation dataset, *RecLLMSim*. The dataset contains task-based recommendation conversations in four daily scenarios.
- We conducted multi-dimensional quality validation and manual annotations to confirm *RecLLMSim*'s applicability across different scenarios. Preliminary intent detection experiments and further discussion about its potential usage show the availability of the dataset.

2 Related Work

2.1 Conversational Dataset

The development of high-quality conversational datasets is crucial for advancing conversational systems. One traditional approach is collecting dialogues directly from human interactions, which is considered the gold standard due to its authenticity. A common method is to collect existing user dialogue data, such as JDDC(Chen et al., 2019; Zhao et al., 2021, 2022), E-commerce Dialogue Corpus(Zhang et al., 2018b). Another method involves setting up specific pipelines and employing crowdsourced annotators to generate dialogues, such as ReDial(Li et al., 2018), KdConv(Zhou et al., 2020), etc. Another approach is to extract keywords from user reviews or other static texts to construct *System Ask - User Respond* conversations(Zhang et al., 2018a).

More recently, researchers have explored leveraging LLMs to generate conversational datasets. ShareGPT and LMSYS-Chat-1M(Zheng et al., 2023) collect user-LLM conversations to construct a real-life conversational dataset. There are also conversational datasets generated by multi LLMs chatting to each other, such as UltraChat(Ding et al., 2023), Baize(Xu et al., 2023b), PEARL(Kim et al., 2024).

However, these methods are either inefficient or limited to fewer and simpler task scenarios. Our proposed framework enables rapid automated data generation across various customized task scenarios.

2.2 Synthetic Data Generation via LLMs

LLMs' in-context learning ability enables researchers to apply LLMs to synthetic data generation(Kaddour et al., 2023). Some researchers have experimented with manually engineered prompts and LLMs to annotate unlabeled data(Wang et al., 2021; Ding et al., 2022; Ye et al., 2022). Others have explored employing LLM for automated pairwise feedback annotation to assist in aligning other LLMs or the LLM itself(Lee et al., 2023; Yuan et al., 2024). In addition to using LLMs for automatic data annotation, some studies have leveraged LLMs to directly generate natural language instructions(Xu et al., 2023a; Wang et al., 2022) or complete data(Dai et al., 2023; Ding et al., 2023; Jeronymo et al., 2023).

However, most of these generated data do not consider user interaction behaviors, which is crucial in information retrieval.

2.3 User Simulation

User simulation can effectively approximate real user behavior at a lower cost, reducing the need for actual user data. Before the era of LLMs, there was already significant research on user simulation(Georgila et al., 2006; Biswas et al., 2012). The advent of LLMs has further propelled this field. Some researchers have explored using LLMs to simulate users for recommendation systems(Wang et al., 2023a; Zhang et al., 2024) or to propose a novel evaluation paradigm for conversational recommendation(Wang et al., 2023b; Zhu et al., 2024). Additionally, user simulation has been applied to

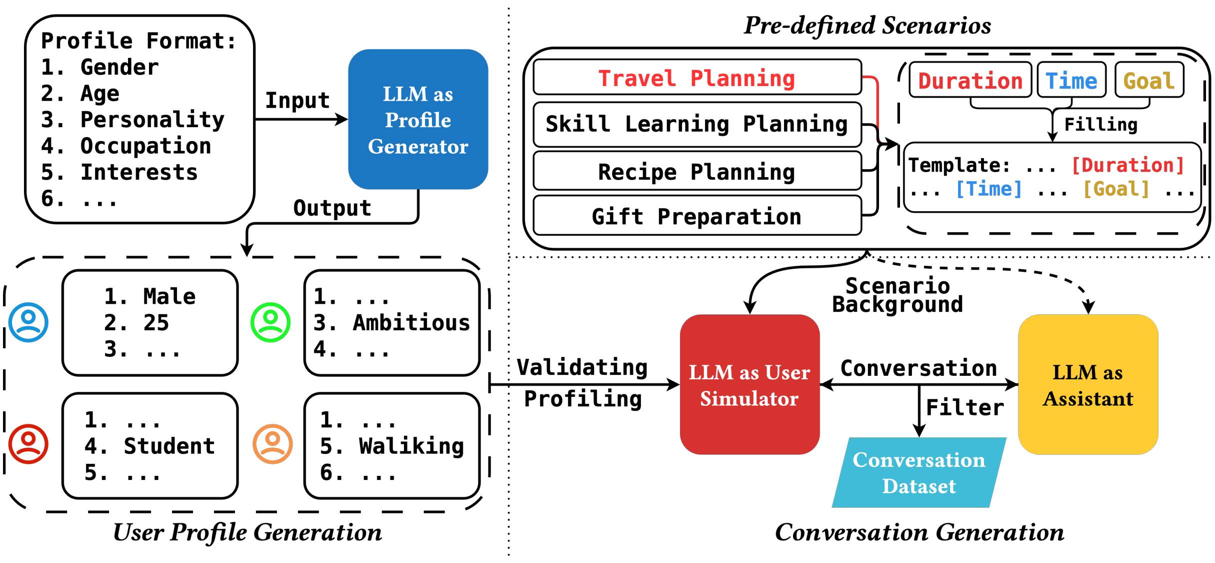

Figure 2: The overview of our framework to generate task-based recommendation conversation.

