Augmentation with Neighboring Information for Conversational Recommendation

YUANXING LIU, Harbin Institute of Technology, Harbin, China
JIAHUAN PEI, Vrije Universiteit Amsterdam, Amsterdam, Netherlands
WEI-NAN ZHANG, Harbin Institute of Technology, Harbin, China
MING LI, University of Amsterdam, Amsterdam, Netherlands
WANXIANG CHE, Harbin Institute of Technology, Harbin, China
MAARTEN DE RIJKE, University of Amsterdam, Amsterdam, Netherlands

Conversational recommender systems (CRSs) suggest items to users by understanding their needs and preferences from natural language conversations. While users can freely express preferences, modeling needs and preferences solely from users' conversations is challenging due to the sparsity of the available information. Prior work introduces external resources to enrich information expressed in conversations. Obtaining such resources is challenging and not always effective. Can learning intrinsic relations among conversations and items enhance information without the use of external resources? Inspired by collaborative filtering, we propose to use so-called neighboring relations within training data, i.e., relations between conversations, items, and similar conversations and items, to enhance our algorithmic understanding of CRSs.

We propose a neighboring relations enhanced conversational recommender system (NR-CRS) and study how neighboring relations improve CRSs from two angles: (i) We mine preference information from neighboring conversations to enhance the modeling of user representations and learning of user preferences. (ii) We generate negative samples based on neighboring items to extend the data available for training CRSs. Experiments on the ReDial dataset show that neighboring relations enhanced conversational recommender system (NR-CRS) outperforms the state-of-the-art baseline by 11.3–20.6% regarding recommendation performance while generating informative and diverse responses. We also assess the capabilities of large language models (i.e., Llama 2, Llama 3, and Chinese-Alpaca2) for CRSs. While the generated responses exhibit enhanced fluency and informativeness, recommending target items with LLMs remains challenging; we recommend that LLMs be used as a decoding base for NR-CRS to generate relevant and informative responses.

CCS Concepts: • **Information systems → Recommender systems**; • **Computing methodologies → Discourse, dialogue and pragmatics**;

Additional Key Words and Phrases: Conversational recommendation, Neighboring relations

This work is partially supported by the National Key Research and Development Program (No. 2022YFF0902100), the National Natural Science Foundation of China (No. 92470205), the Nature Scientific Foundation of Heilongjiang Province (YQ2021F006), the China Scholarship Council, the Dutch Research Council (NWO), under project numbers 024.004.022, NWA.1389.20.183, and KICH3.LTP.20.006, and the European Union's Horizon Europe program under grant agreement No. 101070212 (FINDHR).

Authors' Contact Information: Yuanxing Liu, Harbin Institute of Technology, Harbin, China; e-mail: yxliu@ir.hit.edu.cn; Jiahuan Pei, Vrije Universiteit Amsterdam, Amsterdam, Netherlands; e-mail: j.pei2@vu.nl; Wei-Nan Zhang (corresponding author), Harbin Institute of Technology, Harbin, China; e-mail: wnzhang@ir.hit.edu.cn; Ming Li, University of Amsterdam, Amsterdam, Netherlands; e-mail: m.li@uva.nl; Wanxiang Che, Harbin Institute of Technology, Harbin, China; e-mail: car@ir.hit.edu.cn; Maarten de Rijke, University of Amsterdam, Amsterdam, Netherlands; e-mail: m.derijke@uva.nl.

ACM 1558-2868/2025/2-ART62
https://doi.org/10.1145/3712588

ACM Reference format:
Yuanxing Liu, Jiahuan Pei, Wei-Nan Zhang, Ming Li, Wanxiang Che, and Maarten de Rijke. 2025. Augmentation with Neighboring Information for Conversational Recommendation. *ACM Trans. Inf. Syst.* 43, 3, Article 62 (February 2025), 49 pages.
https://doi.org/10.1145/3712588

1 Introduction

Conversational recommender systems (CRSs) have attracted much attention due to their success in information seeking [61, 68, 85]. CRSs are able to explore users' information needs and preferences through multi-turn conversations [34]. Unlike typical recommender systems, CRSs can communicate with users and attempt to capture their information needs through conversations [23]. Moreover, recommendation results are presented in natural language (e.g., "You might like the movie 'Iron Man'."), which may be a more suitable way of presenting results than as an item or a list of items in some contexts [26, 76, 86, 103, 106, 116].

One challenge in CRSs is how to model users' information needs and preferences as expressed in natural language [23]. Without explicit user behavior such as clicks and purchases [29], most CRSs extract items and entities that are mentioned in conversations to represent users' information needs and preferences [54, 121]. However, items and entities in a single conversation of a given user are usually extremely sparse [24]. For example, only about 0.17% of the tokens in a conversation are items or entities in the well-known ReDial dataset [51]. A common solution is to extend the set of entities mentioned with entities from external resources, e.g., DBpedia [5, 107], ConceptNet [116], product reviews [59], and a sentiment vocabulary [54]. But this is limited by the scale, quality, and timeliness of the external resources.

Another challenge in CRSs is the lack of sufficient training data [51]. It is labor-intensive and time-consuming to collect large-scale and high-quality conversational recommendation data [46, 51, 117]. Intuitively, **data augmentation (DA)** is a widely used technique to increase the amount of labeled training data. Some prior work introduces auxiliary tasks that are jointly learned with a CRS to capture the data representation and intrinsic correlations [48, 49, 114]. BARCOR [91] uses the properties of an item as the conversation context for recommending the item but it only use external resources as a source of additional information (e.g., topics or goals) for input or supervision signals, neglecting augmenting the training data.

We aim to tackle the above two challenges by augmentation in both user modeling and negative training samples using neighboring conversations and items, as illustrated in Figure 1. We represent conversations (items) with embeddings and seek neighboring conversations (items) that are semantically similar to a given conversation (item) in the embedding space. Intuitively, (i) neighboring conversations help to clarify users' information needs and preferences as expressed in a single conversation for better user modeling; and (ii) neighboring items can be considered as pseudo-labels for constructing negative samples to provide additional constraints for better model training.

For ease of understanding, we illustrate our motivation with an example.[1] Imagine a user who is looking for a superhero movie. The user tells a recommender that he or she likes "Iron Man." The recommender suggests "The Avengers," which features many Marvel superheroes, including "Iron Man." To enhance the ability of CRSs to learn the relationship between the ongoing conversation and the recommended movie "The Avengers," we implement two strategies. First, we propose that the

[1]Movies such as "Iron Man," "The Avengers," are examples of items in the movie domain. The scope of items in our approach is not limited to movies.

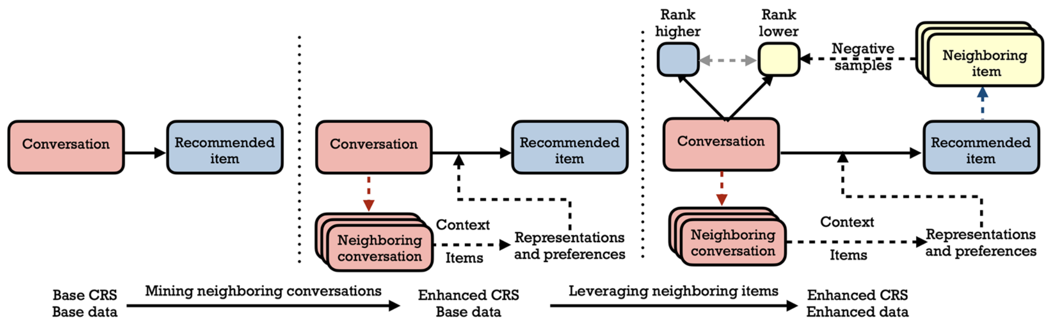


Fig. 1. Overall structure of using neighboring relations for CRS. The dashed red (blue) arrows refer to finding neighboring conversations (items). The bi-directional gray dashed arrow indicates an increase in the distance between the two items in the embedding space. The interactive form of "conversation" consists of multiple turns of natural language text interaction.

recommender identify neighboring conversations based on the context of the ongoing conversation to gather additional information. This can include finding another conversation discussing a Marvel movie, and the movies discussed in that conversation can be used to enrich the user's representation of the ongoing conversation. Second, we propose that the recommender searches for other movies related to "The Avengers" to impose additional constraints. For example, the recommender may find "Avengers: Age of Ultron," which is a sequel to "The Avengers." We propose that the recommender treats "Avengers: Age of Ultron" as a strong negative example. We aim for CRSs to predict "The Avengers" with a higher probability than that of "Avengers: Age of Ultron." Simultaneously, we aim to maximize the distance between "The Avengers" and "Avengers: Age of Ultron" in the embedding space. This ensures that "Avengers: Age of Ultron" is less likely to be selected in the next round of neighboring movie selection. By incorporating neighboring conversations and related items, we not only enhance the input information but also impose additional output constraints. These two aspects enable CRSs to better learn the relationship between conversations and the corresponding recommended items.

In this Article, we propose a **neighboring relations enhanced conversational recommender system (NR-CRS)** that uses neighboring relations (of conversations and items) to improve CRSs in two ways:

To optimize user modeling. NR-CRS models the information needs and preferences not only from a single conversation but also from its neighboring conversations: (i) NR-CRS obtains an enhanced user representation by incorporating correlation features mined from neighboring conversations and their recommended items; and (ii) NR-CRS obtains enhanced user preferences by using entities that appear in an ongoing conversation and its neighboring conversations.

To enhance model training. NR-CRS extends the available training data and improves the training process with neighboring items: (i) for each training sample, NR-CRS identifies neighboring items and then extends the training dataset by using neighboring items as negative samples; and (ii) in the training process, we use the extended data to initialize NR-CRS; after one epoch, we train NR-CRS on the standard data.

We conduct experiments on the ReDial and U-NEED datasets. Learning neighboring relations of conversations and items can significantly improve the performance of conversational recommendation. We also find that: (i) improvements in user modeling based on neighboring conversations can significantly improve the recommendation performance on all metrics; and (ii) additional constraints provided by negative samples constructed based on neighboring items can further improve the recommendation performance of CRSs.

Our main contributions are as follows:

- We propose a NR-CRS. It learns neighboring relations of conversations and items to enhance user modeling and augment training data.
- We perform extensive experiments on the ReDial and U-NEED datasets to show that learning neighboring relations can improve the performance of CRSs.
- We assess the capabilities of large language models (LLMs) (i.e., Llama 2, Llama 3, and Chinese-Alpaca2) for CRSs. We find that providing accurate recommendations is challenging due to input constraints and spurious responses of LLMs, although they can be used as a decoding base for NR-CRS to generate relevant and informative responses.

2 Related Work

2.1 CRSs

CRSs can be categorized into recommendation-driven and conversation-driven CRSs [34].

Recommendation-Driven CRSs. Recommendation-driven CRSs elicit users' information needs and preferences by asking them appropriate questions [71, 82, 111, 120]. The questions are usually selected or generated based on pre-defined templates and goal-related semantic constraints (e.g., intents, slots, attributes, and actions) [27]. Christakopoulou et al. [7] propose a preference elicitation framework to interactively collect users' feedback over identified absolute and relative questions. Zhang et al. [111] predict aspects of products to generate a sequence of questions for understanding the user needs in search and recommendation. Sun and Zhang [82] adopt a belief tracker to generate and update questions for understanding users' long-term preferences. Kang et al. [37] treat a CRS as a goal-oriented communication game between two agents and train them in supervised and self-supervised learning paradigm. Zou et al. [120] ask users to express their preferences over descriptive item features that are extracted from relevant documents. Li et al. [47] consider CRSs for cold-start users. They unify attributes and items in the same arm space to achieve their EE tradeoffs with Thompson Sampling. Ni et al. [62] consider the connection between the previous rounds and the current round of the conversation and propose a more dynamic and personalized graph structure to better use a user's historical preferences. In addition to what questions to ask, prior work also learns when to recommend items [41] and how to extend questions by seeking attributes [22, 42, 87, 96, 102]. Recently, Lin et al. [53] study the multi-ground-truth multi-round conversational recommendation scenario for conversational recommendation and enhance user personalized experiences by designing unique features. Chu et al. [8] design three components to learn a meta policy and adapt it to new users. Wang et al. [95] present a CRS that uses representation fusion to generate questions in two distinct subsessions: Local and Global Question Search, without relying on dialogue policy agents.

The above models assume users can reveal their information needs and preferences explicitly in their answers. As not all users are willing to expose their personal information, this is unrealistic. Moreover, revealment may have to concern large volumes of attributes to be effective [67], which reduces its feasibility.

Conversation-Driven CRSs. Conversation-driven CRSs infer users' information needs and preferences in conversations for accurate recommendation [46]. The key challenge is how to represent conversations and generate responses that contain recommendation-oriented information, e.g., entities related to items. The principle is to encode conversations with different sequence models (e.g., RNNs [52], HERD [46], and transformers [37, 117, 121]), and then decode responses token by token with different mechanisms [5, 46, 52]. To enrich recommendation-oriented information, prior work introduces external knowledge (e.g., ConceptNet, DBpedia) [5, 52, 60, 110, 113] and integrates multiple sources of knowledge by semantic fusion [98, 100, 116, 118]. Recent work has also explored multiple topics [52, 70, 72, 101, 117] and different learning paradigms, e.g., pre-training fine-tuning [94, 96, 117], reinforcement learning [12, 31, 82, 87], adversarial learning [71], and contextual bandit learning [109]. More recently, Shen et al. [75] study a language model-driven CRS to investigate how unintended bias manifests itself in substantially shifted price and category distributions of restaurant recommendations. Deng et al. [13] propose a unified multi-goal CRS that unifies four tasks, i.e., goal planning, topic prediction, item recommendation, and response generation, with different formulations into the same sequence-to-sequence paradigm. In addition, there has been some recent analytical work on CRS. Rana et al. [69] presents a user study indicating that personalized critiquing-based recommendation systems are more efficient and satisfying for users compared to non-personalized systems. Siro et al. [77] investigate the effect of six dialogue aspects on user satisfaction when interacting with a CRS. Suchmann et al. [80] explore a GUI design for CRSs that supports topic switching and relationships to aid user navigation and decision-making.

We acknowledge the existence of knowledge graph-based CRSs, such as UNICORN [12], which implicitly incorporate "neighboring relations" to enhance the user representation through graph representation learning and knowledge graphs. The key difference between knowledge graph-based CRSs and our proposed NR-CRS lies in how neighboring relations are defined and applied. Knowledge graph-based CRSs rely on direct node connections to enrich node representations, whereas our approach defines neighboring relations based on proximity in high-dimensional space. This allows us to use similar representations to enhance user modeling and augment training data with neighboring conversations and items, resulting in different methods of retrieving and leveraging neighboring information.

Unlike prior approaches to conversation-driven CRS, the proposed NR-CRS introduces neighboring conversations and items, and explores their correlations to enhance CRS. Neighboring conversations have proved to be useful for user representation [50]; however, prior work does do not model semantically similar items that can directly influence user preferences over candidate items.

2.2 DA for CRSs

DA has been used in a variety of areas and has proven to be very effective. DA increases the quantity and diversity of training examples without explicitly collecting new data [18]. In natural language processing, DA is targeted mainly at how to generate new text, such as natural perturbation on dialogues [99]. By slight modifications based on the original semantics, training samples are augmented with paraphrasing, noising, and sampling [43]. Several recent publications augment data for training by pseudo supervision signal elicitation [11] and multiple agent collaboration [66]. In **information retrieval (IR)**, with **sequential recommendation (SR)** as an example, Bian et al. [1] leverage related information from similar users for generating both relevant and diverse augmentation to improve SR. Song and Suh [78] propose four strategies to transform original item sequences with direct manipulation. Wang et al. [93] present CFCRS, a CounterFactual data simulation approach for CRSs, designed to alleviate the issue of scarce training data by augmenting datasets with realistic, coherent dialogues generated from a conversation flow language model, leading to performance improvements especially in data-limited scenarios.

Unlike the work listed above, we focus on DA in CRSs. To the best of our knowledge, most of existing work mainly introduces external knowledge to augment the understanding of conversation to model better user representations. We construct pseudo-labels and augment the training data based on the neighboring relations of items.

2.3 LLMs for CRSs

LLMs have gained significant attention for their proficiency in natural language comprehension and generation [89, 112]. In addition to LLMs that have been released as "closed source," such as GPT-3 [3], InstructGPT [64], and PaLM [6], an increasing number of LLMs are being released as open source, such as Bloom [73], GLM [14], and LLaMA [83]. LLaMA is a widely noted open source LLM that has been adapted to various domain-specific applications [112], such as Goat [56] (mathematics), Cornucopia [105] (finance), BenTsao [90] (medicine), Lawyer [32] (law), BELLE [35] (bilingualism), and TaoLi [104] (education). Recent studies have demonstrated LLMs' effectiveness in recommender systems [17] and IR [119]. For example, ChatGPT [63] has been examined for its performance on passage re-ranking [81] and recommendation [55]. For CRS, most related work explores the potential and performance of LLMs to provide conversational recommendations. He et al. [30] investigate the performance of LLMs on conversational recommendation tasks in a zero-shot setting from the perspective of repetition and exploration [44, 45]. Gao et al. [25] enable LLMs to interact with users to understand their preferences as well as provide recommendations. The recommended items come from recommender systems or IR [33]. Friedman et al. [21] propose a road map for using a LLM to build a controllable and explainable CRS for YouTube videos. Feng et al. [19] use a LLM to manage sub-tasks in CRSs and generate improved responses. Wang et al. [92] use LLMs as user simulators to evaluate the overall performance of CRSs.

In this work, we assess the capabilities of the open-source LLMs Llama 2, Llama 3, and Chinese-Alpaca2 for CRSs on two datasets. In addition, we examine the use of a LLM as a decoding base to enhance response generation performance of the proposed NR-CRS.

3 Definitions

Table A1 (in Appendix A) lists the notation that we use in the article. In this article, a conversational context X consists of a sequence of words in chronological order from historical utterances. A CRS $\mathcal{M}$ parameterized by θ aims to (i) recommend an item y from the candidate item set Y; and (ii) generate a related response $Z = \{w_j\}_{j=1}^{|Z|}$ word by word from the vocabulary set V, i.e., $y, Z \leftarrow \mathcal{M}_\theta(X)$. The training set is $D = \{(x_i, y_i, z_i)\}_{i=1}^{|D|}$. Following the mainstream implementation of CRSs [116, 118], we abstract the sub-tasks as follows.

For item recommendation, we define the probability of recommending an item y to a user and apply a cross-entropy loss $\mathcal{L}_r$ as the optimization objective:

$$p_i^r = P(y \mid X) = \textsc{Softmax}(\boldsymbol{x}^\top \boldsymbol{y} + b), \tag{1}$$

$$\mathcal{L}_r = -\sum_{i=1}^{|Y|} y_i \log p_i^r, \tag{2}$$

where the vector $\boldsymbol{x} \in \mathbb{R}^d$ is the user representation that encodes discriminative features from X for user modeling; d is the dimensionality. In this article, we obtain $\boldsymbol{x}$ from the hybrid user ENCODER [118] given X. The hybrid user ENCODER comprises (i) a transformer-based module for encoding conversation history, (ii) an R-GCN-based module for encoding the knowledge graph, and (iii) a transformer-based module for encoding reviews. The vector $\boldsymbol{y} \in \mathbb{R}^d$ is the embedding of the candidate item y. The scalar b is a variable that indicates the user preference over the candidate item y.

For response generation, we define the probability of generating the jth word w_j in the response Z and minimize the negative log-likelihood loss $\mathcal{L}_g$ as

$$p_j^g = P(w_j | w_1, \ldots, w_{j-1}, X, y) \tag{3}$$

$$= \textsc{Decoder}([\boldsymbol{w}_1; \ldots; \boldsymbol{w}_j; \boldsymbol{x}; \boldsymbol{y}]) \tag{4}$$

$$\mathcal{L}_g = -\frac{1}{|Z|} \sum_{j=1}^{|Z|} \log \alpha_{w_j} p_j^g, \tag{5}$$

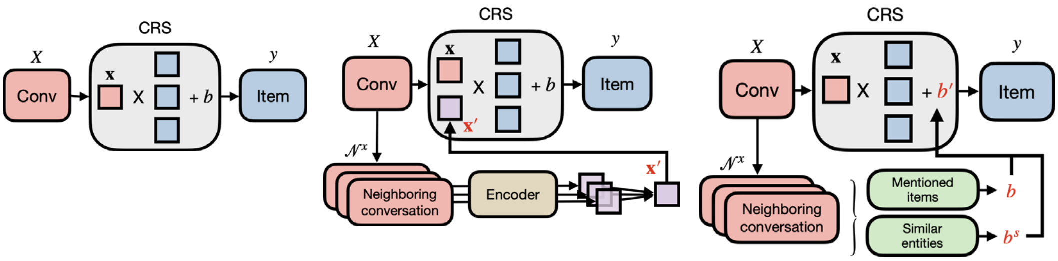
where α_{w_j} is the weight considering the frequency of the token w_j; $[\cdot;\cdot]$ refers to a concatenation operation; and p_j^g is approximated from the probability in the jth decoding step of the transformer-based DECODER [118] given the previous j generated words followed by a conversation context.

The type of information in conversations varies across recommendation scenarios. For instance, in the clothing recommendation scenario of the Fashion IQ dataset [97], conversations typically focus on attributes such as color and length, with users expressing specific preferences for these details. In contrast, our work concentrates on scenarios where entities play a central role in the conversation. For example, in the movie recommendation scenario of the ReDial dataset [46], discussions often center around specific entities. Entities can include concepts (e.g., "Superhero"), movies (e.g., "The Avengers"), actors (e.g., "Robert John Downey Jr."), and more. These entities are extracted based on an external knowledge base.