studies of social behavior(Park et al., 2023) and dialogue generation(Chan et al., 2023).

However, most work applying LLM user simulators to conversational systems focuses on improving evaluation methods. The few studies on dataset generation are often limited by specific task scenarios. Our framework, in contrast, focuses on using LLMs as user simulators to generate conversational data across a wide range of scenarios.

3 Framework

3.1 Framework Overview

Figure 2 has shown the overview of our conversation generation framework. Our framework leverages LLMs to facilitate the generation of conversational data across multiple scenarios. The process is structured as follows:

- **User Profile Generation**: We utilize LLMs to create user profiles. We provide specific fields that the LLMs need to fill, accompanied by examples of well-constructed profiles to guide the generation process.

- **Pre-defined Scenarios**: We manually define four distinct task scenarios. We establish several optional variables and construct a description template for each task. The various scenario descriptions, resulting from different combinations of these variables, will be used in the subsequent conversation generation process.

- **Conversation Generation**: We use the generated user profile and task scenario descriptions as input for the user simulator LLM, instructing it to follow predefined rules and engage in conversation with the assistant LLM. This interaction generates individual conversations. Upon completion, we perform a basic filtering to ensure the quality of the generated data.

We will detail each component in the following subsections.

3.2 User Profile Generation

No specific example
Please generate a user profile based on at least six aspects, including gender, age, personality, occupation, hobbies, and dislikes, that belongs to your own identity...

Example based on the initial generation by GPT-4		
Gender: Male	**Age**: 24	**Personality**: Outgoing and cheerful
Occupation: Software Engineer		
Hobbies: Enjoys coding and exploring the latest tech trends; likes outdoor activities such as hiking and rock climbing.		
Dislikes: Dislikes sitting in front of the TV to watch games or series.		

Example after many revisions and improvements		
Gender: Male	**Age**: 24	**Personality**: Outgoing and cheerful
Occupation: Software Engineer		
Daily Interests and Hobbies: Enjoys programming and exploring the latest tech trends; dislikes sitting in front of the TV for matches or series.		
Travel Habits: Enjoys outdoor activities like hiking and rock climbing; not fond of visiting many museums or art galleries.		
Dining Preferences: Loves sweets, especially Italian cuisine; dislikes Sichuan cuisine and other spicy foods.		
Spending Habits: Values product quality and can accept relatively higher prices.		
Other Aspects: Enjoys socializing and has a large social circle.		

Figure 3: Human-involved example generation.

To minimize the bias caused by our manual design, the user profiles we used in the dataset construction were all generated by GPT-4. Despite LLMs' extensive generalized knowledge and problem-solving capabilities, we still encountered difficulties in the user profile generation process, e.g., duplication, forgetfulness, etc. To overcome the difficulties, we require LLM to generate user profiles in terms of *gender, age, personality, occupation, daily interests and hobbies, travel habits, dining preferences, spending habits, and other aspects* to enhance the distinction and diversity of user-profiles. When the length of context reaches LLM's limitations, we re-emphasize the simplified requirements to continue generating to reduce the impact of LLM's forgetfulness. Moreover, to ensure the rationality of the examples we give, we have adopted human-involved example generation, i.e., we improve the given examples based on the generation results. The comparisons in Figure 3 show how the examples are improved.

3.3 Pre-defined Scenarios

In this work, we want to test and utilize LLMs' ability to understand and accomplish complicated tasks. Each task should be complex and difficult to solve within a single chat round and related to the user's preferences. Based on the above requirements for task complexity, we pre-defined four types of tasks: *travel planning*, *skill learning planning*, *recipe planning*, and *gift preparation*. We set several specific scenarios for each task. Taking *travel planning* as an example, we construct different scenarios from three optional variables: travel season, travel duration, and travel type (e.g., business trip, family trip). A scenario-specific task description is formed by filling the identified template with the abovementioned specific options. The template also contains some standard requirements for travel planning tasks, such as accommodations or transport.