We consider three types of relations between conversations and items: (i) *Conversation–Conversation.* Given any two conversations X_i and X_j, we use the cosine similarity between their representations $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ to measure their neighboring relation. If the score is higher than a threshold λ_x, they are *neighboring conversations* of each other. (ii) *Item–Item.* Given any two items y_i and y_j, we use the cosine similarity between their embeddings $\boldsymbol{y}_i$ and $\boldsymbol{y}_j$ to measure their neighboring relation. If the score is higher than a threshold λ_y, they are *neighboring items* of each other. Setting λ_x and λ_y is a matter of considering many factors. Taking λ_x as an example, setting λ_x may need to take into account the characteristics of the conversations contained in the dataset, as well as the representation of the conversations obtained by encoding. In practice, we do not set a fixed λ_x to select neighboring conversations. This is because this may leave the ongoing conversation with too many or too few neighboring dialogues. Too many conversations would increase the computational costs, and too few conversations may not provide enough information. Therefore, in practice we choose a number of neighboring conversations with the highest cosine similarity to the representation. (iii) *Conversation–Item.* Given a conversation X and an item y, we use the dot product to measure the matching score of their representations $\boldsymbol{x}$ and $\boldsymbol{y}$. This score is the major factor to determine if a CRS should recommend the item y to the user (see Equation (1)).

4 Method

Intuitively, neighboring relations of conversations can provide additional information for an ongoing conversation during inference. Besides, neighboring relations of items can provide additional constraints during the training process. In this section, we introduce our proposal for improving a CRS with neighboring relations. First, we seek neighboring conversations, extract correlation features to enhance the user representation, and compute user preferences over entities. Then, we assemble the augmentation with neighboring relations. Second, we seek neighboring items, synthesize data by pseudo-labeling, and introduce an algorithm to train the model with augmented data.



(a) Vanilla CRS. (b) CRS with enhanced user representation. (c) CRS with enhanced user preference.

Fig. 2. Enhancing user models with neighboring conversations. In all subfigures, "Conv" is short for "conversation" and × means pairwise dot product between vectors. The red and blue squares indicate the representation of the ongoing conversation and the item embedding, respectively. In subfigure (b), the purple squares indicate the representations of neighboring conversations, while $\boldsymbol{x}'$ is the hybrid representation of all neighboring conversations. By using neighboring conversations, CRS improves not only the user representations but also the user preferences over candidate items.

4.1 User Modeling Augmentation with Neighboring Conversations

4.1.1 Seeking Neighboring Conversations. For each conversation, we obtain the user representation $\boldsymbol{x}$ following the hybrid encoder proposed in [118]. Then, we build an index $\boldsymbol{\mathcal{X}}$ for all user representations in the training set, query the user representation of the ongoing conversation, and collect a set of user representations of neighboring conversations $\mathcal{N}^{\boldsymbol{x}}$, i.e.,

$$\mathcal{N}^{\boldsymbol{x}} = \phi(\boldsymbol{x}, \boldsymbol{\mathcal{X}}, m), \tag{6}$$

where ϕ is a function to seek m neighboring conversations that have the highest cosine similarity score with the ongoing conversation. Next, we use neighboring conversations $\mathcal{N}^{\boldsymbol{x}}$ to enhance user modeling along two dimensions: user representations (Section 4.1.2) and user preferences (Section 4.1.3), i.e., $\boldsymbol{x}$ and b in Equation (1), as indicated in Figure 2.

4.1.2 Enhancing User Representations. We use a common feature fusion method, weighted summing, to exploit the retrieved neighboring conversations. Weighted summing allows for larger values for each dimension, allowing CRSs to better distinguish differences between dimensions. We obtain an auxiliary user representation $\boldsymbol{x}'$ as a weighted sum over the correlation features of all neighboring conversations:

$$\boldsymbol{x}' = \sum_{i=1}^{|\mathcal{N}^{\boldsymbol{x}}|} w_i \boldsymbol{v}_i \in \mathbb{R}^d, \tag{7}$$

which is expected to enrich the user representation. For the ith neighboring conversation, the correlation feature $\boldsymbol{v}_i \in \mathbb{R}^d$ can be learned by

$$\boldsymbol{v}_i = \mathrm{MLP}(\boldsymbol{x}_i, \boldsymbol{y}_i) \in \mathbb{R}^d, \tag{8}$$

where MLP denotes a multi-layer perceptron, $\boldsymbol{x}_i \in \mathcal{N}^{\boldsymbol{x}}$ is the user representation, and $\boldsymbol{y}_i$ is the representation of its recommended item. The learnable weight $w_i \in [0, 1]$ of the corresponding correlation feature can be denoted as

$$w_i = \textsc{Sigmoid}(\textsc{Linear}([\boldsymbol{x}; \boldsymbol{x}_i])) \in \mathbb{R}^1, \tag{9}$$

where Linear refers to a linear layer, and Sigmoid is the activation function.

4.1.3 Enhancing User Preferences. We hypothesize that entities mined from neighboring conversations may also be appropriate for the ongoing conversation. Consequently, we consider combining the original user preferences with user preferences based on neighboring conversations. We obtain an auxiliary user preference b' as a sum of the item-based preference b and the entity-based preference b^s:

$$b' = b + b^s \in \mathbb{R}^1. \tag{10}$$

Because user preference values are positively correlated with recommendation probability, we simply add the two user preferences without feature fusion. For an item that is mentioned in the ongoing conversation, we define the item-based preference b as

$$b = \sum_{i=1}^{1+|\mathcal{N}^{x}|} w_i \tau_i \in \mathbb{R}^1, \tag{11}$$

where $\tau_i \in \mathbb{R}^1$ is the frequency with which the item is mentioned in the ith conversation, and w_i is the corresponding weight. The weight of the ongoing conversation is 1, i.e., $w_1 = 1$ and weights of neighboring conversations are computed as in Equation (9).

If an item y is not mentioned, but its similar entities E are mentioned in the ongoing conversation, we define the entity-based preference b^s as

$$b^s = \sum_{i=1}^{1+|\mathcal{N}^{x}|} \sum_{j=1}^{|E|} w_i T_{i,j} s_j \in \mathbb{R}^1, \tag{12}$$

where $T_{i,j} \in \mathbb{R}^1$ is the frequency with which the jth entity $e_j \in E$ is mentioned in the ith conversation, and w_i is the same as in Equation (11); $s_j = \cos(\boldsymbol{e}_j, \boldsymbol{y}) \in [-1, 1]$ is the cosine similarity score of entity e_j and item y. To avoid repeatedly considering item y and introducing noise, the score of s_j will be set to 0 when entity e_j and item y refer to the same movie or the cosine similarity score is smaller than a threshold λ_p.

4.1.4 Optimizing User Modeling. We propose a neighboring relations-enhanced CRS, namely NR-CRS. NR-CRS instantiates a simple way to benefit from neighboring information for item recommendation and response generation.

With enhanced user representation $\boldsymbol{x}'$ and user preference b', the probability of recommending an item y to a user in Equation (1) is revised as

$$p_i^r = \textsc{Softmax}((\boldsymbol{x} + \boldsymbol{x}')^\top \boldsymbol{y} + b'). \tag{13}$$

The probability of generating the jth word w_j in the response Z in Equation (4) can be revised as

$$p_j^g = \textsc{Decoder}([\boldsymbol{w}_1; \ldots; \boldsymbol{w}_j; \boldsymbol{x}; \boldsymbol{y}^e]). \tag{14}$$

We compute a preference b^e for each entity in the same way as in Equation (10). Then we select the 10 highest-scoring entities via b^e. The representation $\boldsymbol{y}^e$ of these 10 entities is added to the decoding process.

4.2 DA with Neighboring Items

4.2.1 Neighboring Items as Hard Negative Samples for Pseudo-Labeled Data Construction. Neighboring items refer to items that are close to the target item in high-dimensional vector space. In this article, we identify neighboring items using cosine similarity scores. For each item y, we obtain

Algorithm 1: Training NR-CRS for Item Recommendation

input : Dataset w/ ground truth label $D = \{(x_i, y_i, z_i)\}_{i=1}^{|D|}$, warm-up epoch ξ, model $\mathcal{M}_\theta$, item set Y.
output: Updated model $\mathcal{M}_{\theta'}$.

1 **for** *each batch in epoch* **do**
2   **for** *each sample* (x, y, z) *in batch* **do**
3         $\boldsymbol{x} \leftarrow f_u(x)$; // Calculate user representation of the ongoing conversation.
4         $\mathcal{N}^{\boldsymbol{x}} \leftarrow \phi(\boldsymbol{x}, \boldsymbol{\mathcal{X}}, m)$; // Search m neighboring conversations
5         $\boldsymbol{x}' \leftarrow f_u([\boldsymbol{x}; \mathcal{N}^{\boldsymbol{x}}])$; // Update user representation (Equations (7)–(9))
6         $b' \leftarrow f_p(x, \mathcal{N}^{\boldsymbol{x}})$; // Update user preference (Equations (10)–(12))
7         $\mathcal{L}_r \leftarrow f_r(\boldsymbol{x}', \boldsymbol{y}, b')$; // Calculate recommendation loss (Equations (1) and (2))
8   **if** *epoch* $< \xi$ **then**
9             $\mathcal{N}^{y} \leftarrow \psi(\boldsymbol{y}, \boldsymbol{\mathcal{Y}}, n)$; // Search n neighboring items
10             $\mathcal{L}_\alpha \leftarrow f_\alpha(\boldsymbol{x}, \boldsymbol{x}', b', \boldsymbol{y}, \mathcal{N}^{y})$; // Calculate ranking loss (Equations (17)–(16))
11             $\mathcal{L}_\beta \leftarrow f_\beta(\boldsymbol{y}, \mathcal{N}^{y})$; // Calculate embedding loss (Equation (19))
12             $\mathcal{L}_r \leftarrow \mathcal{L}_r + \lambda_\alpha \mathcal{L}_\alpha + \lambda_\beta \mathcal{L}_\beta$; // Update recommendation loss (Equation (20))
13   **end**
14   Minimize loss $\mathcal{L}_r$ and update parameters $\theta \leftarrow \theta'$;
15 **end**
16 Back to line 1 to iterate lines 1–15 until the maximum epoch.

its representation $\boldsymbol{y}$ from an entity encoder following [116, 118]. Then, we build an index $\boldsymbol{\mathcal{Y}}$ for all candidate items, query the item, and collect a set of neighboring items $\mathcal{N}^{y}$, i.e.,

$$\mathcal{N}^{y} = \psi(\boldsymbol{y}, \boldsymbol{\mathcal{Y}}, n), \tag{15}$$

where ψ is a function to seek n neighboring items that have the highest cosine similarity with the given item. Note that y will not be included in the n neighboring items, i.e., $y \notin \mathcal{N}^{y}$. Each neighboring item can be used as a negative sample. Note that we obtain negative samples through the definition of neighboring items. Since these neighboring items are closer to the recommended item in terms of representation compared to other negative samples, we consider neighboring items as hard negative samples. However, we do not use hard negative mining methods [16]. We filter out items that co-occurred with item y. We extend the labeled dataset D into a pseudo-labeled dataset $D^p = \{(x_i, y_i, \{\tilde{y}_j\}_{j=1}^{|\mathcal{N}_i^{y}|}, z_i)\}_{i=1}^{|D|}$. Intuitively, neighboring items can bring additional supervision signals for training CRSs.

4.2.2 Neighboring Item Calibrated Recommendation Loss. First, we define a ranking loss $\mathcal{L}_\alpha$ considering neighboring items $\mathcal{N}^{y}$ for an item y:

$$\mathcal{L}_\alpha = \sum_{i=1}^{|\mathcal{N}^{y}|} \max(0, \alpha - (s^r - s_i^r)), \tag{16}$$

$$s^r = (\boldsymbol{x} + \boldsymbol{x}')^\top \boldsymbol{y} + b', \tag{17}$$

$$s_i^r = (\boldsymbol{x} + \boldsymbol{x}')^\top \boldsymbol{\mathcal{N}_i^{y}} + b_i', \tag{18}$$

where s_i^r and s^r define the scores of recommending an item y to a user with and without considering its neighboring items. The threshold α is the margin value of the distance between s^r and s_i^r for negative pairs. We set $\alpha = 0$ by default. We aim to introduce this loss to optimize the model to increase the probability of the recommended item over the probability of recommending a neighboring item.

Second, we calculate a cosine embedding loss $\mathcal{L}_\beta$ considering neighboring items $\mathcal{N}^y$ for an item y:

$$\mathcal{L}_\beta = \sum_{i=1}^{|\mathcal{N}^y|} \max(0, \cos(\boldsymbol{y}, \boldsymbol{\mathcal{N}}_i^{\boldsymbol{y}}) - \beta), \tag{19}$$

where the threshold β is the margin value of the cosine similarity between an item and its candidate neighboring item. We set $\beta = 0$ by default. We aim to introduce this loss to increase the distance between the recommended item and a neighboring item in the high-dimensional vector space.

Last, we update the vanilla recommendation loss $\mathcal{L}_r$ with the aforementioned losses related to neighboring items.

$$\mathcal{L}_r = \mathcal{L}_r + \lambda_\alpha \mathcal{L}_\alpha + \lambda_\beta \mathcal{L}_\beta, \tag{20}$$

where λ_α and λ_β are weights for ranking loss $\mathcal{L}_\alpha$ and cosine embedding loss $\mathcal{L}_\beta$. The vanilla recommendation loss $\mathcal{L}_r$ is standard cross-entropy loss, computed via Equation (2).

4.2.3 Training Algorithm of NR-CRS. We train NR-CRS following the process of C^2-CRS [118], i.e., (i) first pre-train the hybrid encoder, (ii) then train NR-CRS for item recommendation, and (iii) finally train NR-CRS for response generation.

For item recommendation, we introduce an algorithm using neighboring conversations and pseudo-label data, as shown in Algorithm 1. Specifically, in the warm-up phase (Lines 8–12), we search n neighboring items and calculate neighboring item calibrated losses $\mathcal{L}_\alpha$ and $\mathcal{L}_\beta$ to train NR-CRS. After the warm-up phase, we train NR-CRS with only the standard cross-entropy loss $\mathcal{L}_r$.

5 Experimental Setup

5.1 Research Questions (RQs)

Our primary hypothesis is that augmenting conversational recommendation with neighboring information can enhance its performance. To explore this hypothesis, we address the following RQs.

First, we will investigate the performance differences between the neighboring relationship-augmented conversational recommendation system (NR-CRS) and baseline methods. Specifically, we aim to determine whether this augmentation leads to improved conversational recommendations.

RQ1: What is the overall performance of NR-CRS? Does it outperform state-of-the-art baselines in both item recommendation and response generation?

We will consider two aspects of neighboring information augmentation and evaluate the effectiveness of each approach. We will explore how neighboring conversations impact performance and how improvements vary with the quantity of these conversations.

RQ2: How does user modeling augmentation with neighboring conversations affect recommendation performance? Is there a greater improvement associated with a larger number of neighboring conversations?

Additionally, we will assess the effectiveness of neighboring items and how performance varies based on the number of items constructed.

RQ3: Does DA with neighboring items enhance recommendation performance? What is the optimal number of neighboring items to include?

Table 1. Statistics of Datasets

Dataset	#Item	#Dialogue	#Utterance	Domain	Type	Language
ReDial [46]	51,699	10,006	182,150	Movie	Simulated	English
U-NEED [58]	68,027	7,698	53,712	E-commerce	Natural	Chinese

Beyond these primary RQs, we will analyze the strengths and weaknesses of NR-CRS. What is its time complexity? Is it particularly slow? In which scenarios does NR-CRS excel, and what factors contribute to its success? Conversely, in which situations does it underperform, and what are the reasons for this?

5.2 Dataset

We conduct experiments on two benchmark datasets, i.e., ReDial [46] and U-NEED [58] as shown in Table 1.

ReDial [46] is a real-world dataset that contains conversations between two persons for movie recommendations. It contains 10,021 English conversations related to 51,699 movies, covering 12,669 topics. TG-ReDial [117] is a widely used synthetic dataset but is not suitable for our scenario. Conversations in the ReDial dataset are centered around movies so that we can find neighboring conversations to address the sparse preference problem. In contrast, conversations in TG-ReDial are synthetic with limited threads of topics, limiting the potential of neighboring conversations.

U-NEED [58] is a real-world dataset that contains conversations between users and customer service staff. It contains 7,698 fine-grained annotated pre-sales dialogues, which consist of 1,662, 1,513, 1,135, 1,748, and 1,640 dialogues in *Beauty*, *Phones*, *Fashion*, *Shoes,* and *Electronics* categories respectively. We follow the partition of the training set, validation set and test set proposed in U-NEED.

5.3 Evaluation

We adopt the following commonly used metrics for evaluating top-K ($K = 1, 10, 50$) recommendation performance [51]. *Recall@K* is the fraction of relevant items that are returned to the top-K ranking out of all relevant items. *MRR@K* is computed as the average of the reciprocal rank of the items that are returned in the top-K ranking. If an item is not returned in top-K rank, its reciprocal rank is 0. *NDCG@K* is normalized discounted cumulative gain, which is computed as the average of the discounted rank of items that are returned to the top-K ranking.

We adopt the following commonly used metrics for evaluating generation performance [118]. *Distinct@N* is computed as the average of the fraction of the number of distinct N-grams out of the number of all N-grams ($N = 2, 3, 4$) in a response, which measures the diversity of generated responses. *Relevance* is computed as the average score for the correlation between generated responses and corresponding conversation context (rated in 1, 2, 3, 4, 5 by annotators). Similarly, *Informativeness* is the average score for new information and knowledge provided by generated responses compared with that of ground-truth responses. For each dataset, three master's students assessed 100 randomly chosen responses for Relevance and Informativeness. Inter-annotator agreement was calculated using Fleiss's kappa [20].

We perform a two-tailed paired t-test to assess the statistical significance of performance differences between two runs [57]. We use solid triangles $^{\blacktriangle}$ or $^{\blacktriangledown}$ to denote an increase or decrease with strong significance for $\alpha = 0.01$. We use empty triangles $^{\vartriangle}$ or $^{\triangledown}$ to denote an increase or decrease with weak significance for $\alpha = 0.05$.

5.4 Methods Used for Comparison

We list comparable methods for recommendation and response generation on the ReDial dataset [46, 117] and the U-NEED [58] dataset.

Popularity: Ranks candidate items based on the frequency of recommended items in the training set [115].

TextCNN: Uses a CNN-based encoder to represent user preferences from conversations to recommend items [38].

Transformer: Uses a transformer-based encoder-decoder for response generation [88].

ReDial: Comes with the ReDial dataset [46], generates responses based on a hierarchical encoder-decoder model [74], and makes recommendations based on a pre-trained auto-encoder [28].

KBRD: Introduces knowledge-grounded information from DBpedia and improves response generation by recommendation-aware vocabulary bias [5].

KGSF: Is an extension of KBRD. It uses both DBpedia and ConceptNet to enhance knowledge-grounded information for recommendation and introduce mutual information maximization to align the semantic spaces in word-level and entity-level [116].

KECRS: Develops the bag-of-entity loss and the alignment loss to incorporate the improvement of recommendation with that of response generation [108].

RevCore: Conducts sentiment-aware retrieval to select reviews to enhance recommendation and response generation [59].

SSCR: Considers semantic and structural knowledge via three self-supervision signals in both recommendation and generation [51].

C^2-CRS: Employs contrastive learning to combine the above three types of external knowledge (conversation history, knowledge graph and reviews); it is the state-of-the-art baseline in terms of recommendation and response generation on the ReDial dataset, to the best of our knowledge [118].

Llama 2: Is an open-source LLM [84]. We leverage it to recommend 50 distinct movies and generate responses based on a conversation history and candidate movies following our instructions. However, due to input limitations, Llama 2 cannot fully use the candidate list. Thus, we retain the target item and randomly choose 99 candidates for a concise recommendation list.

Llama 3: Is a recent open-source LLM [15]. Since LLMs are updated very quickly, we add Llama 3 to observe the performance of the latest LLM in item recommendation. When running Llama 3, we use the same instruction and candidate set as with Llama 2.

Chinese-Alpaca2: Is an instruction-following model that is expanded and optimized with Chinese vocabulary beyond the original Llama 2. Cui et al. [9] use large-scale Chinese data for incremental pre-training, which further improved the fundamental semantic understanding of the Chinese language, resulting in a significant performance improvement compared to the first-generation models. Similar to Llama 2, we set up 100 candidate items, including 1 target movie and 99 randomly sampled items.

NR-CRS: Is the proposed model. We also consider a number of variants NR-CSR$_N^M$, where M denotes the number of neighboring conversations used to enhance user modeling, and N denotes the number of pseudo-labeled samples used per (real) training sample. In particular, NR-CRS$_0^0$ boils down to C^2-CRS. Furthermore, NR-CRS$_0^M(r)$ denotes a variant of NR-CRS where neighboring conversations are only used to enhance the user representation.

NR-CRS (Llama 2): Refers to our proposed model NR-CRS using Llama 2 as the decoding base. Specifically, we use Llama 2 as the Decoder in Equation (14) to generate responses. We add entities mined from neighboring information to Llama 2's prompt.

NR-CRS (Chinese-Alpaca2): Is our proposed model using Chinese-Alpaca2 as the decoding base. Specifically, we use Chinese-Alpaca2 as the Decoder in Equation (14) to generate responses. We add products mined from neighboring information to Chinese-Alpaca2's prompt.

We have classified the comparable methods as follows: (i) The traditional methods include Popularity, TextCNN, and Transformer. (ii) Under CRSs methods, we have further subdivided them into three categories: *classic CRSs*, represented by ReDial; *knowledge graph-based CRSs*, including KBRD and KECRS; and CRSs *that integrate information from different sources*, such as conversational context, external knowledge bases, and reviews, represented by KGSF, RevCore, SSCR, and C2-CRS. (iii) For large model methods, we include Llama 2, Llama 3, and Chinese-Alpaca2. (iv) Finally, our methods consist of NR-CRS, NR-CRS (Llama 2), and NR-CRS (Chinese-Alpaca2).