A task description example for a travel planning scenario is as follows: "*You are planning a thirty-day-long trip this winter. The trip has no specific destination restrictions. You would like to use LLM to help refine your travel plans. During this conversation, you will need to make arrangements for accommodations, transportation, and itinerary details. Your ultimate travel plan should be as comprehensive as possible, ensuring that you can realistically follow it when the time comes.*"

In the same way as above, we similarly designed scenario-specific task descriptions for the other three tasks.

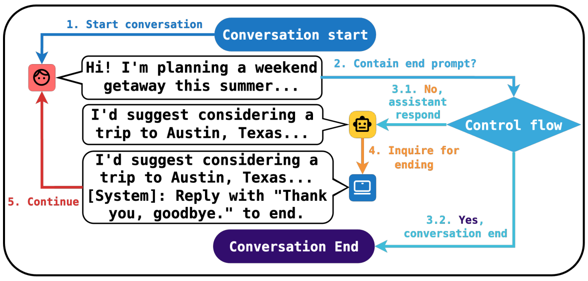
1. Start conversation
Conversation start
Hi! I'm planning a weekend getaway this summer...
2. Contain end prompt?
3.1. No, assistant respond
Control flow
I'd suggest considering a trip to Austin, Texas...
4. Inquire for ending
I'd suggest considering a trip to Austin, Texas... [System]: Reply with "Thank you, goodbye." to end.
5. Continue
3.2. Yes, conversation end
Conversation End

Figure 4: The conversation generation process.

3.4 Conversation Generation

Figure 4 illustrates the basic process of generating conversations using our framework. We used two LLMs to talk to each other for the generation. Before starting the conversation between the two LLMs, we provide the scenario-specific task description for the simulator and ask it to act as a user with the given user profile. Since the structure of LLMs does not contain a memory module and multiple conversations are entirely independent, the impact of all feedback occurs only within the current conversation and does not affect all other generations. We sent each other the simulator and assistant responses during the conversation to complete the automated conversation generation.

To make the end of the conversation controllable, we added a system-command-like end prompt at the end of each response from the assistant to the simulator, which contains a description of the command to end the conversation. The end prompt may look like "*[System]: If you wish to conclude this conversation, please reply with 'Thank you, goodbye.'*". Based on the above technique, we can achieve dynamic conversation length control by simply checking the simulator's responses.

It is worth noting that the quality of text generated by LLMs does not always perfectly match our requirements. Therefore, after obtaining the generated conversations, we performed a basic conversation quality validation to ensure that both the user simulator and assistant were playing and performing their roles correctly.

4 *RecLLMSim* Dataset

We generated a task-based recommendation conversation dataset called *RecLLMSim* based on the framework we proposed in the previous section.

4.1 Characteristics

We give the basic statistics of *RecLLMSim* in Table 1, including the scenarios, the number of conversations, the average number of rounds of conversations, and the average number of tokens of conversations.

Datasets	#scenarios	#conv.	Avg. #utt.	Avg. #tokens
Raw	22	2130	7.68	1191.35
Filtered	22	1857	7.36	1368.49

Table 1: Statistics of datasets.

We also analyzed the distribution of preferences exhibited by the user simulator in the conversations for each task. Specifically, we used GPT-4 to automatically analyze the simulator's preferences in each conversation and asked it to give three to five phrases for the conversation description. For tasks that have corresponding entries in our given user profile (i.e., travel planning and recipe planning), we merge the entries into the above preferences. We then synonymically merged all preference description phrases that appeared under each task and used word clouds to show the frequency of occurrence of different preference descriptions. As shown in Figure 5, the simulator exhibited a broad and diverse range of user preferences in conversations across different task scenarios.

4.2 Quality Verification

A high-quality conversational dataset must ensure the reasonableness of role-playing by both parties in the conversation. Therefore, we have designed multi-dimensional evaluation metrics for both the user simulator and the assistant. We mainly focus on two aspects: for the user simulator, we emphasize its ability to profile users and accurately align with real-world scenarios; for the assistant, we focus on the usefulness of responses and their coverage of a wide range of situations.

For user profiling of the simulator, we consider the following metrics:

- **Preference Alignment** evaluates how well the simulator adheres to the provided user profile. It ensures that the behavior and preferences exhibited by the user simulator align with the specified profile characteristics.

- **Additional Preferences** checks whether the simulator can exhibit additional interests and preferences beyond the given user profile.

- **Role-Playing Completeness** evaluates how effectively the simulator's inquiries and requests contribute to resolving the given task under the specific scenario. It ensures that the interactions are coherent and relevant to the task at hand.

To evaluate the usefulness of the assistant, we considered the following aspects:

- **Memorization** verifies whether the assistant forgets the user's requirements or requests throughout the conversation. It ensures that the assistant does not forget or overlook previously mentioned user requirements.

- **Detail Level** measures the level of detail in the assistant's responses and descriptions.

- **Practical Usefulness** evaluates the practicality and usefulness of the assistant's answers and suggestions. It determines how well the provided information can help in real-world scenarios.

- **Diversity** measures the variety and richness of the assistant's responses. It ensures that the answers are varied and not repetitive, promoting a richer conversation experience.

We employed a group of qualified crowdsourced annotators to evaluate the conversations across the abovementioned dimensions. Specifically, *Memorization* and *Additional Preference* were binary-rated (0 for absence, 1 for presence), while other metrics were rated on a three-point scale. The distribution of scores for each metric is detailed in Figure 6, and Table 2 presents the average scores for each metric across different tasks.

From the score distribution in Figure 6, it is evident that the responses of the assistants in the *RecLLMSim* dataset have been positively validated by the annotators across various metrics. The assistant's responses are generally detailed, practical, and diverse, with a strong memory consistency in recalling user requests. The user simulator generally aligns well with the given user profile and effectively assists in task completion. However, it rarely exhibits additional preferences.

Breaking it down by specific tasks, the user simulator performs best in the *travel planning* task, consistently providing role-play that closely aligns with the given user profile. The assistant performs notably well in the *skills learning planning* and

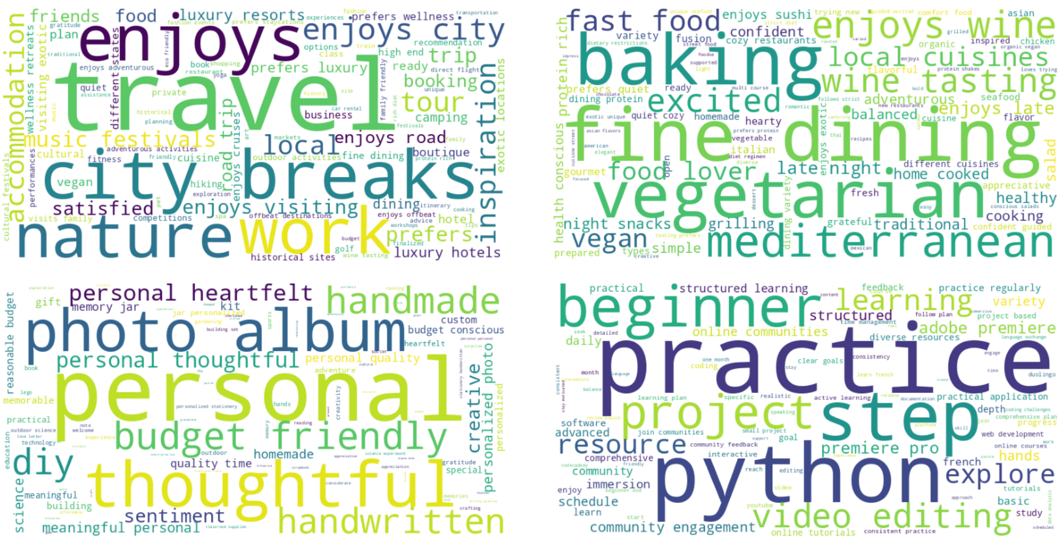

Figure 5: The distribution of user preferences in all 4 tasks.

Task	Pref. Alignment	Add'l. Pref.	RP. Compl.	Memo.	Detail Level	Practical Usef.	Div.
skills learning planning	1.233	0.144	1.564	0.941	**1.951**	**1.888**	1.841
travel planning	**1.837**	**0.436**	1.551	0.943	1.795	1.736	1.707
preparing gifts	1.228	0.019	**1.618**	**0.998**	1.920	1.867	**1.989**
recipe planning	1.670	0.389	1.616	0.977	1.830	1.795	0.653

Table 2: Average scores for each metric across 4 tasks. For all metrics, higher scores indicate better quality.

preparing gifts tasks, offering detailed and practical responses. The assistant's response diversity in the *recipe planning* task is relatively average, indicating room for improvement.

4.3 Dataset Annotation

We conducted additional annotations for both the simulator and the assistant's utterances, focusing on **user intent** and **hallucinations**.

User Intent Annotators were given options including *Search*, *Recommendation*, *Search & Recommendation*, *Planning*, and *Others* to select from. The following section will demonstrate the detailed distributions and usage of intent annotations.

Hallucinations Annotators were asked to identify fundamental common-sense errors in the assistant's responses. Eventually, annotators identified 39 responses with common-sense hallucinations out of a total of over 7,000 assistant replies. For each response with hallucinations, we also noted the reason for the hallucination to help future use.