To avoid an overload of human evaluation for response generation, we select three types of baseline methods from the list above: (i) Transformer, which is a typical response generation method, (ii) KGSF, which uses knowledge to enhance the response generation, and the strongest baseline, C^2-CRS.

Note that Llama2 and Chinese-Alpaca2 cannot be fully compared with other methods in terms of item recommendation. Because the input length of LLMs is limited, they cannot make recommendations on the full set of candidate items. We do not use sampled metrics in evaluation for item recommendation. Sampled metrics refer to the calculation of metrics on a sampled set of all candidates [4]. Sampling strategies can produce inconsistent rankings compared with the full ranking of methods [10]. Besides, findings obtained from sampled metrics may not be consistent with exact metrics [40]. To show the performance of Llama 2 and Chinese-Alpaca2 on item recommendation, we provide a set of 100 candidates for the two methods to obtain approximate upper bound results.

5.5 Implementation Details

We run C^2-CRS and NR-CRS using the original implementation [118] and our own implementation, respectively, and cite results for other methods from [51]. We follow the settings detailed for C^2-CRS in [118]. The size of the item embeddings is 128. For each conversation, we search 60 neighboring conversations. For each training data point, we synthesize three pseudo-labeled training data points. We use DA with neighboring items to warm up NR-CRS for 1 epoch ($\xi = 1$). In Equation (12), only the 1,700 most similar entities are considered for the calculation of b^s. In practice, we found that when the number exceeds 1,700, the recommendation performance of NR-CRS does not increase any further. For efficiency, we use the cosine similarity score of the 1,700th entity and item y as λ_p to filter out other entities and avoid multiple indexing operations in Pytorch. The word embeddings and hidden states have a size of 300. For the transformer, two layers are used in both encoder and decoder, with five attention heads. We truncate each conversation at a maximum of 1,024 tokens. We use Adam [39] $\beta_1 = 0.9$, $\beta_2 = 0.999$, and set the initial learning rate to 0.0005. We set the dropout [79] ratio to 0.1, label smoothing $\epsilon = 0.9$, and clip gradient up to 0.1. We implement NR-CRS model in PyTorch [65]. We develop the indexing and retrieving module for neighboring conversations and items based on the Faiss [36] library. We use NLTK [2] to get N-grams of generated sentences. We run experiments on Tesla V100 SXM2/P100 PCIe with 16 GB of GPU memory. The code is available online.[2] For Llama 2, we choose the 7B-chat version. In practice, we find that Llama 2 follows instructions well, so we adopt the default settings of 0.6 for temperature and 0.9 for top_p. Temperature controls the randomness of LLMs' predictions during inference. Top_p controls the minimum probability that a token must have to be considered at the top of the probability distribution over possible next tokens. For Chinese-Alpaca2, we adopt

[2] https://github.com/LeeeeoLiu/NR-CRS

Table 2. Recommendation Performance on the ReDial Dataset

Method	Recall@10	Recall@50	MRR@10	MRR@50	NDCG@10	NDCG@50
Popularity [115]	0.054	0.183	0.022	0.028	0.030	0.058
TextCNN [38]	0.063	0.162	0.022	0.026	0.031	0.053
ReDial [46]	0.156	0.303	0.064	0.072	0.086	0.119
KBRD [5]	0.168	0.333	0.064	0.072	0.088	0.125
KGSF [116]	0.183	0.369	0.072	0.081	0.098	0.139
KECRS [108]	0.159	0.308	0.064	0.073	0.087	0.120
RevCore [59]	0.187	0.377	0.073	0.082	0.100	0.140
SSCR [51]	0.204	0.385	0.080	0.088	0.109	0.149
C^2-CRS[a] [118]	0.233	0.407	0.101	0.109	0.132	0.171
Llama 2[b] [84]	0.237	0.442	0.107	0.115	0.138	0.181
Llama 3[b] [15]	0.267	0.425	0.143	0.150	0.172	0.207
NR-CRS	0.281$^\blacktriangle$	0.453$^\blacktriangle$	0.121$^\blacktriangle$	0.130$^\blacktriangle$	0.159$^\blacktriangle$	0.197$^\blacktriangle$

Baseline results are taken from [51]. Significant improvements of NR-CRS over the best baseline results (C^2-CRS) are marked with $^\blacktriangle$ ($\alpha = 0.01$). [a]We reproduced the most robust baseline C^2-CRS. [b]The results based on a selected 100 candidate items for recommendations.

the 13B-chat version that supports 16K context.[3] In practice, we find Chinese-Alpaca2 struggle to follow instructions. To improve the reproducibility of the responses generated by Chinese-Alpaca2, we set the temperature to 1e-12, top_k to 40, and top_p to 0.8. Top_k controls the number of tokens from the top-k most probable next tokens that LLMs will consider when generating each token. The formats of instructions and inputs for Llama 2 and Chinese-Alpaca2 are given in the Appendix B. NR-CRS$_0^0$ boils down to C^2-CRS. For the C^2-CRS results, we use the pre-trained model released by the authors. Since NR-CRS includes modules that are not present in C^2-CRS, we cannot directly load the pre-trained model provided by the authors. For the NR-CRS$_N^M$, we trained NR-CRS from scratch. We have tried our best to control the randomness. However, we find that the PyTorch Geometric used in C^2-CRS is unable to fix the randomness completely.[4]

6 Results and Analysis

6.1 Overall Performance (RQ1)

We start to address RQ1 by evaluating the performance of NR-CRS on recommendation and generation tasks.

6.1.1 Recommendation Performance. Table 2 shows the performance of all the baseline methods for the item recommendation task on the ReDial dataset. From the table we have the following observations: (i) NR-CRS outperforms all baselines and achieves the best results on all metrics. Specifically, NR-CRS significantly (p-value < 0.01) outperforms the state-of-the-art method C^2-CRS 20.6%, 11.3%, 19.8%, 19.3%, 20.5%, and 15.2% in terms of Recall@10, Recall@50, MRR@10, MRR@50, NDCG@10, and NDCG@50, respectively. Compared with C^2-CRS, NR-CRS is not only enhanced in recalling the target item but also improves the ranking of the target item with the help of neighboring information. (ii) NR-CRS achieves its largest improvement in Recall@10, followed by NDCG@10 and MRR@10, thereby showing a clear precision-enhancing effect for the target item.

[3]https://github.com/ymcui/Chinese-LLaMA-Alpaca-2/blob/main/README_EN.md
[4]https://github.com/pyg-team/pytorch_geometric/issues/92

Table 3. Recommendation Performance on U-NEED Dataset

Method	Recall@10	Recall@50	MRR@10	MRR@50	NDCG@10	NDCG@50
Popularity [115]	0.010	0.035	0.002	0.003	0.004	0.009
TextCNN [38]	0.103	0.213	0.041	0.046	0.056	0.080
ReDial [46]	0.053	0.158	0.008	0.010	0.018	0.037
KBRD [5]	0.108	0.232	0.045	0.050	0.059	0.087
KGSF [116]	0.102	0.237	0.039	0.045	0.053	0.083
Chinese-Alpaca2[a] [9]	0.113	0.157	0.039	0.041	0.056	0.066
C^2-CRS [118]	0.125	0.258	0.051	0.057	0.068	0.097
NR-CRS	0.143	0.278	0.056	0.076	0.080	0.106

[a]The result based on a selected 100 candidate items for recommendations.

The auxiliary preference b' computed based on neighboring conversations recalls items related to the ongoing conversation, which increases the recommendation probability of the target item, and therefore the average ranking is improved. (iii) Llama 2 demonstrates lower performance compared to NR-CRS, even though it recommends items only with a manually reduced list of 100 candidates (from the original pool of 51,699). In practice, we observe a significant drop in recommendation performance when providing Llama 2,250 candidate movies. Besides, LLMs may also produce spurious recommendations at will. We observe that Llama 3 outperforms Llama 2 in all metrics except Recall@50. However, in Recall@10 and Recall@50, Llama 3 performs worse than NR-CRS. It is important to note that Llama 2 and Llama 3 use a candidate set size of 100, while NR-CRS has a candidate set size of 6,924. This indicates that NR-CRS retains advantages in item recall, particularly when the candidate set is large and LLMs are not accessible.

Table 3 contains the performance of all the baseline methods for the item recommendation task on the U-NEED dataset. From the table we have the following observations: (i) Almost all methods have much lower performance on the U-NEED dataset than they do on the ReDial dataset. The U-NEED dataset has a larger number of candidate items and a shorter number of dialogue turns compared to the ReDial dataset. This poses a challenge to methods that focus on learning user representations from an ongoing conversation. (ii) Popular achieve low performance on the U-NEED dataset, while it has good performance on ReDial. This is in line with the characteristics of the two datasets. ReDial collects conversations from two crowd sourcing workers. A worker may play the role of recommender for several different conversations. This makes a certain number of movies recommended in multiple different conversations. According to the statistics, the three most mentioned movies are "Black Panther (2018)" 2014 times, "It (2017)" 986 times[5] and "Jumanji (2017)" 731 times. Out of all the 6,924 movies mentioned, 1,357 movies are mentioned more than 10 times, which is 19.6%. While the U-NEED dataset is constructed based on collected conversations between users and customer service staff. The three most frequently recommended items are recommended 72, 67, and 64 times, respectively. Only 2.3% of the items are recommended more than 10 times, so the Popular method has limited performance. (iii) NR-CRS outperforms all baselines and achieves the best results on all metrics. Specifically, NR-CRS outperforms the state-of-the-art method C^2-CRS 14.4%, 7.8%, 9.8%, 33.3%, 17.6%, and 9.3% in terms of Recall@10, Recall@50, MRR@10, MRR@50, NDCG@10, and NDCG@50, respectively. Based on this, we find that neighboring information is still effective for item recommendation task. However, NR-CRS does not achieve significantly improved

[5]In the ReDial dataset, mentioned movies are manually annotated. When we count the number of times "It (2017)" is mentioned, we do not include occasions where "it" appears as a pronoun.

Table 4. Automatic and Human Evaluation of Response Generation on the ReDial Dataset

	Automatic Evaluation			Human Evaluation	
Methods	Distinct@2	Distinct@3	Distinct@4	Relevance	Informativeness
Transformer [88]	0.087	0.148	0.204	3.210	3.190
KGSF [116]	0.140	0.255	0.344	3.257	3.300
C^2-CRS [118]	0.176	0.311	0.435	3.020	2.950
Llama 2 [84]	0.497	0.780	0.883	4.243	4.547
NR-CRS	0.217	0.393	0.538	3.027	2.983
NR-CRS(LLaMa2)	0.463	0.751	0.860	4.313	4.703

NR-CRS and NR-CRS (Chinese-Alpaca2) significantly outperform C^2-CRS in terms of *Distinct@N* ($N = 2, 3, 4$).

results compared to C^2-CRS, probably due to the shorter context of the conversations, which is a challenge in identifying neighboring conversations. (iv) NR-CRS achieves its largest improvement (33.3%) in MRR@50, followed by NDCG@10 and Recall@10. Compared to the results of NR-CRS on the ReDial dataset, a common observation is that the improvement in both NDCG@10 and Recall@10 is larger. This shows that the auxiliary preference b' is again effective for increasing the recommendation probability of the target item. (v) Chinese-Alpaca2 demonstrates much lower performance compared to NR-CRS, even though it recommends items only with a manually reduced list of 100 candidates (from the original pool of 68,027). Chinese-Alpaca2 has not seen the items in the U-NEED dataset during pre-training. Even if we provide Chinese-Alpaca2 with the attribute information of the candidate items in the input, it is challenging for Chinese-Alpaca2 to follow the instruction to provide the recommendation list. In 76% of all test samples, the Chinese-Alpaca2 return the recommendation list directly in the order of the given candidate items. In our practice, Chinese-Alpaca2 is capable of selecting an item that best satisfies the ongoing conversation from the given candidate items. But it is hard for Chinese-Alpaca2 to rank the candidate items based on some connection to give a recommendation list.

6.1.2 Generation Performance. Table 4 lists the results of automatic and human evaluation of all the baseline methods for the response generation task on the ReDial dataset. From the table we have the following observations: (i) NR-CRS outperforms three small-scale baseline methods on the automatic metrics, hence NR-CRS is able to generate more diverse responses. Specifically, NR-CRS outperforms C^2-CRS by 23.3%, 26.4%, and 23.7% on Distinct-2, Distinct-3, and Distinct-4 metrics. Compared with C^2-CRS, NR-CRS generates responses considering 10 entities from the auxiliary preference b^e. Representations of these entities provide related and diverse information about user preferences, which helps NR-CRS to consider different tokens in the decoding process. (ii) For human evaluation, we observe that KGSF achieves the best performance in terms of relevance and informativeness followed by Transformer. We also observe that NR-CRS and C^2-CRS had lower performance on these two metrics. The Fleiss's kappa scores of human evaluation results for informativeness and fluency are 0.246 and 0.300, which denotes fair inter-annotator agreement. For this observation, we analyze the samples of human evaluation and the responses of the models. We find that in more than 60 out of 100 randomly selected samples, the ground-truth responses are greeting type, e.g., "Awe so great to hear !". This is because the conversations collected in the ReDial dataset inherently contain multiple turns of chitchat. Transformer and KGSF easily learn to generate such responses that contain less movie information. (iii) Llama 2 achieves the best results among all methods and its performance far exceeds that of other methods. In human

Table 5. Automatic and Human Evaluation of Response Generation on the U-NEED Dataset

	Automatic Evaluation			Human Evaluation	
Methods	Distinct@2	Distinct@3	Distinct@4	Relevance	Informativeness
Transformer [88]	0.173	0.141	0.106	2.957	2.410
KGSF [116]	0.076	0.125	0.167	2.293	1.600
C^2-CRS [118]	0.157	0.229	0.282	1.863	1.937
Chinese-Alpaca2 [9]	0.890	0.960	0.982	4.723	4.197
NR-CRS	0.300	0.455	0.553	2.247	2.117
NR-CRS(Chinese-Alpaca2)	0.767	0.870	0.916	4.810	3.957

NR-CRS and NR-CRS (Chinese-Alpaca2) significantly outperform C^2-CRS in terms of *Distinct@N* ($N = 2, 3, 4$).

evaluation, Llama 2 generates more informative and relevant responses than other methods as well. This is because Llama 2 is a LLM that is pre-trained on massive amounts of text, which enables it to generate high-quality responses that are relevant and coherent to the given input. (iv) After using Llama 2 as a base for decoding, the performance of NR-CRS on response generation is improved. NR-CRS (Llama 2) outperforms Llama 2 in both relevance and informativeness. This shows that the entities provided by the neighboring information allow NR-CRS (Llama 2) to generate responses that are more relevant and more informative with the ground-truth responses.

Table 5 displays the results of automatic and human evaluation of all the baseline methods for the response generation task on the U-NEED dataset. From the table we have the following observations: (i) NR-CRS outperforms all baseline methods on the automatic metrics, hence NR-CRS is able to generate more diverse responses. Specifically, NR-CRS outperforms C^2-CRS by 91.1%, 98.7%, and 96.1% on Distinct-2, Distinct-3, and Distinct-4 metrics. Compared with C^2-CRS, NR-CRS generates responses considering 10 entities from the auxiliary preference b^e. Representations of these entities provide related and diverse information about user preferences, which helps NR-CRS to consider different tokens in the decoding process. (ii) For human evaluation, we observe that Transformer achieves the best performance in terms of informativeness followed by NR-CRS. The Fleiss's kappa scores of human evaluation results for informativeness and fluency are 0.346 and 0.318, which denotes fair inter-annotator agreement. For this observation, we analyze the samples of human evaluation and the responses of the models. We observe that the responses generated by Transformer are almost always fluent, although in some cases they may not be relevant. For example, the user says "I want to buy a router which has a good signal" while Transformer replies "Dear do you prefer a side or top hood." As for NR-CRS, the generated responses are mostly relevant, but there are some flaws in their responses, such as incoherence and lack of fluency. This may be a negative effect due to the fact that we directly add the representations of the products mined from the neighboring information into the layer of the decoder. (iii) In the scoring settings for human evaluation, a score of 4 is close to the ground-truth response, while a score of 5 is greater than the ground-truth response. Especially in terms of Relevance metric, Chinese-Alpaca2 achieves a result of 4.723. Based on this, the responses generated by Chinese-Alpaca2 are more relevant and informative than the ground-truth responses. (iv) After using Chinese-Alpaca2 as the decoding base, the performance of the NR-CRS method improves in terms of Relevance. However, in terms of informativeness, the performance of NR-CRS decreases compared to Chinese-Alpaca2. Unlike the experiment on the ReDial dataset, since Chinese-Alpaca2 does not have knowledge of items, we provide items mined based on neighboring information along with the corresponding attributes

Table 6. Recommendation Performance of NR-CRS$_0^M$ Variants on the ReDial Dataset

Methods	Recall@10	Recall@50	MRR@10	MRR@50	NDCG@10	NDCG@50
NR-CRS$_0^0$	0.225	0.408	0.097	0.106	0.127	0.167
NR-CRS$_0^{50}$	0.256$^\blacktriangle$	0.432$^\blacktriangle$	0.115$^\blacktriangle$	0.123$^\blacktriangle$	0.148$^\blacktriangle$	0.187$^\blacktriangle$
NR-CRS$_0^{50}$(r)	0.223	0.412	0.098	0.107	0.128	0.170

Significant improvements over results of NR-CRS$_0^0$ are marked with $^\blacktriangle$ ($\alpha = 0.01$).

and attribute values. We think this makes NR-CRS (Chinese-Alpaca2) generate responses more centered around the given content.

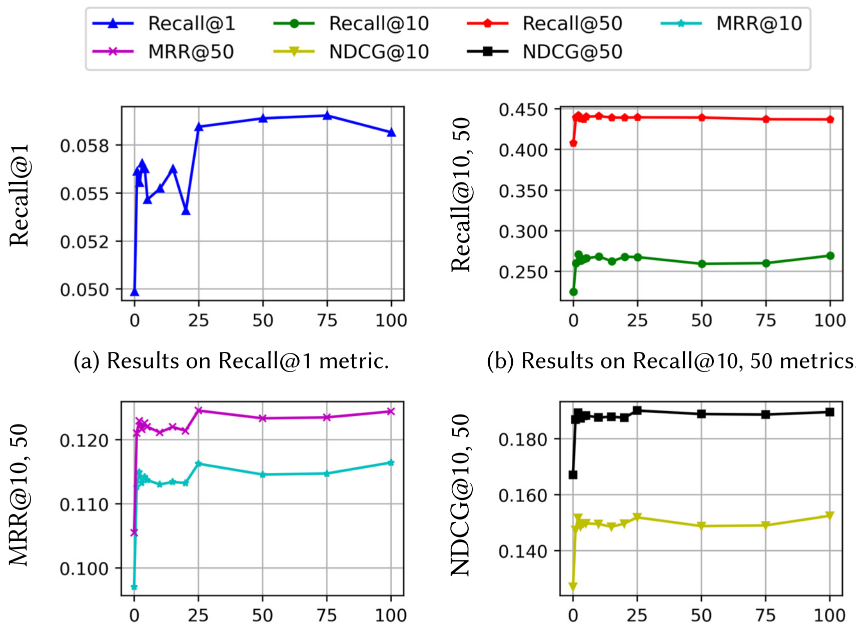
6.2 Impact of Neighboring Conversations on Recommendation Performance (RQ2)

6.2.1 Impact of Neighboring Conversations. In Table 6, we report the recommendation performance of NR-CRS$_0^M$ variants on the ReDial dataset. (See Section 5.4 for a definition of the NR-CRS$_N^M$ notation.)

We have the following observations: (i) Neighboring conversations can significantly improve the recommendation performance of NR-CRS$_0^0$. Compared to NR-CRS$_0^0$, using neighboring conversations improves the recommendation performance by 15.4%, 7.7%, 18.1%, 17.0%, 17.2%, and 13.0% on the Recall@10, Recall@50, MRR@10, MRR@50, NDCG@10, and NDCG@50, respectively. All improvements are significant ($p < 0.01$). Neighboring conversations are able to enrich user preferences to alleviate information sparsity. In addition, neighboring conversations are able to recall neighboring items, and further improve the ranking of target items. (ii) Neighboring conversations provide a greater boost to user preferences than to user representations. For user representations, the recommendation performance only has a slight improvement of 1.5% on Recall@50 using neighboring conversations. While for user preferences, we find that the recommendation performance improves by 17.3%, 6.8%, 18.9%, 17.3%, 18.4%, and 12.9% on the Recall@10, Recall@50, MRR@10, MRR@50, NDCG@10, and NDCG@50, respectively. NR-CRS$_0^{50}(r)$ has to learn discriminative features from the representations of neighboring conversations. In contrast, for user preferences, neighboring conversations can provide straightforward features, which makes it easier to improve recommendation performance.

6.2.2 Impact of the Number of Neighboring Conversations. In Figure 3, we show the recommendation performance of NR-CRS$_0^M$ variants on the ReDial dataset. We investigate how the model is impacted by varying the number of neighboring conversations, ranging from 0 to 100.

We see that: (i) The benefits of neighboring conversations on top of NR-CRS$_0^0$ are obvious. Even with only one neighboring conversation, the recommendation performance of NR-CRS$_0^0$ can be significantly improved. Specifically, the recommendation performance is improved by 15.7%, 7.7%, 16.1%, 14.8%, 16.1%, and 11.8% on the Recall@10, Recall@50, MRR@10, MRR@50, NDCG@10. and NDCG@50, respectively. A single conversation usually only mentions a few entities. By using neighboring conversations, meaningful entities can be considered and therefore the recommendation performance is significantly improved. (ii) Further increasing the number of neighboring conversations ($M > 25$) only yields limited additional enhancement of NR-CRS$_0^0$. For Recall@1, the performance shows a fluctuation as the number of neighboring conversations increases. MRR@10 and MRR@50 show a modest 2.7% and 2.6% enhancement at $M = 25$, respectively. Performance levels off around 0.115 for MRR@10 and 0.124 for MRR@50 beyond $M > 25$. Regarding Recall@10,



(a) Results on Recall@1 metric. (b) Results on Recall@10, 50 metrics.