In the released dataset, we included all these annotations at the utterance level.

Compared to the traditional conversational recommendation datasets, *RecLLMSim* is constructed in **multiple** more **realistic** and **complicated** scenarios.

5 *RecLLMSim* Application

This section will discuss the application of *RecLLMSim*. We conducted the intent detection experiments on our dataset and analyzed other potential usages of the dataset.

5.1 Experimental Results on Intent Detection

5.1.1 Dataset Preprocessing

For our experiments, each utterance from the simulator was treated as a single data sample, with the corresponding labeled intent serving as the target label. This process yielded a total of 7,096 data pieces. The dataset was divided into training, validation, and testing sets in an 8:1:1 ratio. Table 3 shows the basic characteristics of each split.

5.1.2 Models

We evaluated various types of text classification models in our experiments.

Traditional Machine Learning Models We tested the performance of several frequency-based statistical machine-learning methods on

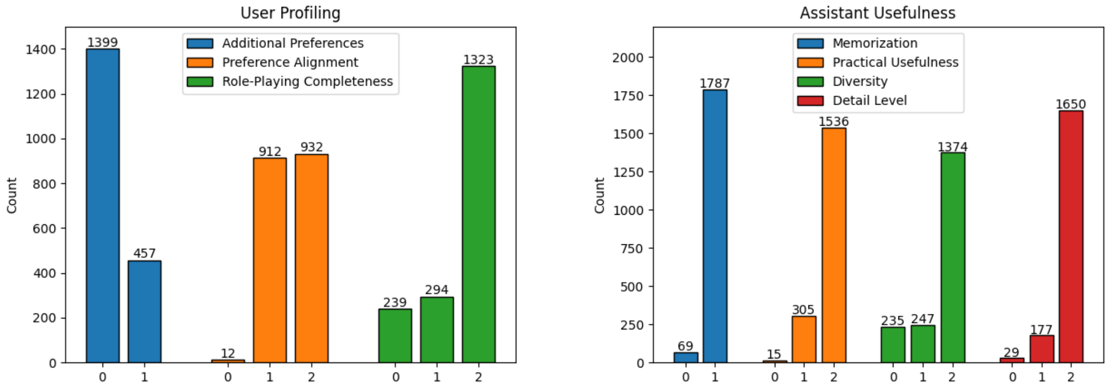

Figure 6: The score distribution across the seven metrics for the filtered dataset.

Split	#samples	Avg. #tokens	Search	Rec.	Search & Rec.	Planning	Others
Training	5676	104.27	307	1014	816	1930	1609
Validation	710	104.24	35	129	113	241	192
Testing	710	104.02	38	120	80	245	227

Table 3: Basic characteristics of each split.

the dataset, including SVM(Vapnik, 2013), decision tree(Quinlan, 1986), random forests(Breiman, 2001), AdaBoost(Freund and Schapire, 1997), and Bernoulli Naive Bayes(Bayes, 1763). We first vectorize the text using TF-IDF and then apply these models to the resulting vector representations.

Pre-trained Language Models (PLMs) We selected various PLMs, including BERT(Devlin et al., 2018), RoBERTa(Liu et al., 2019), T5(Raffel et al., 2020), GPT-2(Radford et al., 2019), to evaluate their performance. We manually designed prompt templates to test the performance of the language models before and after prompt-tuning.

Large Language Models (LLMs) We evaluate the performance of several LLMs, including ChatGLM(Zeng et al., 2022), GPT-4(Achiam et al., 2023), and Claude 3(Anthropic, 2024).

5.1.3 Implement Details

The specific hyperparameters for each PLM were adjusted to fit within the constraints of available GPU memory. To evaluate the intent detection and learning capabilities of different PLMs and LLMs, we considered three settings in our experiments:

- **Zero-shot**: We tested the zero-shot intent detection capability of the model directly based on the manual prompt template.

- **Few-shot In-conext**: We tested the in-context learning capability of the model with few-shot examples provided within the template.

- **Fine Tuning** We tested the model's performance after prompt tuning using the manual prompt template (PLMs only).

To evaluate the performance of each model, we employed both accuracy and macro-F1 as the metrics. These metrics provide a comprehensive view of model performance, enabling us to compare the effectiveness of different pre-trained language models on our dataset.

5.1.4 Results and Analysis

Table 4 shows the performance of three types of models in our experiments. The table is divided into three columns, each corresponding to one of the experimental settings for PLMs.

As shown in Table 4, PLMs, regardless of their scale and architecture, performed poorly under untrained conditions. In contrast, LLMs demonstrated significantly better performance in both settings. Specifically, GPT-4 and Claude 3 showed notable performance improvements when provided with few-shot examples, illustrating the in-context learning capabilities of LLMs.

Interestingly, traditional machine learning models outperformed even the best LLMs, suggesting

PLM	**Zero-shot**		**Few-shot In-context**		**Fine Tuning**	
	Accuracy	Macro-F1	Accuracy	Macro-F1	Accuracy	Macro-F1
SVM	/	/	/	/	0.7577	0.6274
DecisionTree	/	/	/	/	0.6901	0.5573
RandomForest	/	/	/	/	0.7479	0.5834
AdaBoost	/	/	/	/	0.7239	0.5858
BernoulliNB	/	/	/	/	0.6901	0.5748
BERT-base	0.1620	0.0638	0.1690	0.0664	0.7732	0.6340
BERT-large	0.1704	0.0673	0.1690	0.0578	0.7803	0.6535
RoBERTa-base	0.1761	0.0853	/	/	0.7887	**0.6723**
RoBERTa-large	0.1718	0.0702	/	/	0.7901	0.6694
GPT2-base	0.1690	0.0578	/	/	0.7972	0.6517
GPT2-medium	0.1113	0.0830	/	/	0.7930	0.6586
GPT2-large	0.1648	0.0609	/	/	0.7789	0.6421
T5-small	0.0930	0.0648	0.2944	0.1439	0.7803	0.6453
T5-base	0.0690	0.0523	0.1676	0.0620	0.7944	0.6706
T5-large	0.1718	0.0953	0.1507	0.1007	**0.7986**	0.6447
ChatGLM-6B	0.1352	0.0708	0.1282	0.0693	/	/
ChatGLM2-6B	0.1296	0.1025	0.0930	0.0828	/	/
ChatGLM3-6B	0.3521	0.1296	0.3662	0.1952	/	/
Claude-3-opus	0.5887	0.4243	0.6338	0.4780	/	/
GPT-4-turbo	0.5324	0.4113	**0.6648**	**0.5305**	/	/
GPT-4o	**0.6056**	**0.4249**	0.6183	0.4111	/	/

Table 4: Experimental results of intent detection task in RecLLMSim. We provide the result of traditional machine learning models in the last column of the table. In the **Few-shot In-context column**, a '/' symbol indicates that the corresponding PLM cannot support the template length that includes few-shot examples.

that the user simulator's simulation in our dataset is complicated. This finding demonstrates the effectiveness of using LLMs for user simulation. Moreover, prompt-tuned PLMs achieved the best results, indicating their advancements over traditional machine learning methods.

5.2 Other Potential Usage

The *RecLLMSim* dataset proposed in this work has other potential usages as follows:

- **Recommendation in different scenarios**: The conversations in this dataset, set across four different scenarios, can be used to study the processes and paradigms of making recommendations in various application contexts.
- **User profiling and simulation**: Instead of relying solely on the attributes of the items from user interactions to construct user profiles, our dataset offers a diverse and rich set of user profile descriptions. These profiles and the corresponding conversations can facilitate more user profiling and simulation research.
- **Hallucinations detection**: Our conversation data with manual labels of common-sense hallucinations can be directly used in researching hallucination detection for LLMs.

6 Conclusion and Future Work

To the best of our knowledge, this is the first work to generate task-specific conversational recommendation data across diverse scenarios entirely using LLMs as both user simulators and assistants. The dataset we introduced, *RecLLMSim*, is a valuable resource for various conversational and recommendation-related research.

The framework we proposed can be utilized to support research in user profiling and simulation, offering a flexible tool for creating dynamic conversational datasets. Our results demonstrate the potential of LLMs in automated conversation generation, paving the way for future advancements in conversational systems and recommender systems.

For future work, incorporating user studies where humans engage in conversations under the same conditions would provide a valuable benchmark against LLM-generated datasets. Additionally, interesting research topics include integrating external item lists to support the recommender system and improving the user profiling methodology.

Limitations

While our work leverages LLMs to generate conversational data, the choice of LLM significantly impacts the quality of the dialogues and user simulations. A comparative analysis involving different LLMs as user simulators and assistants would enhance the robustness and credibility of our conclusions. Future work should explore the performance variations across multiple LLMs to provide a more comprehensive evaluation.

Additionally, our proposed dataset, *RecLLMSim*, has not been compared against existing conversational datasets through either manual or automated methods. Conducting comparative analyses with other datasets would offer valuable insights into the relative strengths and weaknesses of *RecLLMSim*, further validating its utility and effectiveness in various research applications. Addressing these limitations in future studies would strengthen the findings and broaden the impact of our framework and dataset.