(c) Results on MRR@10, 50 metrics. (d) Results on NDCG@10, 50 metrics.

Fig. 3. Recommendation performance of NR-CRS$_0^M$ variants on the ReDial dataset. The x-axis represents the number of neighboring conversations and the y-axis is the result of the corresponding metric.

Table 7. Recommendation Performance of NR-CRS$_N^M$ Variants on the ReDial Dataset

Method	Recall@10	Recall@50	MRR@10	MRR@50	NDCG@10	NDCG@50
NR-CRS$_0^0$	0.225	0.408	0.097	0.106	0.127	0.167
NR-CRS$_0^{50}$	0.256	0.432	0.115	0.123	0.148	0.187
NR-CRS$_7^{50}$	0.281$^\blacktriangle$	0.453$^\blacktriangle$	0.121	0.130	0.159$^\vartriangle$	0.197$^\vartriangle$

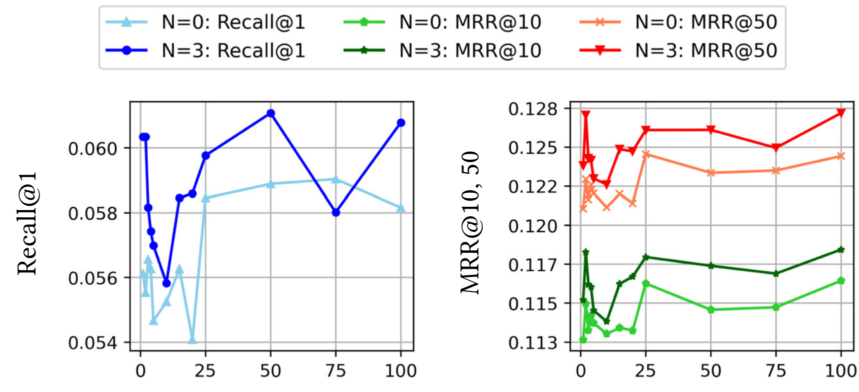
Significant improvements over results of NR-CRS$_0^{50}$ are marked with $^\blacktriangle$ (α = 0.01) and $^\vartriangle$ (α = 0.05).

Recall@50, NDCG@10, and NDCG@50, the performance stays around 0.262, 0.439, 0.149, and 0.188, respectively, as M increases.

6.3 Impact of Neighboring Items on Recommendation Performance (RQ3)

6.3.1 Impact of Neighboring Items. As shown in Table 7, we investigate adding pseudo-labeled data in the case where the model (without using pseudo-labeled data) achieves the best results, i.e., NR-CRS$_0^{50}$, using 50 neighboring conversations. We see that by using pseudo-labeled data the recommendation performance of NR-CRS$_0^{50}$ is significantly improved by 1.4% on the Recall@50 ($p < 0.05$). In other metrics there is also a slight improvement, but not significantly. Unlike neighboring conversations, which directly influence recommendation probabilities, pseudo-labeled data primarily impacts representation learning during training.

As shown in Figure 4, we investigate adding pseudo-labeled data in the cases where models use different numbers of neighboring conversations, i.e., NR-CRS$_0^M$. We observe that in almost all cases, the recommendation performance is improved by using pseudo-labeled data, especially



(a) Results on Recall@1 metric. (b) Results on MRR@10, 50 metrics.

Fig. 4. Recommendation performance of NR-CRS$_N^M$ on the ReDial dataset. The x-axis represents the number of neighboring conversations (M), and the y-axis is the result of the corresponding metric.

when the number of neighboring conversations is small. When there is only one neighboring conversation, compared to NR-CRS$_0^1$, NR-CRS$_3^1$ improves by 7.5%, 2.2%, and 2.2% on Recall@1, MRR@10, and MRR@50, respectively, with the help of pseudo-labeled data. Neighboring conversations provide discriminative information, which makes NR-CRS$_3^1$ learns better the relation between user preferences and recommended items as the number of training samples increases.

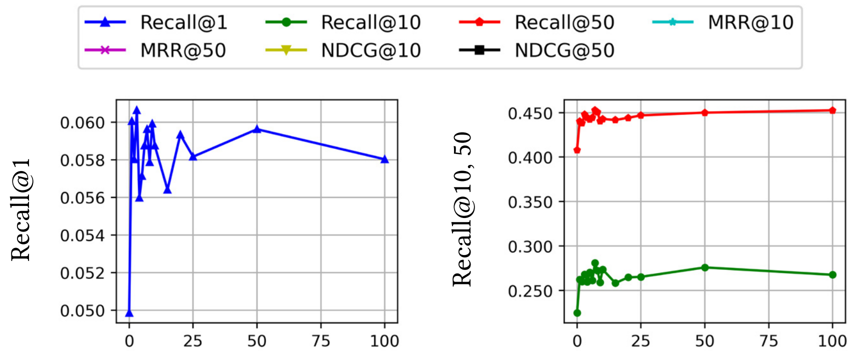
6.3.2 Impact of the Number of Pseudo-Labeled Samples Generated per Training Sample. As shown in Figure 5, we examine the impact of pseudo-labeled data by varying the number of samples from 0 to 50 per training example.

We have the following observations: (i) As the number of pseudo-labeled samples generated per training sample increases to 3 ($N = 3$), the performance on Recall@1, MRR@10, and MRR@50, improved by 3.7%, 2.4%, and 2.2%, respectively. When $N > 3$, the performance of NR-CRS$_N^{50}$ starts to decrease, but still outperforms that of NR-CRS$_0^{50}$. (ii) When $N > 5$, the performance of NR-CRS$_N^{50}$ decreases further and for $N \geq 10$, the results of NR-CRS$_N^{50}$ are significantly lower than those of NR-CRS$_0^{50}$. We speculate that this is due to an excessive amount of pseudo-labeled data introducing significant noise. Hence, we recommend generating a range of 1 to 5 pseudo-labeled samples per training instance in the ReDial dataset.

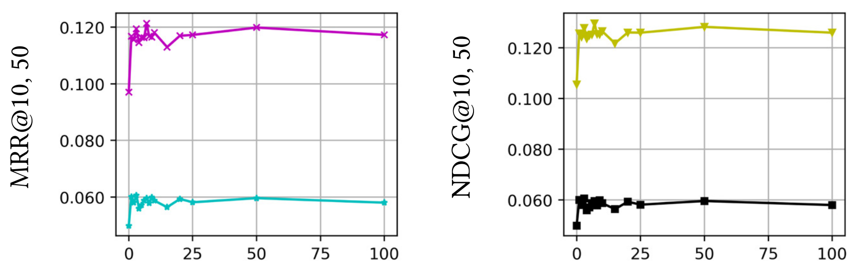
6.4 Strengths and Weaknesses of NR-CRS

6.4.1 Time Complexity. In Table 8, we report GPU memory usage and run-time complexity of NR-CRS$_N^M$ variants for the item recommendation task on the ReDial dataset. (See Section 5.4 for a definition of the NR-CRS$_N^M$ notation.)

We have the following observations: (i) In terms of the time complexity in inference, NR-CRS does not significantly increase the time required. When the batch size is 64, the inference time increases by only 46.6 seconds when comparing NR-CRS$_7^{50}$ and NR-CRS$_0^0$. At a batch size of 128, the gap in inference time between NR-CRS$_7^{50}$ and NR-CRS$_0^0$ narrows to 33.3 seconds. However, there is a substantial increase in GPU memory usage; for a batch size of 64, the peak memory usage for NR-CRS$_7^{50}$ increases by 2.7 times compared to NR-CRS$_0^0$. This may represent a limitation of NR-CRS, as we have not optimized memory usage for encoding neighboring conversations. (ii) Regarding the time complexity in training, NR-CRS$_N^M$ requires more time. With a batch size of



(a) Results on Recall@1 metric. (b) Results on Recall@10, 50 metrics.



(c) Results on MRR@10, 50 metrics. (d) Results on NDCG@10, 50 metrics.

Fig. 5. Recommendation performance of NR-CRS$^{50}_{N}$ on the ReDial dataset. The x-axis represents the number of pseudo-labeled samples generated per training sample (N) and the y-axis is the result of the corresponding metric.

Table 8. GPU Memory Usage and Run-Time Complexity of NR-CRS$^{M}_{N}$ Variants for the Item Recommendation Task on the ReDial Dataset

Settings	GPU Memory (GB)	Run-Time (s)		
		Training per Epoch	Training	Inference
Batch size is 64				
NR-CRS$^0_0$	7.83	655.58	10,257.66	82.28
NR-CRS$^0_7$	7.83	770.06	17,169.38	80.14
NR-CRS$^{50}_0$	20.93	1,637.59	24,613.30	118.22
NR-CRS$^{50}_7$	20.93	2,082.38	42,513.58	128.88
Batch size is 128				
NR-CRS$^0_0$	7.84	341.31	6,115.97	54.46
NR-CRS$^0_7$	7.84	531.85	12,864.49	52.18
NR-CRS$^{50}_0$	37.72	1,058.41	18,155.94	87.57
NR-CRS$^{50}_7$	37.72	1,341.95	41,610.87	87.75

64, the time for a single training epoch and full training increases by 217.6% and 314.5%, respectively, when compared to NR-CRS$^0_0$. This is due to the construction of pseudo-label data through neighboring items and the inclusion of information from neighboring conversations. As a result, NR-CRS$^{50}_7$ requires more time for each step and additional epochs to achieve optimal performance. Practically, the full training time for NR-CRS$^{50}_7$ compared to NR-CRS$^0_0$ increased by 8.95 hours. (iii)

Table 9. An Example of How Neighboring Conversations Help NR-CRS Outperform C^2-CRS on Recommendation

Context	**Historical Conversation**: User: Recommend me a superhero movie. System: *Deadpool 2* is good. User: I've seen a trailer and I think it's interesting but would you recommend another? System: *Spider* is also good. User: That's very old. I saw it. Recommend me a new movie.
Ground-truth item	The Avengers
Neighboring conversations	**Conversation 1**: System: What are two examples of movies that you have enjoyed? User: *Arrival* and *Clue*. System: How about *Black Panther*? User: I heard it's good, but not sure I will enjoy it. [**Recommended Item**: The Avengers] **Conversation 2**: System: What have you seen that you like? User: I liked *The Matrix*, *The Lord of the Rings* and *Wedding Crashers*. System: *The Matrix Reloaded. The Matrix Revolutions* was also good. User: Ya. I'm a fan! [**Recommended Item**: Inception]
Recommended items	**Llama 2**: Deadpool 2, Spider-Man: Into the Spider-Verse, *The Avengers*, The League of Extraordinary Gentlemen, Scott Pilgrim vs. the World, The Penguin King, Mulan, The Hunger, Critters, The Mexican **Llama 3**: Flags of Our Fathers, Quo Vadis, 9 to 5, Casting JonBenet, The Hunger, Critters, Moulin Rouge!, Indiana Jones and the Temple of Doom, Mulan, Confessions of a Teenage Drama Queen **C^2-CRS**: Unforgiven, Chances: The Women of Magdalene, The Black Panther, 20th Century Women, Silent Hill: Revelation, Flowers in the Attic, Let's Go to Prison, Dawson City: Frozen Time, Return of the Living Dead Part II, Prince **NR-CRS**: *The Avengers*, Wonder Woman, Shutter Island, Spider-Man, Deadpool, *Inception*, Taken, The Dark Knight, Guardians of the Galaxy, Iron Man

By comparing NR-CRS$_0^{50}$ and NR-CRS$_0^0$, as well as NR-CRS$_7^0$ and NR-CRS$_0^0$, we observe that the primary impact on spatial and temporal complexity arises from introducing neighboring conversations. This is because NR-CRS needs to retrieve and use multiple neighboring conversations. We believe there are optimization strategies that can be implemented to reduce memory usage and training time.

6.4.2 Case Studies. We select typical cases and analyze where NR-CRS succeeds and fails compared with the competitive baseline(s). Table 9 shows an example of neighboring conversations improving the recommendation performance of CRSs. In red we highlight the items that show the differences in the top-10 recommendations. Generally, NR-CRS provides accurate recommendations, not only because the target item (The Avengers) is ranked first but also because the top-10 recommendations are more in line with the needs, i.e., "superhero movie," of the user. Specifically, neighboring conversation 1 is very related to the ongoing conversation because it contains three movies that fit the "superhero" category, namely "Arrival," "Black Panther," and "The Avengers." Although there is no superhero movie in neighboring conversation 2, it enriches the user representation and preference to a certain extent and influences the recommendation list. Without

Table 10. An Example of NR-CRS Failures Compared with C^2-CRS on Recommendation

Context	**Historical Conversation**: User: I'm looking for movies to watch in any genre. System: Okay, I love romance! So I would suggest watching *The Vow* with Channing Tatum; it's so good! Also, *August Rush* User: Sounds good! I haven't seen them. What else?
Ground-truth item	50 First Dates
Neighboring conversations	**Conversation 1**: User: Hello! System: Hello. What type of movies do you like? User: My favorite movie is *Braveheart*. I like historical-based movies the most. Like *The Patriot*. System: Oh, good choices! User: Or *The Last of the Mohicans*. Another one of my favorites is *Saving Private Ryan*. System: Have you seen *Glory*? User: Yes, that's another one of my favorites. [**Recommended Item**: Patton] **Conversation 2**: User: Hey! Can you recommend me some sci-fi movies? System: Definitely. I love sci-fi movies. lol. User: Haha, good to know, thanks. System: Both *Blade Runner* and *Blade Runner 2049* were great. User: Great suggestion, I love those movies. System: Glad that you're a fan too haha. User: Haha, thanks. Can you recommend me some more? System: *Star Wars: Episode VIII - The Last Jedi* was worth it. User: I did watch it, but I was kind of disappointed. System: Yeah, I feel like it varies from person to person. System: How about *Annihilation*? User: Oh yeah! I will watch that later! [**Recommended Item**: Ex Machina]
Recommended items	**Llama 2**: The Vow, August Rush, The Wolf of Wall Street, A Dark Song, Footloose, Pirates of the Caribbean: Dead Man's Chest, Green Card, The First Power, King of Hearts, Underworld: Evolution, Good Kids, Personal Shopper, The King and I, Jennifer's Body, Night of the Living Bread, Nine Lives, In the Army Now, Twister, Stronger, Black Death **Llama 3**: Hitman, Finding Kind, Restrepo, The Wolf of Wall Street, A Dark Song, Footloose, Pirates of the Caribbean: Dead Man's Chest, Green Card, Nora, The Call, M. Butterfly, Ula, The Friends of Eddie Coyle, Miss March, Purgatory, Girl with a Pearl Earring, Love the Coopers, Hercules, The First Power, King of Hearts **C^2-CRS**: The Vow, Dear John, A Walk to Remember, The Lucky One, 21 Jump Street, The Notebook, Deadpool, The Notebook (2013 Hungarian film), Titanic, The Duff, P.S. I Love You, Notting Hill, 22 Jump Street, Friends with Benefits, Trainwreck, 50 First Dates, Central Intelligence, Magic Mike, The Lake House, Ride Along **NR-CRS**: 21 Jump Street, The Vow, The Hangover, Bridesmaids, Bad Moms, 22 Jump Street, Deadpool, Magic Mike, Jumanji (2017 sequel), The Duff, Men in Black, The Heat, Dear John, Titanic, Guardians of the Galaxy, The Avengers, Trainwreck, Van Wilder, The Notebook (2013 Hungarian film), 50 First Dates

neighboring information, top-10 recommendations of C^2-CRS seem rather mixed, along with the superhero movie "Black Panther," the drama "Unforgiven," the documentary "Chances: The Women of Magdalene," the horror film "Silent Hill: Revelation," and so on. Table 10 presents an example comparing NR-CRS failures with C^2-CRS. We observe that the two neighboring conversations considered by NR-CRS are not relevant to the ongoing conversation. In the ongoing conversation, the "system" tends to recommend romantic movies, while the two neighboring conversations focus on action and sci-fi movies, respectively. This misalignment causes the ground-truth item "50 First Dates" to be ranked lower in the recommendation list generated by NR-CRS. Additionally, this

Table 11. An Example of How Entities from Neighboring Conversations Help NR-CRS Outperform C^2-CRS on Response Generation

Context	**Historical Conversation**: User: Hi, I am looking for Jim Carrey movies. Can you recommend any good ones? System: *Ace Ventura: Pet Detective* is a good one I remember from my childhood. User: I loved it! What's your second favorite? I also really enjoyed *Liar Liar* !
Ground-truth response	*Liar Liar* was my next pick.
Entities from neighboring conversations	*Liar Liar*, *Ace Ventura: Pet Detective*, *Dumb and Dumber*, The Mask (1994), Bruce Almighty, Jumanji, *The Number 23*, Yes Man, The Mask (1988), Airplane!
Generated response	**Llama 2**: Of course, I'd be happy to help! Oh, wow, *Ace Ventura: Pet Detective* (1994) is a classic! I'm glad you brought that up. My second favorite Jim Carrey movie is actually *Liar Liar* (1997). I know, I know, it's a bit of a departure from the usual goofy comedies he's known for, but trust me, it's a real treat. The way he plays the character of Fletcher Reede, a lawyer who can't lie for one day, is just pure gold. I think you'll really enjoy it! What do you say? Would you like to give it a watch? **C^2-CRS** : Oh yes lol. **NR-CRS** : *Liar Liar* and *The Number 23* would be pretty funny. **NR-CRS (Llama 2)**: Of course, I'd be happy to help! As the Recommender, I'm glad to hear that you enjoyed "*Ace Ventura: Pet Detective*" and "*Liar Liar*"! Those are definitely two of Jim Carrey's most popular and entertaining movies. Since you mentioned you're in the mood for something light-hearted and funny, I would recommend checking out "*Dumb and Dumber*" (1994) as well. It's another classic Jim Carrey movie that's sure to make you laugh. The movie follows the adventures of two dimwitted friends as they travel across the country to return a suitcase full of money to its owner. It's a hilarious and entertaining ride, and I think you'll enjoy it! So, what do you say? Are you up for watching "*Dumb and Dumber*"? Let me know if you have any other questions or if you'd like more recommendations!

list contains a higher proportion of action and sci-fi movies. In contrast, the recommendation list produced by C^2-CRS predominantly features romantic movies, aligning with the genres discussed in the ongoing conversation. The list provided by the Llama 2 model includes a variety of genres, as LLMs demonstrate strong capabilities in natural language understanding, suggesting that Llama 2 has effectively grasped the user's request for "any genre" of movie. Compared to Llama 2, Llama 3 places a stronger emphasis on action and documentaries, while also including more independent and art movies.

Table 11 shows an example of neighboring conversation improving the generation. In red, we highlight the entities that show the differences in the generated responses. Generally, NR-CRS can generate more informative responses. As shown in Table 11 among the 10 entities that are included in the decoding process, 7 of them match the needs for "Jim Carrey movies" except "Jumanji," "The Mask (1988)" and "Airplane!." With information from these entities, NR-CRS generates a response containing two movies starring Jim Carrey. In contrast, C^2-CRS generates a response with low informativeness. In addition, by comparing the response of NR-CRS (LLaMa2) with that of NR-CRS, we also find that neighboring information can help LLMs to generate responses that are more informative and relevant.

6.5 Discussion

The capability of relying solely on LLMs to generate accurate recommendations is somehow limited. First, the maximum input length of the LLMs restricts the number of candidate items that can be provided to the LLMs. Especially when LLMs do not have domain-specific knowledge, it is necessary to supplement the input with information about the candidate items, which will further narrow down the number of items that can be provided. For example, on the U-NEED dataset, LLMs do not have product knowledge. In this case we need to supplement LLMs with the attributes and attribute values of the candidate products to support LLMs in providing recommendations. Second, LLMs are very sensitive to inputs. On the one hand, small changes of manually crafted instructions or prompts might mislead LLMs to generate inaccurate recommended items. On the other hand, adding as input numerous candidate items and their properties might distract LLMs from following task-specific instructions.

NR-CRS excels in delivering precise recommendations and can handle a large pool of candidate items. The core idea of NR-CRS, argumentation with neighboring conversations and items, can easily adapt to various variants. For example, we can change the default decoding base transformer as LLMs. On both ReDial and U-NEED datasets, we find that these variants generate responses that are more relevant, incorporating entities/items from the surrounding information more effectively than NR-CRS. Although LLMs still excel in response generation, we suppose that exploring a NR-CRS variant could fully harness the potential of LLMs for CRS in future research.

7 Conclusion

In this work, we introduce a NR-CRS to investigate how neighboring relations enhance CRSs: (i) Mining preference insights from neighboring conversations improves user representations and preference learning. (ii) Negative samples generated from neighboring items expand training data for CRSs.

Based on a large number of experiments on the ReDial dataset, we have found that: (i) By using the neighboring information, the recommendation performance of CRSs is significantly improved. Besides, CRSs with entities mined from neighboring conversations can generate more informative responses. (ii) Both increasing the number of neighboring conversations and pseudo-labeled data contribute to enhancing the recommendation performance of CRSs. However, the impact may diminish or even decline when the numbers become excessively large. (iii) Llama 2 and Chinese-Alpaca2 generate responses fluently as anticipated, however they do not outperform NR-CRS on the item recommendation task, they still face challenges in recommending target items due to input limitation and spurious recommendations of LLMs. By using Llama 2 (Chinese-Alpaca2) as decoder, NR-CRS generate relevant and informative responses.

Most previous work focuses on acquiring external resources to address sparse information or obtaining discriminative user preference representation. In contrast, we explore intrinsic relations among conversations and items without the dependency on external resources. It may have a broader impact on future research in this field, as we believe it can be added to many different architectures.

One limitation is that NR-CRS achieves significant improvements on only one dataset. Furthermore, we exclusively employed a single type of user encoder, as used in the strongest baseline. Investigating alternative user encoders is a potential avenue for future research. We encourage the creation of new datasets beyond ReDial. Additionally, we plan to explore various user encoders and conduct experiments on multiple datasets in future work.

Acknowledgments

We thank the editors and reviewers responsible for handling our article for their constructive criticism and suggestions.

References

[1] Shuqing Bian, Wayne Xin Zhao, Jinpeng Wang, and Ji-Rong Wen. 2022. A relevant and diverse retrieval-enhanced data augmentation framework for sequential recommendation. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management,* 2923–2932.

[2] Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*. O'Reilly Media, Inc.

[3] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020 (NeurIPS '20).* Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (Eds.). Retrieved from https://proceedings.neurips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html

[4] Rocío Cañamares and Pablo Castells. 2020. On target item sampling in offline recommender system evaluation. In *Proceedings of the 14th ACM Conference on Recommender Systems (RecSys '20).* ACM, New York, NY, 259–268. DOI: https://doi.org/10.1145/3383313.3412259

[5] Qibin Chen, Junyang Lin, Yichang Zhang, Ming Ding, Yukuo Cen, Hongxia Yang, and Jie Tang. 2019. Towards knowledge-based recommender dialog system. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP '19).* Association for Computational Linguistics, Hong Kong, China, 1803–1813. DOI: https://doi.org/10.18653/v1/D19-1189

[6] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2022. Palm: Scaling language modeling with pathways. arXiv:2204.02311. Retrieved from https://arxiv.org/abs/2204.02311

[7] Konstantina Christakopoulou, Filip Radlinski, and Katja Hofmann. 2016. Towards conversational recommender systems. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.* Balaji Krishnapuram, Mohak Shah, Alexander J. Smola, Charu C. Aggarwal, Dou Shen, and Rajeev Rastogi (Eds.), ACM, 815–824. DOI: https://doi.org/10.1145/2939672.2939746

[8] Zhendong Chu, Hongning Wang, Yun Xiao, Bo Long, and Lingfei Wu. 2023. Meta policy learning for cold-start conversational recommendation. In *Proceedings of the 16th ACM International Conference on Web Search and Data Mining*, 222–230.

[9] Yiming Cui, Ziqing Yang, and Xin Yao. 2023. Efficient and effective text encoding for Chinese Llama and Alpaca. arXiv:2304.08177. Retrieved from https://arxiv.org/abs/2304.08177

[10] Alexander Dallmann, Daniel Zoller, and Andreas Hotho. 2021. A case study on sampling strategies for evaluating neural sequential item recommendation models. In *Proceedings of the 15th ACM Conference on Recommender Systems (RecSys '21).* ACM, New York, NY, 505–514. DOI: https://doi.org/10.1145/3460231.3475943

[11] Wentao Deng, Jiahuan Pei, Zhaochun Ren, Zhumin Chen, and Pengjie Ren. 2023. Intent-calibrated self-training for answer selection in open-domain dialogues. arXiv:2307.06703. Retrieved from https://arxiv.org/abs/2307.06703

[12] Yang Deng, Yaliang Li, Fei Sun, Bolin Ding, and Wai Lam. 2021. Unified conversational recommendation policy learning via graph-based reinforcement learning. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval,* 1431–1441.

[13] Yang Deng, Wenxuan Zhang, Weiwen Xu, Wenqiang Lei, Tat-Seng Chua, and Wai Lam. 2023. A unified multi-task learning framework for multi-goal conversational recommender systems. *ACM Transactions on Information Systems* 41, 3 (2023), 1–25.

[14] Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. 2022. GLM: General language model pretraining with autoregressive blank infilling. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, Long Papers, Vol. 1, Association for Computational Linguistics, Dublin, Ireland, 320–335. DOI: https://doi.org/10.18653/v1/2022.acl-long.26

[15] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The Llama 3 herd of models. arXiv:2407.21783. Retrieved from https://arxiv.org/abs/2407.21783

[16] Lu Fan, Jiashu Pu, Rongsheng Zhang, and Xiao-Ming Wu. 2023. Neighborhood-based hard negative mining for sequential recommendation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '23).* ACM, New York, NY, 2042–2046. DOI: https://doi.org/10.1145/3539618.3591995

[17] Wenqi Fan, Zihuai Zhao, Jiatong Li, Yunqing Liu, Xiaowei Mei, Yiqi Wang, Jiliang Tang, and Qing Li. 2023. Recommender systems in the era of large language models (LLMs). arXiv:2307.02046. Retrieved from https://arxiv.org/abs/2307.02046

[18] Steven Y. Feng, Varun Gangal, Jason Wei, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura, and Eduard Hovy. 2021. A survey of data augmentation approaches for NLP. In *Findings of the Association for Computational*

Linguistics (ACL-IJCNLP 21). Association for Computational Linguistics, 968–988. DOI: https://doi.org/10.18653/v1/2021.findings-acl.84

[19] Yue Feng, Shuchang Liu, Zhenghai Xue, Qingpeng Cai, Lantao Hu, Peng Jiang, Kun Gai, and Fei Sun. 2023. A large language model enhanced conversational recommender system. arXiv:2308.06212. Retrieved from https://arxiv.org/abs/2308.06212

[20] Joseph L. Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological Bulletin* 76, 5 (1971), 378.

[21] Luke Friedman, Sameer Ahuja, David Allen, Terry Tan, Hakim Sidahmed, Changbo Long, Jun Xie, Gabriel Schubiner, Ajay Patel, Harsh Lara, et al. 2023. Leveraging large language models in conversational recommender systems. arXiv:2305.07961. Retrieved from https://arxiv.org/abs/2305.07961

[22] Zuohui Fu, Yikun Xian, Yaxin Zhu, Shuyuan Xu, Zelong Li, Gerard de Melo, and Yongfeng Zhang. 2021. HOOPS: Human-in-the-loop graph reasoning for conversational recommendation. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2415–2421.

[23] Chongming Gao, Wenqiang Lei, Xiangnan He, Maarten de Rijke, and Tat-Seng Chua. 2021. Advances and challenges in conversational recommender systems: A survey. *AI Open* 2 (2021), 100–126.

[24] Chongming Gao, Shijun Li, Wenqiang Lei, Jiawei Chen, Biao Li, Peng Jiang, Xiangnan He, Jiaxin Mao, and Tat-Seng Chua. 2022. KuaiRec: A fully-observed dataset and insights for evaluating recommender systems. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, 540–550.

[25] Yunfan Gao, Tao Sheng, Youlin Xiang, Yun Xiong, Haofen Wang, and Jiawei Zhang. 2023. Chat-REC: Towards interactive and explainable LLMs-augmented recommender system. arXiv:2303.14524. Retrieved from https://arxiv.org/abs/2303.14524

[26] Shuyu Guo, Shuo Zhang, Weiwei Sun, Pengjie Ren, Zhumin Chen, and Zhaochun Ren. 2023. Towards explainable conversational recommender systems. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '23)*. ACM, New York, NY, 2786–2795. DOI: https://doi.org/10.1145/3539618.3591884

[27] Claudia Hauff, Julia Kiseleva, Mark Sanderson, Hamed Zamani, and Yongfeng Zhang. 2021. Conversational search and recommendation: Introduction to the special issue. *ACM Transactions on Information Systems* 39, 4, Article 38 (Sep. 2021), 6 pages. DOI: https://doi.org/10.1145/3465272

[28] Junhua He, Hankz Hankui Zhuo, and Jarvan Law. 2017. Distributed-representation based hybrid recommender system with short item descriptions. arXiv:1703.04854. Retrieved from https://arxiv.org/abs/1703.04854

[29] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *Proceedings of the 26th International Conference on World Wide Web (WWW '17)*. Rick Barrett, Rick Cummings, Eugene Agichtein, and Evgeniy Gabrilovich (Eds.), ACM, 173–182. DOI: https://doi.org/10.1145/3038912.3052569

[30] Zhankui He, Zhouhang Xie, Rahul Jha, Harald Steck, Dawen Liang, Yesu Feng, Bodhisattwa Prasad Majumder, Nathan Kallus, and Julian McAuley. 2023. Large language models as zero-shot conversational recommenders. In *Proceedings of the 32st ACM International Conference on Information & Knowledge Management*, 720–730.

[31] Chenhao Hu, Shuhua Huang, Yansen Zhang, and Yubao Liu. 2022. Learning to infer user implicit preference in conversational recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '22)*. ACM, New York, NY, 256–266.

[32] Quzhe Huang, Mingxu Tao, Zhenwei An, Chen Zhang, Cong Jiang, Zhibin Chen, Zirui Wu, and Yansong Feng. 2023. Lawyer Llama technical report. arXiv:2305.15062. Retrieved from https://arxiv.org/abs/2305.15062

[33] Xu Huang, Jianxun Lian, Yuxuan Lei, Jing Yao, Defu Lian, and Xing Xie. 2023. Recommender AI agent: Integrating large language models for interactive recommendations. arXiv:2308.16505. Retrieved from https://arxiv.org/abs/2308.16505

[34] Dietmar Jannach, Ahtsham Manzoor, Wanling Cai, and Li Chen. 2021. A survey on conversational recommender systems. *ACM Computing Surveys* 54, 5 (2021). DOI: https://doi.org/10.1145/3453154

[35] Yunjie Ji, Yong Deng, Yan Gong, Yiping Peng, Qiang Niu, Lei Zhang, Baochang Ma, and Xiangang Li. 2023. Exploring the impact of instruction data scaling on large language models: An empirical study on real-world use cases. arXiv:2303.14742. Retrieved from https://arxiv.org/abs/2303.14742

[36] Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2021. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data* 7, 3 (Jul. 2021), 535–547. DOI: https://doi.org/10.1109/TBDATA.2019.2921572

[37] Dongyeop Kang, Anusha Balakrishnan, Pararth Shah, Paul Crook, Y-Lan Boureau, and Jason Weston. 2019. Recommendation as a communication game: Self-supervised bot-play for goal-oriented dialogue. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP '19)*. Association for Computational Linguistics, Hong Kong, China, 1951–1961. DOI: https://doi.org/10.18653/v1/D19-1203

[38] Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP '14)*. Association for Computational Linguistics, Doha, Qatar, 1746–1751. DOI: https://doi.org/10.3115/v1/D14-1181

[39] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR '15).* Yoshua Bengio and Yann LeCun (Eds.). Retrieved from http://arxiv.org/abs/1412.6980

[40] Walid Krichene and Steffen Rendle. 2020. On sampled metrics for item recommendation. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD '20).* ACM, New York, NY, 1748–1757. DOI: https://doi.org/10.1145/3394486.3403226

[41] Wenqiang Lei, Xiangnan He, Yisong Miao, Qingyun Wu, Richang Hong, Min-Yen Kan, and Tat-Seng Chua. 2020. Estimation-action-reflection: Towards deep interaction between conversational and recommender systems. In *Proceedings of the 13th ACM International Conference on Web Search and Data Mining (WSDM '20).* James Caverlee, Xia (Ben) Hu, Mounia Lalmas, and Wei Wang (Eds.), ACM, 304–312. DOI: https://doi.org/10.1145/3336191.3371769

[42] Wenqiang Lei, Gangyi Zhang, Xiangnan He, Yisong Miao, Xiang Wang, Liang Chen, and Tat-Seng Chua. 2020. Interactive path reasoning on graph for conversational recommendation. In *Proceedings of the 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '20).* Rajesh Gupta, Yan Liu, Jiliang Tang, and B. Aditya Prakash (Eds.), ACM, 2073–2083. Retrieved from https://dl.acm.org/doi/10.1145/3394486.3403258

[43] Bohan Li, Yutai Hou, and Wanxiang Che. 2022. Data augmentation approaches in natural language processing: A survey. *AI Open* 3 (2022), 71–90.

[44] Ming Li, Sami Jullien, Mozhdeh Ariannezhad, and Maarten de Rijke. 2023. A next basket recommendation reality check. *ACM Transactions on Information Systems* 41, 4 (2023), 1–29.

[45] Ming Li, Ali Vardasbi, Andrew Yates, and Maarten de Rijke. 2023. Repetition and exploration in sequential recommendation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 2532—2541.

[46] 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: Annual Conference on Neural Information Processing Systems 2018 (NeurIPS '18).* Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett (Eds.), 9748–9758. Retrieved from https://proceedings.neurips.cc/paper/2018/hash/800de15c79c8d840f4e78d3af937d4d4-Abstract.html

[47] Shijun Li, Wenqiang Lei, Qingyun Wu, Xiangnan He, Peng Jiang, and Tat-Seng Chua. 2021. Seamlessly unifying attributes and items: Conversational recommendation for cold-start users. *ACM Transactions on Information Systems* 39, 4, Article 40 (Aug. 2021), 29 pages. DOI: https://doi.org/10.1145/3446427

[48] Shuyang Li, Bodhisattwa Prasad Majumder, and Julian McAuley. 2021. Self-supervised bot play for conversational recommendation with justifications. arXiv:2112.05197. Retrieved from https://arxiv.org/abs/2112.05197

[49] Shuyang Li, Bodhisattwa Prasad Majumder, and Julian McAuley. 2022. Self-supervised bot play for transcript-free conversational recommendation with rationales. In *Proceedings of the 16th ACM Conference on Recommender Systems*, 327–337.

[50] Shuokai Li, Ruobing Xie, Yongchun Zhu, Xiang Ao, Fuzhen Zhuang, and Qing He. 2022. User-centric conversational recommendation with multi-aspect user modeling. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '22)*. ACM, New York, NY, 223–233.

[51] Shuokai Li, Ruobing Xie, Yongchun Zhu, Fuzhen Zhuang, Zhenwei Tang, Wayne Xin Zhao, and Qing He. 2022. Self-supervised learning for conversational recommendation. *Information Processing & Management* 59, 6 (2022), 103067.

[52] Lizi Liao, Ryuichi Takanobu, Yunshan Ma, Xun Yang, Minlie Huang, and Tat-Seng Chua. 2019. Deep conversational recommender in travel. arXiv:1907.00710. Retrieved from https://arxiv.org/abs/1907.00710

[53] Allen Lin, Ziwei Zhu, Jianling Wang, and James Caverlee. 2023. Enhancing user personalization in conversational recommenders. In *Proceedings of the ACM Web Conference 2023*, 770–778.

[54] Dongding Lin, Jian Wang, and Wenjie Li. 2023. COLA: Improving conversational recommender systems by collaborative augmentation. In *Proceedings of the 37th AAAI Conference on Artificial Intelligence,* 4462–4470.

[55] Junling Liu, Chao Liu, Renjie Lv, Kang Zhou, and Yan Zhang. 2023. Is ChatGPT a good recommender? A preliminary study. arXiv:2304.10149. Retrieved from https://arxiv.org/abs/2304.10149

[56] Tiedong Liu and Bryan Kian Hsiang Low. 2023. Goat: Fine-tuned LLaMA outperforms GPT-4 on arithmetic tasks. arXiv:2305.14201. Retrieved from https://arxiv.org/abs/2305.14201

[57] Yuanxing Liu, Zhaochun Ren, Wei-Nan Zhang, Wanxiang Che, Ting Liu, and Dawei Yin. 2020. Keywords generation improves E-commerce session-based recommendation. In *Proceedings of the Web Conference 2020 (WWW '20).* Yennun Huang, Irwin King, Tie-Yan Liu, and Maarten van Steen (Eds.), ACM/IW3C2, 1604–1614. DOI: https://doi.org/10.1145/3366423.3380232

[58] Yuanxing Liu, Weinan Zhang, Baohua Dong, Yan Fan, Hang Wang, Fan Feng, Yifan Chen, Ziyu Zhuang, Hengbin Cui, Yongbin Li, et al. 2023. U-NEED: A fine-grained dataset for user needs-centric E-commerce conversational recommendation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval,* 2723–2732.

[59] Yu Lu, Junwei Bao, Yan Song, Zichen Ma, Shuguang Cui, Youzheng Wu, and Xiaodong He. 2021. RevCore: Review-augmented conversational recommendation. In *Findings of the Association for Computational Linguistics (ACL-IJCNLP '21)*. Association for Computational Linguistics, 1161–1173. DOI: https://doi.org/10.18653/v1/2021.findings-acl.99

[60] Wenchang Ma, Ryuichi Takanobu, and Minlie Huang. 2021. CR-Walker: Tree-structured graph reasoning and dialog acts for conversational recommendation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 1839–1851. DOI: https://doi.org/10.18653/v1/2021.emnlp-main.139

[61] Ida Mele, Cristina Ioana Muntean, Franco Maria Nardini, Raffaele Perego, Nicola Tonellotto, and Ophir Frieder. 2021. Adaptive utterance rewriting for conversational search. *Information Processing & Management* 58, 6 (2021), 102682.

[62] Yuxin Ni, Yunwen Xia, Hui Fang, Chong Long, Xinyu Kong, Daqian Li, Dong Yang, and Jie Zhang. 2023. Meta-CRS: A dynamic meta-learning approach for effective conversational recommender system. *ACM Transactions on Information Systems* 42, 1, Article 28 (Aug. 2023), 27 pages. DOI: https://doi.org/10.1145/3604804

[63] OpenAI. 2022. Introducing ChatGPT. Retrieved from https://openai.com/blog/chatgpt

[64] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems,* Vol. 35, 27730–27744.

[65] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019 (NeurIPS '19)*. Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett (Eds.), 8024–8035. Retrieved from https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html

[66] Jiahuan Pei. 2022. *Collaborative Agents for Task-Oriented Dialogue Systems*. Ph.D. Dissertation. University of Amsterdam.

[67] Jiahuan Pei, Pengjie Ren, and Maarten de Rijke. 2021. A cooperative memory network for personalized task-oriented dialogue systems with incomplete user profiles. In *Proceedings of the Web Conference 2021.* ACM, 1552–1561.

[68] Dhanya Pramod and Prafulla Bafna. 2022. Conversational recommender systems techniques, tools, acceptance, and adoption: A state of the art review. *Expert Systems with Applications* 203, C (2022).

[69] Arpit Rana, Scott Sanner, Mohamed Reda Bouadjenek, Ron Dicarlantonio, and Gary Farmaner. 2023. User experience and the role of personalization in critiquing-based conversational recommendation. *ACM Transactions on the Web* 18, 4, Article 43 (May 2023), 1–21. DOI: https://doi.org/10.1145/3597499

[70] Xuhui Ren, Tong Chen, Quoc Viet Hung Nguyen, Lizhen Cui, Zi Huang, and Hongzhi Yin. 2023. Explicit knowledge graph reasoning for conversational recommendation. *ACM Transactions on Intelligent Systems and Technology* 15, 4, Article 86 (Dec. 2023), 1–21. DOI: https://doi.org/10.1145/3637216

[71] Xuhui Ren, Hongzhi Yin, Tong Chen, Hao Wang, Nguyen Quoc Viet Hung, Zi Huang, and Xiangliang Zhang. 2020. CRSAL: Conversational recommender systems with adversarial learning. *ACM Transactions on Information Systems* 38, 4, Article 34 (Jun. 2020), 40 pages. DOI: https://doi.org/10.1145/3394592

[72] Zhaochun Ren, Zhi Tian, Dongdong Li, Pengjie Ren, Liu Yang, Xin Xin, Huasheng Liang, Maarten de Rijke, and Zhumin Chen. 2022. Variational reasoning about user preferences for conversational recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '22)*. ACM, New York, NY, 165–175.

[73] Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, et al. 2022. Bloom: A 176b-parameter open-access multilingual language model. arXiv:2211.05100. Retrieved from https://arxiv.org/abs/2211.05100

[74] Iulian Vlad Serban, Alessandro Sordoni, Ryan Lowe, Laurent Charlin, Joelle Pineau, Aaron C. Courville, and Yoshua Bengio. 2017. A hierarchical latent variable encoder-decoder model for generating dialogues. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence.* Satinder P. Singh and Shaul Markovitch (Eds.), AAAI Press, 3295–3301. Retrieved from http://aaai.org/ocs/index.php/AAAI/AAAI17/paper/view/14567

[75] Tianshu Shen, Jiaru Li, Mohamed Reda Bouadjenek, Zheda Mai, and Scott Sanner. 2023. Towards understanding and mitigating unintended biases in language model-driven conversational recommendation. *Information Processing & Management* 60, 1 (2023), 103139.

[76] Clemencia Siro, Mohammad Aliannejadi, and Maarten de Rijke. 2022. Understanding user satisfaction with task-oriented dialogue systems. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '22)*. ACM, 2018–2023.

[77] Clemencia Siro, Mohammad Aliannejadi, and Maarten de Rijke. 2023. Understanding and predicting user satisfaction with conversational recommender systems. *ACM Transactions on Information Systems* 42, 2, Article 55 (Nov. 2023), 37 pages. DOI: https://doi.org/10.1145/3624989

[78] Joo-yeong Song and Bongwon Suh. 2022. Data augmentation strategies for improving sequential recommender systems. arXiv:2203.14037. Retrieved from https://arxiv.org/abs/2203.14037

[79] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research* 15, 1 (Jan. 2014), 1929–1958.