Ethical Considerations

In the user profile generation and conversation generation process, no personal privacy content is used to generate. We did not introduce any bias in the process of dataset generation. After we obtained the generated dataset, we translated all the utterances into Chinese to facilitate future usage.

Additionally, all crowdsourced annotators we hire are qualified. We offer an average effective per-conversation rate of $0.75, which varies based on the complexity of annotating each conversation.

This dataset could be used for research purposes to model user intent and build more intelligent conversational systems. Note that this *RecLLMSim* dataset uses an Apache License.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

AI Anthropic. 2024. The claude 3 model family: Opus, sonnet, haiku. *Claude-3 Model Card*.

Thomas Bayes. 1763. Lii. an essay towards solving a problem in the doctrine of chances. by the late rev. mr. bayes, frs communicated by mr. price, in a letter to john canton, amfr s. *Philosophical transactions of the Royal Society of London*, (53):370–418.

Pradipta Biswas, Peter Robinson, and Patrick Langdon. 2012. Designing inclusive interfaces through user modeling and simulation. *International Journal of Human-Computer Interaction*, 28(1):1–33.

Leo Breiman. 2001. Random forests. *Machine learning*, 45:5–32.

Chi-Min Chan, Weize Chen, Yusheng Su, Jianxuan Yu, Wei Xue, Shanghang Zhang, Jie Fu, and Zhiyuan Liu. 2023. Chateval: Towards better llm-based evaluators through multi-agent debate. *arXiv preprint arXiv:2308.07201*.

Meng Chen, Ruixue Liu, Lei Shen, Shaozu Yuan, Jingyan Zhou, Youzheng Wu, Xiaodong He, and Bowen Zhou. 2019. The jddc corpus: A large-scale multi-turn chinese dialogue dataset for e-commerce customer service. *arXiv preprint arXiv:1911.09969*.

Haixing Dai, Zhengliang Liu, Wenxiong Liao, Xiaoke Huang, Yihan Cao, Zihao Wu, Lin Zhao, Shaochen Xu, Wei Liu, Ninghao Liu, et al. 2023. Auggpt: Leveraging chatgpt for text data augmentation. *arXiv preprint arXiv:2302.13007*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Bosheng Ding, Chengwei Qin, Linlin Liu, Lidong Bing, Shafiq Joty, and Boyang Li. 2022. Is gpt-3 a good data annotator? *arXiv preprint arXiv:2212.10450*.

Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Zhi Zheng, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. 2023. Enhancing chat language models by scaling high-quality instructional conversations. *arXiv preprint arXiv:2305.14233*.

Yoav Freund and Robert E Schapire. 1997. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1):119–139.

Kallirroi Georgila, James Henderson, and Oliver Lemon. 2006. User simulation for spoken dialogue systems: learning and evaluation. In *Interspeech*, pages 1065–1068. Citeseer.

Vitor Jeronymo, Luiz Bonifacio, Hugo Abonizio, Marzieh Fadaee, Roberto Lotufo, Jakub Zavrel, and Rodrigo Nogueira. 2023. Inpars-v2: Large language models as efficient dataset generators for information retrieval. *arXiv preprint arXiv:2301.01820*.

Jean Kaddour, Joshua Harris, Maximilian Mozes, Herbie Bradley, Roberta Raileanu, and Robert McHardy. 2023. Challenges and applications of large language models. *arXiv preprint arXiv:2307.10169*.

Minjin Kim, Minju Kim, Hana Kim, Beong-woo Kwak, Soyeon Chun, Hyunseo Kim, SeongKu Kang, Youngjae Yu, Jinyoung Yeo, and Dongha Lee. 2024. Pearl: A review-driven persona-knowledge grounded conversational recommendation dataset. *arXiv preprint arXiv:2403.04460*.

Harrison Lee, Samrat Phatale, Hassan Mansoor, Kellie Lu, Thomas Mesnard, Colton Bishop, Victor Carbune, and Abhinav Rastogi. 2023. Rlaif: Scaling reinforcement learning from human feedback with ai feedback. *arXiv preprint arXiv:2309.00267*.

Raymond Li, Samira Ebrahimi Kahou, Hannes Schulz, Vincent Michalski, Laurent Charlin, and Chris Pal. 2018. Towards deep conversational recommendations. In *Advances in Neural Information Processing Systems 31 (NIPS 2018)*.

Ruibo Liu, Jerry Wei, Fangyu Liu, Chenglei Si, Yanzhe Zhang, Jinmeng Rao, Steven Zheng, Daiyi Peng, Diyi Yang, Denny Zhou, et al. 2024. Best practices and lessons learned on synthetic data for language models. *arXiv preprint arXiv:2404.07503*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. 2023. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, pages 1–22.