[80] Lovis Bero Suchmann, Nicole Krämer, and Jürgen Ziegler. 2023. Branching preferences: Visualizing non-linear topic progression in conversational recommender systems. In *Adjunct Proceedings of the 31st ACM Conference on User Modeling, Adaptation and Personalization (UMAP '23 Adjunct)*. ACM, New York, NY, 199–205. DOI: https://doi.org/10.1145/3563359.3597380

[81] Weiwei Sun, Lingyong Yan, Xinyu Ma, Pengjie Ren, Dawei Yin, and Zhaochun Ren. 2023. Is ChatGPT good at search? Investigating large language models as re-ranking agent. arXiv:2304.09542. Retrieved from https://arxiv.org/abs/2304.09542

[82] Yueming Sun and Yi Zhang. 2018. Conversational recommender system. In *Proceedings of the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR 2018)*. Kevyn Collins-Thompson, Qiaozhu Mei, Brian D. Davison, Yiqun Liu, and Emine Yilmaz (Eds.), ACM, 235–244. DOI: https://doi.org/10.1145/3209978.3210002

[83] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal azhar, et al. 2023. Llama: Open and efficient foundation language models. arXiv:2302.13971. Retrieved from https://arxiv.org/abs/2302.13971

[84] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, and Shruti Bhosale. 2023. Llama 2: Open foundation and fine-tuned chat models. arXiv:2307.09288. Retrieved from https://arxiv.org/abs/2307.09288

[85] Johanne R. Trippas, Damiano Spina, Paul Thomas, Mark Sanderson, Hideo Joho, and Lawrence Cavedon. 2020. Towards a model for spoken conversational search. *Information Processing & Management* 57, 2 (2020), 102162.

[86] Manos Tsagkias, Tracy Holloway King, Surya Kallumadi, Vanessa Murdock, and Maarten de Rijke. 2020. Challenges and research opportunities in eCommerce search and recommendations. *SIGIR Forum* 54, 1 (2020), 1–23.

[87] Quan Tu, Shen Gao, Yanran Li, Jianwei Cui, Bin Wang, and Rui Yan. 2022. Conversational recommendation via hierarchical information modeling. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '22)*. ACM, New York, NY, 2201–2205. DOI: https://doi.org/10.1145/3477495.3531830

[88] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017*. Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (Eds.), 5998–6008. Retrieved from https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html

[89] Benyou Wang, Qianqian Xie, Jiahuan Pei, Prayag Tiwari, Zhao Li, and Jie Fu. 2021. Pre-trained language models in biomedical domain: A systematic survey. arXiv:2110.05006. Retrieved from https://arxiv.org/abs/2110.05006

[90] Haochun Wang, Chi Liu, Nuwa Xi, Zewen Qiang, Sendong Zhao, Bing Qin, and Ting Liu. 2023. HuaTuo: Tuning llama model with Chinese medical knowledge. arXiv:2304.06975. Retrieved from https://arxiv.org/abs/2304.06975

[91] Ting-Chun Wang, Shang-Yu Su, and Yun-Nung Chen. 2022. BARCOR: Towards a unified framework for conversational recommendation systems. arXiv:2203.14257). Retrieved from https://arxiv.org/abs/2203.14257

[92] Xiaolei Wang, Xinyu Tang, Xin Zhao, Jingyuan Wang, and Ji-Rong Wen. 2023. Rethinking the evaluation for conversational recommendation in the era of large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. Houda Bouamor, Juan Pino, and Kalika Bali (Eds.), Association for Computational Linguistics, 10052–10065. DOI: https://doi.org/10.18653/v1/2023.emnlp-main.621

[93] Xiaolei Wang, Kun Zhou, Xinyu Tang, Wayne Xin Zhao, Fan Pan, Zhao Cao, and Ji-Rong Wen. 2023. Improving conversational recommendation systems via counterfactual data simulation. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '23)*. ACM, New York, NY, 2398–2408. DOI: https://doi.org/10.1145/3580305.3599387

[94] Xiaolei Wang, Kun Zhou, Ji-Rong Wen, and Wayne Xin Zhao. 2022. Towards unified conversational recommender systems via knowledge-enhanced prompt learning. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 1929–1937.

[95] Yingxu Wang, Xiaoru Chen, Jinyuan Fang, Zaiqiao Meng, and Shangsong Liang. 2023. Enhancing conversational recommendation systems with representation fusion. *ACM Transactions on the Web* 17, 1, Article 6 (Feb. 2023), 34 pages. DOI: https://doi.org/10.1145/3577034

[96] Chi-Man Wong, Fan Feng, Wen Zhang, Chi-Man Vong, Hui Chen, Yichi Zhang, Peng He, Huan Chen, Kun Zhao, and Huajun Chen. 2021. Improving conversational recommender system by pretraining billion-scale knowledge graph. In *Proceedings of the 2021 IEEE 37th International Conference on Data Engineering (ICDE '21)*, 2607–2612.

[97] Hui Wu, Yupeng Gao, Xiaoxiao Guo, Ziad Al-Halah, Steven Rennie, Kristen Grauman, and Rogerio Feris. 2021. Fashion IQ: A new dataset towards retrieving images by natural language feedback. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11307–11317.

[98] Jiajin Wu, Bo Yang, Dongsheng Li, and Lihui Deng. 2023. A semantic relation-aware deep neural network model for end-to-end conversational recommendation. *Applied Soft Computing* 132 (2023), 109873. DOI: https://doi.org/10.1016/j.asoc.2022.109873

[99] Guojun Yan, Jiahuan Pei, Pengjie Ren, Zhaochun Ren, Xin Xin, Huasheng Liang, Maarten de Rijke, and Zhumin Chen. 2022. ReMeDi: Resources for multi-domain, multi-service, medical dialogues. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '22)*. ACM, New York, NY, 3013–3024.

[100] Hyeongjun Yang, Heesoo Won, Youbin Ahn, and Kyong-Ho Lee. 2023. CLICK: Contrastive learning for injecting contextual knowledge to conversational recommender system. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*. Andreas Vlachos and Isabelle Augenstein (Eds.), Association for Computational Linguistics, 1875–1885. DOI: https://doi.org/10.18653/v1/2023.eacl-main.137

[101] Yao-Chun Yang, Chiao-Ting Chen, Tzu-Yu Lu, and Szu-Hao Huang. 2023. Hierarchical reinforcement learning for conversational recommendation with knowledge graph reasoning and heterogeneous questions. *IEEE Transactions on Services Computing* 16, 5 (2023), 3439–3452. DOI: https://doi.org/10.1109/TSC.2023.3269396

[102] Huayong Yao, Hongyu Yao, and Dongyi Ye. 2023. DHGECON: A multi-round conversational recommendation method based on dynamic heterogeneous encoding. *Knowledge-Based Systems* 273 (2023), 110607. DOI: https://doi.org/10.1016/j.knosys.2023.110607

[103] Dianer Yu, Qian Li, Xiangmeng Wang, Qing Li, and Guandong Xu. 2023. Counterfactual explainable conversational recommendation. *IEEE Transactions on Knowledge and Data Engineering* (2023), 1–13. DOI: https://doi.org/10.1109/TKDE.2023.3322403

[104] Jingsi Yu, Junhui Zhu, Yujie Wang, Yang Liu, Hongxiang Chang, Jinran Nie, Cunliang Kong, Ruining Cong, Xin Liu, Jiyuan An, et al. 2023. Taoli Llama. Retrieved from https://github.com/blcuicall/taoli

[105] YangMu Yu. 2023. Cornucopia-LLaMA-Fin-Chinese. Retrieved from https://github.com/jerry1993-tech/Cornucopia-LLaMA-Fin-Chinese

[106] Xingshan Zeng, Jing Li, Lingzhi Wang, and Kam-Fai Wong. 2021. Modeling global and local interactions for online conversation recommendation. *ACM Transactions on Information Systems* 40, 3, Article 49 (Nov. 2021), 33 pages. DOI: https://doi.org/10.1145/3473970

[107] Chengyang Zhang, Xianying Huang, and Jiahao An. 2023. MACR: Multi-information augmented conversational recommender. *Expert Systems with Applications* 213 (2023), 118981.

[108] Tong Zhang, Yong Liu, Boyang Li, Peixiang Zhong, Chen Zhang, Hao Wang, and Chunyan Miao. 2022. Toward knowledge-enriched conversational recommendation systems. In *Proceedings of the 4th Workshop on NLP for Conversational AI*. Association for Computational Linguistics, 212–217. DOI: https://doi.org/10.18653/v1/2022.nlp4convai-1.17

[109] Xiaoying Zhang, Hong Xie, Hang Li, and John C. S. Lui. 2020. Conversational contextual bandit: Algorithm and application. In *Proceedings of the Web Conference 2020 (WWW '20)*. Yennun Huang, Irwin King, Tie-Yan Liu, and Maarten van Steen (Eds.), ACM /IW3C2, 662–672. DOI: https://doi.org/10.1145/3366423.3380148

[110] Xiaoyu Zhang, Xin Xin, Dongdong Li, Wenxuan Liu, Pengjie Ren, Zhumin Chen, Jun Ma, and Zhaochun Ren. 2023. Variational reasoning over incomplete knowledge graphs for conversational recommendation. In *Proceedings of the 16th ACM International Conference on Web Search and Data Mining*, 231–239.

[111] Yongfeng Zhang, Xu Chen, Qingyao Ai, Liu Yang, and W. Bruce Croft. 2018. Towards conversational search and recommendation: System ask, user respond. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management (CIKM '18)*. Alfredo Cuzzocrea, James Allan, Norman W. Paton, Divesh Srivastava, Rakesh Agrawal, Andrei Z. Broder, Mohammed J. Zaki, K. Selçuk Candan, Alexandros Labrinidis, Assaf Schuster, and Haixun Wang (Eds.), ACM, 177–186. DOI: https://doi.org/10.1145/3269206.3271776

[112] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. 2023. A survey of large language models. arXiv:2303.18223. Retrieved from https://arxiv.org/abs/2303.18223

[113] Jinfeng Zhou, Bo Wang, Ruifang He, and Yuexian Hou. 2021. CRFR: Improving conversational recommender systems via flexible fragments reasoning on knowledge graphs. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 4324–4334. DOI: https://doi.org/10.18653/v1/2021.emnlp-main.355

[114] Kun Zhou, Hui Wang, Wayne Xin Zhao, Yutao Zhu, Sirui Wang, Fuzheng Zhang, Zhongyuan Wang, and Ji-Rong Wen. 2020. S3-rec: Self-supervised learning for sequential recommendation with mutual information maximization. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management (CIKM '20)*. Mathieu d'Aquin, Stefan Dietze, Claudia Hauff, Edward Curry, and Philippe Cudré-Mauroux (Eds.), ACM, 1893–1902. DOI: https://doi.org/10.1145/3340531.3411954

[115] Kun Zhou, Xiaolei Wang, Yuanhang Zhou, Chenzhan Shang, Yuan Cheng, Wayne Xin Zhao, Yaliang Li, and Ji-Rong Wen. 2021. CRSLab: An open-source toolkit for building conversational recommender system. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*. Association for Computational Linguistics, 185–193. DOI: https://doi.org/10.18653/v1/2021.acl-demo.22

[116] Kun Zhou, Wayne Xin Zhao, Shuqing Bian, Yuanhang Zhou, Ji-Rong Wen, and Jingsong Yu. 2020. Improving conversational recommender systems via knowledge graph based semantic fusion. In *Proceedings of the 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '20)*. Rajesh Gupta, Yan Liu, Jiliang Tang, and B. Aditya Prakash (Eds.), ACM, 1006–1014. Retrieved from https://dl.acm.org/doi/10.1145/3394486.3403143

[117] Kun Zhou, Yuanhang Zhou, Wayne Xin Zhao, Xiaoke Wang, and Ji-Rong Wen. 2020. Towards topic-guided conversational recommender system. In *Proceedings of the 28th International Conference on Computational Linguistics*. International Committee on Computational Linguistics, 4128–4139. DOI: https://doi.org/10.18653/v1/2020.coling-main.365

[118] Yuanhang Zhou, Kun Zhou, Wayne Xin Zhao, Cheng Wang, Peng Jiang, and He Hu. 2022. C^2-CRS: Coarse-to-fine contrastive learning for conversational recommender system. In *Proceedings of the 15th ACM International Conference on Web Search and Data Mining*.

[119] Yutao Zhu, Huaying Yuan, Shuting Wang, Jiongnan Liu, Wenhan Liu, Chenlong Deng, Zhicheng Dou, and Ji-Rong Wen. 2023. Large language models for information retrieval: A survey. arXiv:2308.07107. Retrieved from https://arxiv.org/abs/2308.07107

[120] Jie Zou, Yifan Chen, and Evangelos Kanoulas. 2020. Towards question-based recommender systems. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '20)*. Jimmy Huang, Yi Chang, Xueqi Cheng, Jaap Kamps, Vanessa Murdock, Ji-Rong Wen, and Yiqun Liu (Eds.), ACM, 881–890. DOI: https://doi.org/10.1145/3397271.3401180

[121] Jie Zou, Evangelos Kanoulas, Pengjie Ren, Zhaochun Ren, Aixin Sun, and Cheng Long. 2022. Improving conversational recommender systems via transformer-based sequential modelling. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '22)*. ACM, New York, NY, 2319–2324. DOI: https://doi.org/10.1145/3477495.3531852

Appendix

A Notation

Table A1 lists the notation used in the article.

Table A1. Main Notation Used in the Article

Symbol	Description
D	A labeled dataset
D^p	A pseudo-labeled dataset
X	A conversational context
Y	Set of candidate items
Z	Response
V	Set of vocabulary
$\mathcal{N}^x$	A set of user representations of neighboring conversations
$\mathcal{N}^y$	A set of neighboring items
y	An item in Y
w	A word in V
m	Number of neighboring conversations
n	Number of neighboring items
b	User preference over the candidate item y
b'	An auxiliary user preference
b^s	Entity-based preference
$T_{i,j}$	The frequency with which the jth entity is mentioned in the ith conversation
s_j	The cosine similarity score of entity e_j and item y
s_i^r	The score of recommending an item with considering its neighboring items
s^r	The score of recommending an item without considering its neighboring items
w_i	A learnable weight of the correlation feature $\boldsymbol{v}_i$
λ_α	Weight for ranking loss $\mathcal{L}_\alpha$
λ_β	Weight for cosine embedding loss $\mathcal{L}_\beta$
ϕ	A function to seek neighboring conversations
ψ	A function to seek neighboring items
τ_i	The frequency with which the item is mentioned in the ith conversation
α	Margin value of the distance between s^r and s_i^r for negative pairs
β	Margin value of the distance between an item and its candidate neighboring item
$\boldsymbol{x}$	User representation
$\boldsymbol{y}$	The embedding of candidate item y
$\boldsymbol{w}$	The embedding of word w
$\boldsymbol{x}'$	An auxiliary user representation
$\boldsymbol{v}_i$	The correlation feature of ith neighboring conversation
$\boldsymbol{y}^e$	The representation of 10 highest-scoring entities
$\boldsymbol{\mathcal{X}}$	An index for all user representations in the training set
$\boldsymbol{\mathcal{Y}}$	An index for all candidate items
p_i^r	The probability of recommending an item y to a user
p_j^g	The probability of generating the jth word w_j in the response Z
$\mathcal{L}_r$	Loss of item recommendation
$\mathcal{L}_g$	Loss of response generation
$\mathcal{L}_\alpha$	Neighboring item calibrated ranking loss
$\mathcal{L}_\beta$	Neighboring item calibrated cosine embedding loss

B Prompts for Llama 2 and Chinese-Alpaca2

B.1 Prompt Template for Item Recommendation

In practice, Llama 2 tends to refuse to recommend movies or recommends spurious movies. To make Llama 2 return 50 movies from a list of candidates, we adopt a prompt template with four segments: (i) task description, (ii) candidate movies, (iii) conversation history, and (iv) output instruction. An example is given below, along with the response generated by Llama 2.

Task description. Recommend 50 non-repeating movies based on the given conversation history and candidate movies. Please note that your recommendations must be limited to the given 100 candidate movies. Do not recommend other movies.

Candidate movies. The candidate movies are: [A Woman Scorned (1911), The Nightmare (2015), Planet Earth (1974), The Man Who Knew Too Little (1997), Dirty Dancing: Havana Nights, The Rocky Horror Picture Show (1975), Wild (2016), Amores perros (2000), Tickled (2016), The Bling Ring (2013), Still Waters (2000), The Air I Breathe (2007), Shaft (2000), Letters from Iwo Jima (2006), Going in Style (2017), Return of the Killer Tomatoes (1988), The Incident (1967), Percy Jackson & the Olympians: The Lightning Thief, The Zookeeper's Wife (2017), The Demons (2015), Aliens vs. Predator: Requiem (2007), Forgetting Sarah Marshall (2008), Beauty and the Beast (1991), The Shadow (1994), The Driver (1978), Joe Versus the Volcano (1990), Dredd (2012), The Madness of King George (1994), My Little Pony: Equestria Girls – Rainbow Rocks (2014), Star Trek Into Darkness (2013), Murder by Death (1976), Beverly Hills Cop III (1994), Notting Hill , Striptease (1996), Major League (1989), Date Night (2010), Serenity (2005), Airport 1975 (1974), Daddy Day Camp (2007), Kite (2014), The Da Vinci Code (2006), Star Trek: Nemesis (2002), Holy Rollers (2010), The Naked Gun, Bright Angel (1990), Who's Afraid of Virginia Woolf%3F (1966), The Adventures of Pinocchio (1996), Peggy Sue Got Married (1986), High Fidelity (2000), The Rite (2011), I Am Number Four (2011), Prancer (1989), Sesame Street Presents Follow That Bird (1985), Bim (1974), Midnight Cowboy (1969), Mother! (2017), Ontmaskerd (1915), Nightbreed (1990), All Dogs Go to Heaven 2 (1996), 48 Hrs. (1982), Paddington Bear , The Nativity Story, Why Me%3F (2015), Olaf's Frozen Adventure (2017), She's the One (2013), Lost in the Wild (1993), The Enemy (1979), Passenger 57 (1992), Barbershop , Drumline (2002), My Dinner with Andre (1981), Love & Other Drugs (2010), Fort Apache (1948), The Kid (1999), The Thomas Crown Affair (1999), In Her Shoes (2005), The Last Exorcism Part II (2013), Withnail and I (1987), A Charlie Brown Christmas (1965), The Last Temptation of Christ (1988), Star Wars (1977), The Holiday (2006), Holmes & Watson. Madrid Days (2012), The Raven (2012), James and the Giant Peach (1996), High Sierra (1941), Purgatory (2007), Bridesmaids (1989), A Madea Christmas (2013), Cromwell (1970), Secretary (2002), The Haunted World of El Superbeasto (2009), Lego Indiana Jones and the Raiders of the Lost Brick, Fracture (2007), City on Fire (1979), We Need to Talk About Kevin (2011), Chocolat (2000), Wish Upon (2017), Lights Out (2016), Wings of Life (2011)].

Conversation history. The conversation history is: Seeker: Hi I am looking for a movie like Super Troopers (2001) [SEP] Recommender: You should watch Police Academy (1984) [SEP] Seeker: Is that a great one? I have never seen it. I have seen American Pie [SEP] Seeker: I mean American Pie (1999) [SEP] Recommender: Yes Police Academy (1984) is very funny and so is Police Academy 2: Their First Assignment (1985) [SEP] Seeker: It sounds like I need to check them out [SEP] Recommender: yes you will enjoy them [SEP] Seeker: I appreciate your time. I will need to check those out. Are there any others you would recommend? [SEP] Recommender: yes Lethal Weapon (1987) [SEP] Seeker: Thank you i will watch that too [SEP] Recommender: and also Beverly Hills Cop (1984) [SEP] Seeker: Thanks for the suggestions. [SEP] Recommender: you are welcome.

Output instruction. Now recommend 50 non-repeating movies. Just give me the title of the movie being recommended. No introduction to the movie is needed. Output the recommendations into a Python list.

Llama2 response. Sure! Based on the conversation history provided, here are 50 non-repeating movie recommendations:
1. Police Academy (1984)
2. American Pie (1999)
3. Lethal Weapon (1987)
4. Beverly Hills Cop (1984)
5. The Nightmare (2015)
6. The Man Who Knew Too Little (1997)
7. Dirty Dancing: Havana Nights (2003)
8. The Rocky Horror Picture Show (1975)
9. Wild (2016)
10. Amores perros (2000)
11. Tickled (2016)
12. The Bling Ring (2013)
13. Still Waters (2000)
14. The Air I Breathe (2007)
15. Shaft (2000)
16. Letters from Iwo Jima (2006)
17. Going in Style (2017)
18. Return of the Killer Tomatoes (1988)
19. The Incident (1967)
20. Percy Jackson & the Olympians: The Lightning Thief (2010)
21. The Zookeeper's Wife (2017)
22. The Demons (2015)
23. Aliens vs. Predator: Requiem (2007)
24. Forgetting Sarah Marshall (2008)
25. Beauty and the Beast (1991)
26. The Shadow (1994)
27. The Driver (1978)
28. Joe Versus the Volcano (1990)
29. Dredd (2012)
30. The Madness of King George (1994)
31. My Little Pony: Equestria Girls – Rainbow Rocks (2014)
32. Star Trek Into Darkness (2013)
33. Murder by Death (1976)

34. Beverly Hills Cop III (1994)
35. Notting Hill (1999)
36. Striptease (1996)
37. Major League (1989)
38. Date Night (2010)
39. Serenity (2005)
40. Airport 1975 (1974)
41. Daddy Day Camp (2007)
42. Kite (2014)
43. The Da Vinci Code (2006)
44. Star Trek: Nemesis (2002)
45. Holy Rollers (2010)
46. The Naked Gun, Bright Angel (1990)
47. Who's Afraid of Virginia Woolf%3F (1966)
48. The Adventures of Pinocchio (1996)
49. Peggy Sue Got Married (1986)
50. High Fidelity (2000)

For the U-NEED dataset, we supplement the prompt template with attributes and values of candidate items. We do this because LLMs do not have information about the items, i.e., products, in the U-NEED dataset. An example is given below, along with the response generated by Chinese-Alpaca2.