J. Ross Quinlan. 1986. Induction of decision trees. *Machine learning*, 1:81–106.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

Vladimir Vapnik. 2013. *The nature of statistical learning theory*. Springer science & business media.

Lei Wang, Jingsen Zhang, Xu Chen, Yankai Lin, Ruihua Song, Wayne Xin Zhao, and Ji-Rong Wen. 2023a. Recagent: A novel simulation paradigm for recommender systems. *arXiv preprint arXiv:2306.02552*.

Shuohang Wang, Yang Liu, Yichong Xu, Chenguang Zhu, and Michael Zeng. 2021. Want to reduce labeling cost? gpt-3 can help. *arXiv preprint arXiv:2108.13487*.

Xiaolei Wang, Xinyu Tang, Wayne Xin Zhao, Jingyuan Wang, and Ji-Rong Wen. 2023b. Rethinking the evaluation for conversational recommendation in the era of large language models. *arXiv preprint arXiv:2305.13112*.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2022. Self-instruct: Aligning language models with self-generated instructions. *arXiv preprint arXiv:2212.10560*.

Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, and Daxin Jiang. 2023a. Wizardlm: Empowering large language models to follow complex instructions. *arXiv preprint arXiv:2304.12244*.

Canwen Xu, Daya Guo, Nan Duan, and Julian McAuley. 2023b. Baize: An open-source chat model with parameter-efficient tuning on self-chat data. *arXiv preprint arXiv:2304.01196*.

Jiacheng Ye, Jiahui Gao, Qintong Li, Hang Xu, Jiangtao Feng, Zhiyong Wu, Tao Yu, and Lingpeng Kong. 2022. Zerogen: Efficient zero-shot learning via dataset generation. *arXiv preprint arXiv:2202.07922*.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Sainbayar Sukhbaatar, Jing Xu, and Jason Weston. 2024. Self-rewarding language models. *arXiv preprint arXiv:2401.10020*.

Aohan Zeng, Xiao Liu, Zhengxiao Du, Zihan Wang, Hanyu Lai, Ming Ding, Zhuoyi Yang, Yifan Xu, Wendi Zheng, Xiao Xia, et al. 2022. Glm-130b: An open bilingual pre-trained model. *arXiv preprint arXiv:2210.02414*.

Junjie Zhang, Yupeng Hou, Ruobing Xie, Wenqi Sun, Julian McAuley, Wayne Xin Zhao, Leyu Lin, and Ji-Rong Wen. 2024. Agentcf: Collaborative learning with autonomous language agents for recommender systems. In *Proceedings of the ACM on Web Conference 2024*, pages 3679–3689.

Yongfeng Zhang, Xu Chen, Qingyao Ai, Liu Yang, and W Bruce Croft. 2018a. Towards conversational search and recommendation: System ask, user respond. In *Proceedings of the 27th acm international conference on information and knowledge management*, pages 177–186.

Zhuosheng Zhang, Jiangtong Li, Pengfei Zhu, Hai Zhao, and Gongshen Liu. 2018b. Modeling multi-turn conversation with deep utterance aggregation. *arXiv preprint arXiv:1806.09102*.

Nan Zhao, Haoran Li, Youzheng Wu, and Xiaodong He. 2022. Jddc 2.1: A multimodal chinese dialogue dataset with joint tasks of query rewriting, response generation, discourse parsing, and summarization. In *Proceedings of the 2022 conference on empirical methods in natural language processing*, pages 12037–12051.

Nan Zhao, Haoran Li, Youzheng Wu, Xiaodong He, and Bowen Zhou. 2021. The jddc 2.0 corpus: A large-scale multimodal multi-turn chinese dialogue dataset for e-commerce customer service. *arXiv preprint arXiv:2109.12913*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Tianle Li, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zhuohan Li, Zi Lin, Eric Xing, et al. 2023. Lmsys-chat-1m: A large-scale real-world llm conversation dataset. *arXiv preprint arXiv:2309.11998*.

Hao Zhou, Chujie Zheng, Kaili Huang, Minlie Huang, and Xiaoyan Zhu. 2020. Kdconv: A chinese multi-domain dialogue dataset towards multi-turn knowledge-driven conversation. *arXiv preprint arXiv:2004.04100*.

Lixi Zhu, Xiaowen Huang, and Jitao Sang. 2024. How reliable is your simulator? analysis on the limitations of current llm-based user simulators for conversational recommendation. In *Companion Proceedings of the ACM on Web Conference 2024*, pages 1726–1732.