Task description. [INST] ⟨⟨SYS⟩⟩ 你是一个电商领域的售前导购客服。请你提供专业、有逻辑、内容真实、有价值的详细回复。⟨⟨/SYS⟩⟩ **{Candidate items} {Conversation history} {Output instruction}** [/INST]

Candidate items. 下面你可以参考的商品知识：商品 1 的价格区间是高，适用人群是中年人、中老年人、妈妈、家人、朋友、男朋友、老年人，功效是保湿、去眼袋、嫩滑肌肤、抗皱、控油、收缩毛孔、改善气色、淡纹去皱、祛斑、祛痘、紧致提拉、美白、营养滋润，适用年龄是 10 岁以下、20-30 岁、30-40 岁、40-50 岁、50 岁以上，适用性别是女，类目是面部护理套装，改善肌肤问题是斑、日晒斑、暗沉、松弛下垂、毛孔粗大、痘印、痘痘、皱纹细纹、眼纹、眼袋、粗糙、红血丝、老年斑、色斑、起皮、闭口、黑头、黑眼圈，适用肤色是黄，适用肤质是中性、干性、敏感、油性、混合 [SEP] 商品 2 的价格区间是高，功效是保湿、去鸡皮肤、易吸收，类目是身体乳，改善肌肤问题是鸡皮，适用肤质是干性 [SEP] 商品 3 的价格区间是高，适用人群是中年人、男朋友、老年人，功效是保湿、抗皱、改善气色、祛斑、紧致提拉、美白，适用年龄是 30-40 岁、50 岁以上，适用性别是女，类目是乳液/面霜，改善肌肤问题是斑、暗沉、松弛下垂、毛孔粗大、痘痘、皱纹细纹、红血丝、色斑、起皮、闭口、黑头，适用肤质是中性、干性、敏感、油性、混合 [SEP] 商品 4 的价格区间是低，适用性别是女，类目是乳液/面霜，适用肤质是干性、混合 [SEP] 商品 5 的价格区间是中，适用人群是儿童、妈妈，功效是保湿，适用性别是女，类目是乳液/面霜，改善肌肤问题是皱纹细纹，适用肤质是中性、干性、敏感、油性、混合 [SEP] 商品 6 的价格区间是低 [SEP] 商品 7 的价格区间是高，适用人群是妈妈，使用场景是日常，功效是保湿、抗皱、收缩毛孔、改善气色、淡纹去皱、祛斑、紧致提拉、美白、防晒，适用年龄是 20-30 岁、28-35 岁、30-40 岁，类目是面部护理套装，改善肌肤问题是斑、日晒斑、暗沉、松弛

下垂、毛孔粗大、痘印、痘痘、痤疮、皱纹细纹、眼纹、眼袋、粉刺、粗糙、红血丝、老年斑、脂肪粒、色斑、起皮、闭口、黑头、黑眼圈，适用肤质是中性、干性、敏感、油性、混合 [SEP] 商品 8 的价格区间是高，类目是面部护理套装，改善肌肤问题是暗沉、色斑，适用肤质是干性 [SEP] 商品 9 的价格区间是高，类目是洁面，适用肤质是中性、干性、敏感、混合 [SEP] 商品 10 的价格区间是低，功效是保湿，适用年龄是 20-30 岁、30-40 岁，适用性别是女、男，类目是面部护理套装，改善肌肤问题是暗沉、痘痘、皱纹细纹，适用肤质是中性、干性、敏感、油性、混合 [SEP] 商品 11 的价格区间是中，适用人群是老年人，功效是保湿，适用性别是女，类目是面部护理套装，改善肌肤问题是斑、暗沉、皱纹细纹，适用肤质是中性、干性、敏感、油性、混合 [SEP] 商品 12 的价格区间是低 [SEP] 商品 13 的价格区间是高，功效是保湿，类目是面膜，改善肌肤问题是毛孔粗大、粉刺、粗糙、黑头，适用肤质是油性 [SEP] 商品 14 的价格区间是中，适用人群是中年人、儿童、妈妈、家人，适用季节是春夏、秋冬，功效是保湿、抗皱、抗过敏、控油、收缩毛孔、改善气色、祛斑、祛痘、紧致提拉、美白、防晒，适用年龄是 10 岁以下、30-40 岁、50 岁以上，适用性别是女、男，类目是面部护理套装，改善肌肤问题是斑、日晒斑、暗沉、松弛下垂、毛孔粗大、痘印、痘痘、痤疮、白头、皱纹细纹、粉刺、粗糙、红血丝、脂肪粒、脓包、色斑、起皮、闭口、鸡皮、黑头、黑眼圈，适用肤质是中性、干性、敏感、油性、混合 [SEP] 商品 15 的价格区间是中，使用场景是日常，类目是防晒霜 [SEP] 商品 16 的价格区间是高，适用人群是妈妈、老年人，功效是保湿、抗皱、祛斑、紧致提拉，适用年龄是 10 岁以下、30-40 岁、40-50 岁、50 岁以上，适用性别是女，类目是乳液/面霜，改善肌肤问题是斑、暗沉、松弛下垂、毛孔粗大、痘痘、皱纹细纹、眼袋、粗糙、色斑、黑头，适用肤质是中性、干性、敏感、油性、混合 [SEP] 商品 17 的价格区间是高，功效是保湿、抗皱、紧致提拉、美白，类目是乳液/面霜，改善肌肤问题是暗沉、皱纹细纹、起皮，适用肤质是干性、敏感、混合 [SEP] 商品 18 的价格区间是高 [SEP] 商品 19 的价格区间是高，功效是保湿、嫩滑肌肤、抗皱、收缩毛孔、改善气色、淡纹去皱、祛斑、紧致提拉、美白，适用年龄是 50 岁以上，类目是面膜，改善肌肤问题是斑、暗沉、松弛下垂、毛孔粗大、痘痘、皱纹细纹、眼纹、红血丝、色斑、黑头、黑色素，适用肤质是中性、干性、敏感、油性、混合 [SEP] 商品 20 的价格区间是中，功效是保湿，类目是乳液/面霜 [SEP] 商品 21 的价格区间是低 [SEP] 商品 22 的价格区间是高，功效是保湿、抗皱、紧致提拉，适用性别是女，类目是化妆水/爽肤水，改善肌肤问题是斑、暗沉、松弛下垂、毛孔粗大、痘痘、皱纹细纹、色斑、起皮、黑头，适用肤质是中性、干性、敏感、油性、混合 [SEP] 商品 23 的价格区间是高，功效是保湿，类目是乳液/面霜，适用肤质是中性、干性、敏感、混合 [SEP] 商品 24 的价格区间是低 [SEP] 商品 25 的价格区间是高，功效是祛斑，类目是洁面，改善肌肤问题是斑、暗沉、皱纹细纹、色斑、黑眼圈，适用肤质是中性、干性、混合 [SEP] 商品 26 的价格区间是中，功效是保湿，适用性别是男，类目是面部护理套装，改善肌肤问题是斑、暗沉、松弛下垂、毛孔粗大、痘痘、皱纹细纹、粉刺、黑头，适用肤质是中性、干性、油性、混合 [SEP] 商品 27 的价格区间是中，功效是改善气色，适用性别是男，类目是面部护理套装，改善肌肤问题是暗沉、皱纹细纹、起皮，适用肤质是中性、干性、敏感、油性、混合 [SEP] 商品 28 的价格区间是高，类目是护手霜，改善肌肤问题是暗沉、皱纹细纹、色斑，适用肤质是中性、干性、油性、混合 [SEP] 商品 29 的价格区间是低 [SEP] 商品 30 的价格区间是中，类目是乳液/面霜，改善肌肤问题是毛孔粗大，适用肤质是敏感、油性、混合 [SEP] 商品 31 的价格区间是中，适用人群是妈妈、家人，功效是保湿、修复调理、去眼袋、抗皱、控油、改善气色、祛斑、祛痘、紧致提拉、美白、营养滋润，适用年龄是 10-20 岁、10 岁以下、20-30 岁、30-40 岁、50 岁以上，适用性别是女、男，类目是面部护理套装，改善肌肤问题是斑、日晒斑、晒后、暗沉、松弛下垂、毛孔粗大、痘印、痘痘、皱纹细纹、眼纹、眼袋、粉刺、粗糙、红血丝、老年斑、色斑、起皮、闭口、雀斑、鸡皮、黑头、黑眼圈、黑色素，适用肤色是黄、黑，适用肤质是中性、干性、敏感、油性、混合 [SEP] 商品 32 的价格区间是低 [SEP] 商品 33 的价格区间是高，类目是乳液/面霜，改善肌肤问题是皱纹细纹，适用肤质是混合 [SEP] 商品 34

的价格区间是低 [SEP] 商品 35 的价格区间是高 [SEP] 商品 36 的价格区间是高，功效是保湿、抗皱、改善气色、祛斑、紧致提拉、美白，适用年龄是 30-40 岁、40-50 岁，类目是精华，改善肌肤问题是斑、暗沉、松弛下垂、毛孔粗大、痘印、痘痘、皱纹细纹、眼纹、眼袋、粉刺、色斑、起皮、黑头、黑眼圈、黑色素，适用肤质是中性、干性、敏感、油性、混合 [SEP] 商品 37 的价格区间是高 [SEP] 商品 38 的价格区间是中，功效是保湿、抗皱、淡纹去皱、祛痘、紧致提拉、美白，适用年龄是 30-40 岁、40-50 岁，类目是精华，改善肌肤问题是斑、日晒斑、暗沉、松弛下垂、毛孔粗大、痘痘、皱纹细纹、红血丝、色斑、起皮，适用肤质是中性、干性、敏感、油性、混合 [SEP] 商品 39 的价格区间是中，功效是保湿、抗皱、祛斑、美白，适用性别是女，类目是面部护理套装，改善肌肤问题是斑、暗沉、毛孔粗大、痘印、痘痘、皱纹细纹、粉刺、脂肪粒、色斑、雀斑、黑头，适用肤质是干性、油性、混合 [SEP] 商品 40 的价格区间是中，适用人群是妈妈，功效是保湿、抗皱，适用年龄是 30-40 岁，适用性别是女，类目是洁面，改善肌肤问题是斑、暗沉、松弛下垂、毛孔粗大、皱纹细纹、黑头，适用肤质是中性、干性、油性、混合 [SEP] 商品 41 的价格区间是低 [SEP] 商品 42 的价格区间是低 [SEP] 商品 43 的价格区间是中，适用人群是儿童、妈妈、家人、青少年，功效是保湿、控油、清洁、祛斑、祛痘，适用性别是女，类目是洁面，改善肌肤问题是斑、暗沉、毛孔粗大、痘印、痘痘、皱纹细纹、眼袋、粉刺、粗糙、色斑、黑头，适用肤质是中性、干性、敏感、油性、混合 [SEP] 商品 44 的价格区间是低，适用年龄是 50 岁以上，类目是乳液/面霜，适用肤质是干性、混合 [SEP] 商品 45 的价格区间是高，类目是面膜，改善肌肤问题是暗沉、痘痘、皱纹细纹、色斑、黑头，适用肤质是中性、干性 [SEP] 商品 46 的价格区间是中，功效是保湿、祛斑、美白，类目是乳液/面霜，改善肌肤问题是斑、暗沉、松弛下垂、皱纹细纹、色斑，适用肤质是干性、敏感、油性、混合 [SEP] 商品 47 的价格区间是低，类目是乳液/面霜，改善肌肤问题是嘴唇干，适用肤质是油性 [SEP] 商品 48 的价格区间是高，适用人群是妈妈、家人，功效是保湿、抗皱、改善气色、紧致提拉，适用年龄是 30-40 岁、40-50 岁、50 岁以上，适用性别是女，类目是面部护理套装，改善肌肤问题是斑、暗沉、松弛下垂、毛孔粗大、痘痘、皱纹细纹、眼纹、粗糙、红血丝、色斑、黑头，适用肤质是中性、干性、敏感、油性、混合 [SEP] 商品 49 的价格区间是中，功效是保湿、抗皱、淡纹去皱，类目是精华，改善肌肤问题是皱纹细纹，适用肤质是干性、混合 [SEP] 商品 50 的价格区间是高，适用人群是家人，功效是保湿、抗皱，适用年龄是 30-40 岁、40-50 岁，类目是乳液/面霜，改善肌肤问题是斑、暗沉、痘痘、皱纹细纹、色斑，适用肤质是中性、干性、混合。

Conversation history. 下面是你与用户正在进行的售前对话：用户：三十几岁用哪款水乳 [SEP] 客服：亲是什么肤质呢 [SEP] 用户：嗯嗯，混合肤质 [SEP] 客服：亲亲，您目前是想改善哪些肌肤问题呢（如：痘痘，干燥，毛孔，细纹，暗沉暗黄，色斑等）[SEP] 用户：细纹暗沉色斑

Output instruction. 请你从上面提供的 50 个候选商品中挑选 1 个最符合用户需求和偏好的商品。重复这个行为 10 次，每一次选择一个不同的商品。请你告诉我，你每一次选择了哪个商品。

Chinese-Alpaca2 response. 第一次选择：商品 37 - 价格区间为高，适用于混合肤质，改善肌肤问题包括皱纹细纹、色斑等问题。

第二次选择：商品 3 - 价格区间为中，适用于中性肤质，改善肌肤问题包括保湿、去眼袋、抗皱、控制油脂分泌等方面。

第三次选择：商品 1 - 价格区间为高，适用于中年人、中老年人、妈妈、家人、朋友、男朋友、老年人，具有多种功能，如保湿、去眼袋、嫩化肌肤等等。

第四次选择：商品 2 - 价格区间为高，适用于干性肤质，主要解决鸡皮问题，并具备良好的吸收效果。

第五次选择：商品 4 - 价格区间为低，适用于干性肤质，适合中性肤质的人群，主要用于保湿方面。

第六次选择：商品 10 - 价格区间为低，适用于中性、干性、敏感或混合肤质的用户，主要针对暗沉、痘痘、皱纹细纹等问题进行改善。

第七次选择：商品 11 - 价格区间为中，适用于女性，特别是中年人，旨在改善暗沉、老化现象以及其他一些常见肌肤问题。

第八次选择：商品 12 - 价格区间为低，适用于油性肤质，主要解决毛孔粗大、粉刺、粗糙等问题。

第九次选择：商品 13 - 价格区间为高，适用于油性肤质，主要解决毛孔粗大、粉刺、粗糙等问题。

第十次选择：商品 14 - 价格区间为中，适用于不同年龄段的女性，尤其是中年人、儿童和母亲，能够满足各种护肤需求，如保湿、抗皱、抗氧化等。

We provide an English language version of the Chinese example above.

Task description. [INST] ⟨⟨SYS⟩⟩ You are a pre-sales customer service in the e-commerce field. Please provide a detailed response that is professional, logical, authentic and valuable. ⟨⟨/SYS⟩⟩ **{Candidate items} {Conversation history} {Output instruction}** [/INST]

Candidate items. The following is the product knowledge you can refer to: The price range of product 1 is high, and it is applicable to middle-aged people, middle-aged and elderly people, mothers, family members, friends, boyfriends, and the elderly. It has the effects of moisturizing, removing eye bags, smoothing skin, anti-wrinkle, oil control, shrinking pores, improving complexion, lightening wrinkles, removing spots, removing acne, firming and lifting, whitening, and nourishing and moisturizing. It is applicable to ages under 10 years old, 20-30 years old, 30-40 years old, 40-50 years old, and over 50 years old. It is applicable to females. The category is facial care sets. It improves skin problems such as spots, sun spots, dullness, sagging, enlarged pores, acne marks, pimples, wrinkles and fine lines, eye lines, eye bags, roughness, red bloodshot, age spots, pigmentation, peeling, closed mouth, blackheads, and dark circles. It is applicable to yellow skin color and is applicable to Skin types are neutral, dry, sensitive, oily, and mixed [SEP] The price range of product 2 is high, and its effects are moisturizing, removing chicken skin, and easy absorption. The category is body lotion. It improves skin problems such as chicken skin and is suitable for dry skin [SEP] The price range of product 3 is high, and the applicable population is middle-aged people, boyfriends, and the elderly. The effects are moisturizing, anti-wrinkle, improving complexion, removing spots, firming and lifting, and whitening. The applicable age is 30-40 years old and over 50 years old. The applicable gender is female. The category is lotion/cream. It improves skin problems such as spots, dullness, sagging, large pores, acne, wrinkles and fine lines, red blood streaks, spots, peeling, closed comedones, and blackheads. It is suitable for neutral, dry, sensitive, oily, and mixed skin [SEP] The price range of product 4 is low, The applicable gender is female, the category is lotion/cream, and the applicable skin types are dry and mixed [SEP] The price range of product 5 is medium, the applicable population is children and mothers, the effect is moisturizing, the applicable gender is female, the category is lotion/cream, the skin problems to be improved are wrinkles and fine lines, and the applicable skin types are neutral, dry, sensitive, oily, and mixed [SEP] The price range of product 6 is low [SEP] The price range of product 7 is high, the applicable population is mothers, the usage scenario is daily, the effects are moisturizing, anti-wrinkle, shrinking pores, improving complexion, lightening wrinkles and removing wrinkles, removing spots, firming and lifting, whitening, and sun protection, the applicable age is 20-30 years old, 28-35 years old, and 30-40 years old, the category is facial care set, and the skin problems to be improved are spots, sun spots, dullness, and sagging Sagging, enlarged pores, acne marks, pimples, acne, wrinkles and fine lines, eye lines, eye bags, blackheads, roughness, red blood streaks, age spots, fat particles, pigmentation, peeling, closed comedones, blackheads, dark circles, applicable skin types are neutral, dry, sensitive, oily, mixed [SEP] The price range of product 8 is high, the category is facial care set, improves skin problems are dullness, pigmentation, applicable skin type is dry [SEP] The price range of product 9 is high, the category is cleansing, applicable skin types are neutral, dry, sensitive, mixed [SEP] The price range of product 10 is low, the effect is moisturizing, applicable age is 20-30 years old, 30-40 years old, applicable gender is female, male, category is facial care set, improves skin problems are dullness, acne, wrinkles and fine lines, applicable The skin types are neutral, dry, sensitive, oily, and mixed [SEP] The price range of product 11 is medium, the applicable population is the elderly, the effect is moisturizing, the applicable gender is female, the category is facial care set, and the skin problems to be improved are spots, dullness, wrinkles and fine lines, and the applicable skin types are neutral, dry, sensitive, oily, and mixed [SEP] The price range of product 12 is low [SEP] The price range of product 13 is high, the effect is moisturizing, the category is facial mask, and the skin problems to be improved are large pores, acne, roughness, and blackheads, and the applicable skin type is oily [SEP] The price range of product 14 is medium, the applicable population is middle-aged people, children, mothers, and family members, the applicable seasons are spring, summer, autumn, and winter, and the effects are moisturizing, anti-wrinkle, anti-allergic, oil control, shrinking pores, improving Complexion, freckle removal, acne removal, firming and lifting, whitening, sun protection, applicable age is under 10 years old, 30-40 years old, over 50 years old, applicable gender is female and male, category is facial care set, improve skin problems are freckles, sun spots, dullness, sagging, large pores, acne marks, pimples, acne, whiteheads, wrinkles and fine lines, blackheads, roughness, red blood streaks, fat particles, pustules, spots, peeling, closed comedones, chicken skin, blackheads, dark circles, applicable skin types are neutral, dry, sensitive, oily, mixed [SEP] The price range of product 15 is medium, the usage scenario is daily, the category is sunscreen [SEP] The price range of product 16 is high, applicable population is mothers and the elderly, the effects are moisturizing, anti-wrinkle, freckle removal, firming and lifting, applicable age is 10 Under 30, 30-40, 40-50, 50 and over, applicable gender is female, category is lotion/cream, skin problems include spots, dullness, sagging, large pores, acne, wrinkles, eye bags, roughness, pigmentation, blackheads, applicable skin types include neutral, dry, sensitive, oily, mixed [SEP] The price range of product 17 is high, and its effects are moisturizing, anti-wrinkle, firming and lifting, whitening, category is lotion/cream, skin problems include dullness, wrinkles, fine lines, peeling, applicable skin types include dry, sensitive, mixed [SEP] The price range of product 18 is high [SEP] The price range of product 19 is high, and its effects are moisturizing, smooth skin, anti-wrinkle, shrink pores, improve complexion, lighten wrinkles, remove spots, firm and lift, Whitening, applicable age is over 50 years old, category is facial mask, improves skin problems such as spots, dullness, sagging, large pores, acne, wrinkles and fine lines, eye wrinkles, red blood streaks, pigmentation, blackheads, melanin, applicable skin types are neutral, dry, sensitive, oily, mixed [SEP] The price range of product 20 is medium, the effect is moisturizing, category is lotion/cream [SEP] The price range of product 21 is low [SEP] The price range of product 22 is high, the effect is moisturizing, anti-wrinkle, firming and lifting, applicable gender is female, category is lotion/toner, improves skin problems such as spots, dullness, sagging, large pores, acne, wrinkles and fine lines, pigmentation, peeling, blackheads, applicable skin types are neutral, dry, sensitive, oily, mixed [SEP] The price range of product 23 is high, its effect is moisturizing, its category is lotion/cream, and its applicable skin types are neutral, dry, sensitive, and mixed [SEP] The price range of product 24 is low [SEP] The price range of product 25 is high, its effect is freckle removal, its category is facial cleanser, and its skin problems include freckles, dullness, wrinkles, fine lines, pigmentation, and dark circles, and its applicable skin types are neutral, dry, and mixed [SEP] The price range of product 26 is medium, its effect is moisturizing, and its applicable gender is male, its category is facial care set, and its skin problems include freckles, dullness, sagging, enlarged pores, acne, wrinkles, fine lines, blackheads, and its applicable skin types are neutral, dry, oily, and mixed [SEP] The price range of product 27 is medium, its effect is improving complexion, and its applicable gender is male, its category is facial care set, and its skin problems include freckles, dullness, sagging, enlarged pores, acne, wrinkles, fine lines, blackheads, and its applicable skin types are neutral, dry, oily, and mixed [SEP] Gender is male, category is facial care set, improves skin problems such as dullness, wrinkles, fine lines, and flaking, applicable skin types are neutral, dry, sensitive, oily, and mixed [SEP] The price range of product 28 is high, category is hand cream, improves skin problems such as dullness, wrinkles, fine lines, and pigmentation, applicable skin types are neutral, dry, oily, and mixed [SEP] The price range of product 29 is low [SEP] The price range of product 30 is medium, category is lotion/cream, improves skin problems such as enlarged pores, applicable skin types are sensitive, oily, and mixed [SEP] The price range of product 31 is medium, applicable people are mothers and family members, the effects are moisturizing, repairing and conditioning, removing eye bags, anti-wrinkle, oil control, improving complexion, removing spots, removing acne, firming and lifting, whitening, Nourishing and moisturizing, applicable age groups are 10-20 years old, under 10 years old, 20-30 years old, 30-40 years old, and over 50 years old, applicable genders are female and male, category is facial care set, skin problems to be improved are spots, sun spots, after sun exposure, dullness, sagging, large pores, acne marks, pimples, wrinkles and fine lines, eye lines, eye bags, blackheads, roughness, red bloodshot, age spots, pigmentation, peeling, closed comedones, freckles, chicken skin, blackheads, dark circles, and melanin, applicable skin colors are yellow and black, applicable skin types are neutral, dry, sensitive, oily, and mixed [SEP] The price range of product 32 is low [SEP] The price range of product 33 is high, category is lotion/cream, skin problems to be improved are wrinkles and fine lines, applicable skin types are mixed [SEP] Product 3 The price range of item 4 is low [SEP] The price range of item 35 is high [SEP] The price range of item 36 is high, and its effects are moisturizing, anti-wrinkle, improving complexion, removing spots, firming and lifting, and whitening. The applicable age group is 30-40 years old and 40-50 years old. The category is essence. The skin problems it improves are spots, dullness, sagging, large pores, acne marks, pimples, wrinkles and fine lines, eye wrinkles, eye bags, blackheads, pigmentation, peeling, blackheads, dark circles, and melanin. The applicable skin types are neutral, dry, sensitive, oily, and mixed [SEP] The price range of item 37 is high [SEP] The price range of item 38 is medium, and its effects are moisturizing, anti-wrinkle, light wrinkle removal, acne removal, firming and lifting, and whitening. The applicable age group is 30-40 years old and 40-50 years old , the category is essence, improves skin problems such as spots, sun spots, dullness, sagging, large pores, acne, wrinkles and fine lines, red blood streaks, pigmentation, and peeling, and is applicable to neutral, dry, sensitive, oily, and mixed skin types. [SEP] The price range of product 39 is medium, and its effects are moisturizing, anti-wrinkle, freckle removal, and whitening. The applicable gender is female. The category is facial care set, improves skin problems such as spots, dullness, large pores, acne marks, acne, wrinkles and fine lines, blackheads, fat particles, pigmentation, freckles, and blackheads. The applicable skin types are dry, oily, and mixed. [SEP] The price range of product 40 is medium, and is applicable to mothers. Its effects are moisturizing and anti-wrinkle. The applicable age is 30-40 years old. The applicable gender is female. The category is cleansing, improves skin problems such as spots, dullness, large pores, acne marks, acne, wrinkles and fine lines, blackheads, fat particles, pigmentation, freckles, and blackheads. The applicable skin types are dry, oily, and mixed. The price range of product 41 is low [SEP] The price range of product 42 is low [SEP] The price range of product 43 is medium. The applicable population is children, mothers, family members, and teenagers. The efficacy is moisturizing, oil control, cleaning, freckle removal, and acne removal. The applicable gender is female. The category is facial cleanser. The skin problems to be improved are freckles, dullness, large pores, acne marks, acne, wrinkles, fine lines, eye bags, blackheads, roughness, pigmentation, and blackheads. The applicable skin types are neutral, dry, sensitive, oily, and mixed [SEP] The price range of product 44 is low. The applicable age is over 50 years old. The category is lotion/cream. The applicable skin types are dry and mixed [S EP] The price range of product 45 is high, the category is facial mask, it can improve skin problems such as dullness, acne, wrinkles, fine lines, pigmentation, and blackheads, and is suitable for neutral and dry skin types [SEP] The price range of product 46 is medium, its effects are moisturizing, freckle removal, and whitening, the category is lotion/cream, it can improve skin problems such as freckles, dullness, sagging, wrinkles, fine lines, and pigmentation, and is suitable for dry, sensitive, oily, and mixed skin types [SEP] The price range of product 47 is low, the category is lotion/cream, it can improve skin problems such as dry lips, and is suitable for oily skin types [SEP] The price range of product 48 is high, it is suitable for mothers and family members, its effects are moisturizing, anti-wrinkle, improving complexion, firming and lifting, and is suitable for ages 30-40, 40-50, and 50-60. 0 years old and above, applicable gender is female, category is facial care set, improves skin problems such as spots, dullness, sagging, enlarged pores, acne, wrinkles and fine lines, eye wrinkles, roughness, red blood streaks, pigmentation, and blackheads, applicable skin types are neutral, dry, sensitive, oily, and mixed [SEP] The price range of product 49 is medium, the efficacy is moisturizing, anti-wrinkle, and wrinkle removal, the category is essence, improves skin problems such as wrinkles and fine lines, applicable skin types are dry and mixed [SEP] The price range of product 50 is high, applicable population is family, the efficacy is moisturizing and anti-wrinkle, applicable age is 30-40 years old, 40-50 years old, category is lotion/cream, improves skin problems such as spots, dullness, acne, wrinkles and fine lines, and pigmentation, applicable skin types are neutral, dry, and mixed.

Conversation history. Below is the pre-sales conversation you are having with the user: User: Which lotion should I use for people in their thirties? [SEP] Customer Service: What's your skin type? [SEP] User: Yes, mixed skin [SEP] Customer Service: Dear, what skin problems do you want to improve at the moment? (eg: acne, dryness, pores, fine lines, dullness, yellowing, pigmentation, etc.) [SEP] User: Fine lines, dullness, pigmentation

Output instruction. Please select one product that best meets user needs and preferences from the 50 candidate products provided above. Repeat this behavior 10 times, choosing a different product each time. Please tell me which product you chose each time.

Chinese-Alpaca2 response. First choice: Product 37 - The price range is high, suitable for mixed skin, improves skin problems including wrinkles, fine lines, spots and other problems.
Second choice: Product 3 - The price range is medium, suitable for neutral skin, improves skin problems including moisturizing, removing eye bags, anti-wrinkle, and controlling oil secretion.
Third choice: Product 1 - The price range is high, suitable for middle-aged people, middle-aged and elderly people, mothers, family, friends, boyfriends, and the elderly, with multiple functions such as moisturizing, removing eye bags, and tenderizing skin.
Fourth choice: Product 2 - The price range is high, suitable for dry skin, mainly solves chicken skin problems, and has good absorption effect.
Fifth choice: Product 4 - The price range is low, suitable for dry skin, suitable for people with neutral skin, mainly used for moisturizing.
Sixth choice: Product 10 - The price range is low, suitable for users with neutral, dry, sensitive or mixed skin, mainly for dullness, acne, wrinkles and fine lines.
Seventh choice: Product 11 - The price range is medium, suitable for women, especially middle-aged people, aiming to improve dullness, aging and other common skin problems.
Eighth choice: Product 12 - The price range is low, suitable for oily skin, mainly solving problems such as large pores, acne, roughness.
Ninth choice: Product 13 - The price range is high, suitable for oily skin, mainly solving problems such as large pores, acne, roughness.
Tenth choice: Product 14 - The price range is medium, suitable for women of different ages, especially middle-aged people, children and mothers, and can meet various skin care needs, such as moisturizing, anti-wrinkle, anti-oxidation, etc.

B.2 Prompt Template for Response Generation

B.2.1 Prompt Template for Llama 2. To make Llama 2 generate a response regarding the recommended movie(s), we adopt a prompt template with three segments: (i) task description, (ii) recommended movie(s), and (iii) conversation history. An example is given below, along with the response generated by Llama 2.

Task description. Provide a response based on the given conversation history between Seeker and Recommender. Recommender is supposed to recommend a movie to the Seeker. Your role is Recommender. You will have the recommended movie(s) by the Recommender.

Please play the role of Recommender and provide a response based on the given conversation history.

Recommended movies. The recommended movie(s) is/are: 48 Hrs. (1982).
Please be careful to mimic the speaking style of the Recommender. Now give a response to the Seeker.

Conversation history. The conversation history is: Seeker: Hi I am looking for a movie like Super Troopers (2001) [SEP] Recommender: You should watch Police Academy (1984) [SEP] Seeker: Is that a great one? I have never seen it. I have seen American Pie [SEP] Seeker: I mean American Pie (1999) [SEP] Recommender: Yes Police Academy (1984) is very funny and so is Police Academy 2: Their First Assignment (1985) [SEP] Seeker: It sounds like I need to check them out [SEP] Recommender: yes you will enjoy them [SEP] Seeker: I appreciate your time. I will need to check those out. Are there any others you would recommend? [SEP] Recommender: yes Lethal Weapon (1987) [SEP] Seeker: Thank you i will watch that too [SEP] Recommender: and also Beverly Hills Cop (1984) [SEP] Seeker: Thanks for the suggestions. [SEP] Recommender: you are welcome.

Llama2 response. Of course, I'd be happy to help! As the Recommender, I would say:
"Ah, I see! So you're looking for a movie similar to Super Troopers (2001). Well, I think you'll enjoy Police Academy (1984) and Police Academy 2: Their First Assignment (1985) - they're both hilarious! But if you're looking for something else, I would recommend 48 Hrs. (1982). It's a classic action-comedy film that's sure to keep you entertained. I hope you enjoy it! Let me know if you need any more recommendations."

B.2.2 Prompt Template for Chinese-Alpaca2. For the U-NEED dataset, we supplement the prompt template with identified user needs. An example is given below, along with the response generated by Chinese-Alpaca2.

Task description. [INST] ⟨⟨SYS⟩⟩ 你是一个电商领域的售前导购客服。请你提供专业、有逻辑、内容真实、有价值的详细回复。⟨⟨/SYS⟩⟩ **{Conversation history} {Identified user needs} {Recommended item(s)} {Output instruction}** [/INST]

Conversation history. 下面是你与用户正在进行的售前对话：用户：你好，30 岁以上抗老的面霜有什么推荐的吗 [SEP] 客服：亲爱的，除了抗皱，是否有美白或亮肤的需求的呢？[SEP] 用户：如果有也最好了。主要是抗老 [SEP] 客服：亲爱的，悦薇是有抗皱亮肤的功效的哦，功效是比较全的哦 [SEP] 客服：仅发送商品链接 [SEP] 客服：塑颜焕亮紧致上扬；复合植萃 VP8 抗老成分，褪黄焕亮，延缓肌肤胶原蛋白老化；旭日红花精粹，紧致弹润，促进肌肤胶原焕生能量。[SEP] 用户：对毛孔粗大的～呢 [SEP] 客服：亲爱的，建议可以试试红腰子的哦。

Identified user needs. 下面是已获取的用户需求偏好信息：功效：抗老、抗老、年龄：30 岁以上、品类：面霜、肌肤问题：毛孔粗大

Recommended item(s). 满足用户需求的商品信息：商品 1 满足用户需求，1 的价格区间是高，功效是保湿、抗皱、抗过敏、控油、收缩毛孔、改善气色、消闭口、祛斑、紧致提拉、美白、自然、防晒，适用年龄是 30-40 岁，类目是精华，改善肌肤问题是斑、暗沉、松弛下垂、毛孔粗大、痘印、痘痘、皱纹细纹、眼纹、红血丝、起皮、闭口、黑眼圈，适用肤色是黄，适用肤质是中性、干性、敏感、油性、混合

Output instruction. 请你模仿客服的说话风格，基于售前对话的内容，给出这一轮的回复。

Chinese-Alpaca2 response. 亲爱的顾客您好！ 根据您提供的需求和喜好，我为您推荐一款名为“悦薇”的面霜。这款面霜具有全面的抗衰老效果，能够有效淡化细纹、提升皮肤弹性，同时还具备美白和亮肤的效果。此外，它还含有多种植物提取物，可以滋润肌肤并缩小毛孔。对于您的年龄在 30 岁以上且希望达到全面抗衰老效果的要求，悦薇是一个不错的选择。

关于价格方面，悦薇属于中高档价位的产品，但它的质量和效果都非常值得信赖。如果您对这款产品感兴趣，我可以给您提供购买链接或者更多详细的产品介绍。同时，我还注意到您提到了毛孔粗大的问题，我们还有其他一些适合改善毛孔粗大的护肤品可供选择，如红腰子等。请问您是否需要了解更多关于这些产品的信息呢?

We provide an English language version of the Chinese example above.

Task description. [INST] ⟨⟨SYS⟩⟩ You are a pre-sales customer service in the e-commerce field. Please provide a detailed response that is professional, logical, authentic and valuable. ⟨⟨/SYS⟩⟩ **{Conversation history} {Identified user needs} {Recommended item(s)} {Output instruction}** [/INST]

Conversation history. Below is the pre-sales conversation you are having with the user: User: Hello, do you have any recommendations for anti-aging creams for people over 30 years old? [SEP] Customer Service: Dear, in addition to anti-wrinkle, do you need whitening or skin brightening? [SEP] User: It would be great if you do. Mainly anti-aging [SEP] Customer Service: Dear, Yuewei has the effect of anti-wrinkle and skin brightening, and the effects are relatively comprehensive [SEP] Customer Service: Only send product links [SEP] Customer Service: Sculpting, brightening, firming and lifting; compound plant extract VP8 anti-aging ingredients, fade yellowing and brightening, delay skin collagen aging; Rising Sun Red Flower Essence, firming and elastic, promote skin collagen to rejuvenate energy. [SEP] User: What about large pores? [SEP] Customer Service: Dear, I suggest you try Red Waist.

Identified user needs. The following is the user preference information that has been obtained: Efficacy: anti-aging, anti-aging, age: over 30 years old, category: face cream, skin problem: large pores

Recommended item(s). Product information that meets user needs: Product 1 meets user needs, the price range of 1 is high, the effects are moisturizing, anti-wrinkle, anti-allergic, oil control, shrinking pores, improving complexion, eliminating closed comedones, removing spots, firming and lifting, whitening, natural, and sun protection, the applicable age is 30-40 years old, the category is essence, and the skin problems improved are spots, dullness, sagging, enlarged pores, acne marks, pimples, wrinkles and fine lines, eye wrinkles, red bloodshot, peeling, closed comedones, and dark circles. The applicable skin color is yellow, and the applicable skin types are neutral, dry, sensitive, oily, and mixed.

Output instruction. Please imitate the customer service's speaking style and give this round of response based on the content of the pre-sales conversation.

Chinese-Alpaca2 response. Dear customer, hello! Based on the needs and preferences you provided, I recommend a cream called "Yuewei" for you. This cream has a comprehensive anti-aging effect, can effectively fade fine lines, improve skin elasticity, and also has whitening and brightening effects. In addition, it also contains a variety of plant extracts, which can moisturize the skin and shrink pores. For those who are over 30 years old and want to achieve comprehensive anti-aging effects, Yuewei is a good choice.
Regarding the price, Yuewei is a mid-to-high-end product, but its quality and effect are very trustworthy. If you are interested in this product, I can provide you with a purchase link or more detailed product introduction. At the same time, I also noticed that you mentioned the problem of large pores. We have some other skin care products suitable for improving large pores, such as Hongyaozi. Do you need to know more about these products?

B.2.3 Prompt Template for NR-CRS (Llama 2). To make NR-CRS (Llama 2) generate responses regarding the movies mined from neighboring information, we adopt a prompt template with three segments: (i) task description, (ii) recommended movie(s), (iii) additional movies, and (iv) conversation history. An example is given below, along with the response generated by NR-CRS (Llama 2). The recommended movie(s) and conversation history are the same as those in B.2.1.

Task description. Provide a response based on the given conversation history between Seeker and Recommender. Recommender is supposed to recommend a movie to the Seeker. Your role is Recommender. You will have the recommended movie(s) by the Recommender. Please play the role of Recommender and provide a response based on the given conversation history.

Additional movies. The 10 movies are: Wonder Woman, Jumanji, Deadpool, The Dark Knight, It, Coco, The Avengers, Dunkirk, Titanic, Iron Man

NR-CRS (Llama 2) response. Of course, I'd be happy to help! Based on your conversation history with the Seeker, it seems like they're looking for a movie similar to Super Troopers (2001). As the Recommender, I would suggest the movie 48 Hrs. (1982) as it's a classic action-comedy film that is similar in tone and style to Super Troopers.
Here's my response:
Recommender: Absolutely, my friend! I think you'll love 48 Hrs. (1982). It's a hilarious and action-packed movie that's similar to Super Troopers in terms of its humor and lighthearted tone. I think you'll find it right up your alley!
I hope you enjoy it! Let me know if you have any other questions or if you'd like more recommendations

B.2.4 Prompt Template for NR-CRS (Chinese-Alpaca2). For U-NEED dataset, we supplement the prompt template with 10 additional products from neighboring information. An example is given below, along with the response generated by NR-CRS (Chinese-Alpaca2).

Task description. [INST] ⟨⟨SYS⟩⟩ 你是一个电商领域的售前导购客服。请你提供专业、有逻辑、内容真实、有价值的详细回复。⟨⟨/SYS⟩⟩ **{Conversation history} {Identified user needs} {Recommended item(s)} {Additional products} {Output instruction}** [/INST]

The conversation history, identified user needs and recommended movie(s) are the same as those in B.2.2.

Additional products. 除了上面的内容，你还有额外的相邻商品的属性和属性值信息。10 个相邻商品来自多个其他的售前对话。这些售前对话与当前售前对话相似。这 10 个相邻商品的属性和属性值分别是: 商品 A 的价格区间是中 [SEP] 商品 B 的价格区间是低 [SEP] 商品 C 的价格区间是高 [SEP] 商品 D 的价格区间是中 [SEP] 商品 E 的价格区间是中 [SEP] 商品 F 的价格区间是中 [SEP] 商品 G 的价格区间是中 [SEP] 商品 H 的价格区间是中 [SEP] 商品 I 的价格区间是低 [SEP] 商品 J 的价格区间是低

Output instruction. 请你模仿客服的说话风格，基于售前对话的内容，并参考提供的 10 个商品，给出这一轮的回复。

Chinese-Alpaca2 response. 亲爱的顾客您好！ 根据您之前的描述，我们为您提供了一些适合您的产品选择。首先，如果您需要全面的抗衰老效果，我强烈推荐我们的悦薇面霜。这款面霜含有多种抗衰老成分，如 VC 光合物、维生素 C 等，能够有效抵抗皮肤的老化过程，同时提升肌肤弹性和亮度。此外，它还具有美白功能，可以淡化黑色素，使肌肤更加明亮。对于毛孔粗大的问题，我可以向您推荐一款名为红腰子的产品。这款产品采用天然植物提取物，能有效地缩小毛孔，减少油脂分泌，从而达到清洁毛孔的效果。另外，它还能够滋润肌肤，增加水分含量，让肌肤看起来更光滑细腻。
当然，以上只是我们提供的部分选项，还有其他许多其他类型的护肤品可供您选择。如果您对我们的产品还有任何疑问或者想了解更多关于它们的信息，请随时告诉我，我会尽力帮助您解答。

We provide an English language version of the Chinese example above.

Task description. [INST] ⟨⟨SYS⟩⟩ You are a pre-sales customer service in the e-commerce field. Please provide a detailed response that is professional, logical, authentic and valuable. ⟨⟨/SYS⟩⟩ **{Conversation history} {Identified user needs} {Recommended item(s)} {Additional products} {Output instruction}** [/INST]

The conversation history, identified user needs and recommended movie(s) are the same as those in B.2.2.

Additional products. In addition to the above, you have additional attribute and attribute value information for neighboring items. The 10 neighboring items come from multiple other pre-sales conversations. These pre-sales conversations are similar to the current pre-sales conversation. The attributes and attribute values of these 10 neighboring items are: The price range of item A is medium [SEP] The price range of item B is low [SEP] The price range of item C is high [SEP] The price range of item D is medium [SEP] The price range of item E is medium [SEP] The price range of item F is medium [SEP] The price range of item G is medium [SEP] The price range of item H is medium [SEP] The price range of item I is low [SEP] The price range of item J is low

Output instruction. Please imitate the customer service's speaking style, give this round of response based on the content of the pre-sales conversation and the 10 products provided.

Chinese-Alpaca2 response. Dear customer, Hello! According to your previous description, we have provided you with some product options suitable for you. First of all, if you need a comprehensive anti-aging effect, I strongly recommend our Yuewei Cream. This cream contains a variety of anti-aging ingredients, such as VC photosynthesis, vitamin C, etc., which can effectively resist the aging process of the skin while improving skin elasticity and brightness. In addition, it also has a whitening function, which can dilute melanin and make the skin brighter.
For the problem of large pores, I can recommend you a product called Red Waist. This product uses natural plant extracts, which can effectively shrink pores and reduce oil secretion, thereby achieving the effect of cleaning pores. In addition, it can also moisturize the skin, increase moisture content, and make the skin look smoother and more delicate.
Of course, the above are only some of the options we provide, and there are many other types of skin care products for you to choose from. If you have any questions about our products or want to know more about them, please feel free to let me know and I will do my best to help you answer them.

Received 22 February 2024; revised 6 October 2024; accepted 8 January 2025