Multi-Turn Multi-Modal Question Clarification for Enhanced Conversational Understanding

Anonymous ACL submission

Abstract

Conversational query clarification enables users to refine their search queries through interactive dialogue, improving search effectiveness. Traditional approaches rely on text-based clarifying questions, which often fail to capture complex user preferences, particularly those involving visual attributes. While recent work has explored single-turn multi-modal clarification with images alongside text, such methods do not fully support the progressive nature of user intent refinement over multiple turns. Motivated by this, we introduce the Multi-turn Multi-modal Clarifying Questions (MMCQ) task, which combines text and visual modalities to refine user queries in a multi-turn conversation. To facilitate this task, we create a large-scale dataset named ClariMM comprising over 13k multi-turn interactions and 33k question-answer pairs containing multi-modal clarifying questions. We propose Mario, a retrieval framework that employs a two-phase ranking strategy: initial retrieval with BM25, followed by a multi-modal generative re-ranking model that integrates textual and visual information from conversational history. Our experiments show that multi-turn multi-modal clarification outperforms uni-modal and single-turn approaches, improving MRR by 12.88%. The gains are most significant in longer interactions, demonstrating the value of progressive refinement for complex queries.

1 Introduction

Conversational search (CS) enables users and systems to collaboratively refine queries through dialogue (Radlinski and Craswell, 2017), addressing limitations of traditional keyword-matching systems where single queries often fail to capture complete information needs (Aliannejadi et al., 2019; Zamani et al., 2020). Query clarification has emerged as a key mechanism for improving search accuracy by helping users refine ambiguous or incomplete queries (Hancock et al., 2019; Rao and III, 2018).

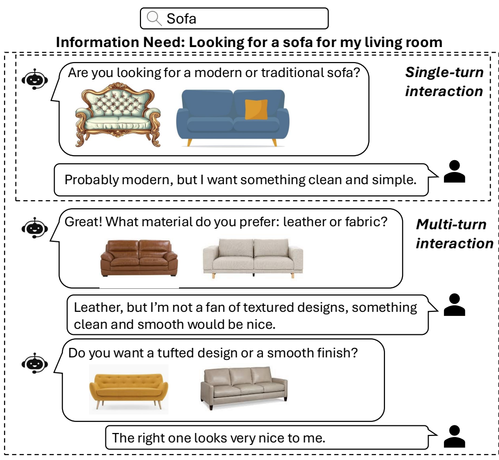

Figure 1: An example conversation comparing the multi-modal query clarification under single-turn and multi-turn scenarios.

Current approaches to query clarification, while showing promise, still face limitations in addressing complex information needs. Traditional systems rely predominantly on text-only clarifying questions (Aliannejadi et al., 2021; Zamani et al., 2020), proving insufficient when users need to understand or express preferences about visual characteristics. This limitation is particularly evident in domains such as healthcare (symptom identification), e-commerce (product selection), and technical support (problem diagnosis), where visual context is crucial for precise understanding (Siro et al., 2025).

Recent work (Yuan et al., 2024) introduces the incorporation of visual content into clarifying questions, enabling systems to present images alongside text within a single interaction. However, restricting the interaction to a single turn hinders accurate intent inference, making it challenging to fully capture user needs. For example, in Figure 1, when searching for a sofa, users need to progressively refine their preferences from general style (modern vs. traditional) to specific materials (leather vs. fabric) and finally to detailed attributes (tufted vs. smooth). Such natural progression in preference articulation cannot be achieved in a single turn without overwhelming users with numerous options. While existing multi-turn approaches (Aliannejadi et al., 2020) support dialogue flow, they lack the crucial visual context for grounding language understanding.

To address these limitations, we introduce the novel task of Multi-turn Multi-modal Clarifying Questions (MMCQ) within open-domain CS systems. MMCQ enables systems to gradually refine user intent over multiple turns, where each interaction builds on the previous one by incorporating both textual questions and relevant images. This step-by-step process enhances the depth and accuracy of the clarification process, leading to more precise disambiguation of user intent and improved retrieval performance. To facilitate research in this direction, we create a new dataset named **ClariMM** that builds upon existing single-turn multi-modal clarification data (Yuan et al., 2024), comprising over 13k instances of multi-turn interactions with over 14k images and 33k question-answering pairs.

Furthermore, we propose a novel ranking model, called **Mario**(**M**ulti-turn Multi-mod**a**l Que**r**y Clarificat**io**n), devising a two-phase ranking method to rank documents based on multi-modal conversational history. Mario adopts the BM25 method for initial retrieval, followed by a multi-modal generative model with a constrained generation mechanism to refine and re-rank the results. Specifically, our model leverages a pretrained multi-modal large language model (LLM) to generate the keywords sequence of relevant documents by integrating textual and visual information from the conversational interaction history.

We compare the performance of Mario against a range of models, from traditional retrieval methods to several open-sourced LLMs, and analyze the impact of multi-modal vs. uni-modal approaches. Our experiments on ClariMM show that incorporating images in multi-turn scenarios improves MRR by up to 12.88% with Mario. Additionally, a comparison between ClariMM and its single-turn counterpart shows that multi-turn interactions consistently outperform single-turn approaches across all retrieval metrics in the multi-modal setting. Further analysis highlights Mario's superiority, particularly in longer interactions, demonstrating the benefits of multi-turn multi-modal clarification for CS.

In summary, our contributions are as follows:

- We introduce MMCQ, a novel task in mixed-initiative CS, enabling query refinement through multi-turn interactions that integrate textual and visual cues.
- We propose a large-scale dataset called ClariMM to support multi-modal interactive search[1]. We then propose Mario for effective multi-modal document retrieval in this setting.
- We demonstrate the effectiveness of our model on retrieval performance by comparing it with its uni-modal and single-turn counterparts.

2 Related Work

Conversational question clarification. Query clarification improves search by refining user queries with additional context (Wang et al., 2023b), addressing ambiguities in various tasks including entity disambiguation (Coden et al., 2015), voice-based interactions (Kiesel et al., 2018), question answering (Nakano et al., 2022) and recommendation (Zou et al., 2020). In mixed-initiative search systems, where the conversational initiative alternates between users and agents, targeted clarifying questions have been shown to improve retrieval performance and user satisfaction (Rahmani et al., 2024; Siro et al., 2024a). For instance, Aliannejadi et al. (2020) introduced the ClariQ benchmark, which employs clarifying questions to disambiguate vague queries. Building on these foundations, Yuan et al. (2024) advanced the field further by developing Melon, a system that integrates visual inputs into the clarification process, thereby helping users refine their queries more effectively. Despite these advances, challenges remain in effectively merging multi-modality with multi-turn conversational interactions.

Generative Retrieval. Generative retrieval is a paradigm in information retrieval that uses generative models to bypass the traditional "index-retrieve-then-rank" architecture and directly generate document identifiers in an end-to-end manner. Instead of relying on dense embeddings and nearest-neighbor search, models like GENRE (Cao et al., 2021), DSI (Chen et al., 2023b), and CorpusBrain (Chen et al., 2022) frame retrieval as

[1]We will release the dataset right after the paper acceptance

a sequence generation task, where relevant document identifiers are produced token-by-token. Generative retrieval has proven effective across various knowledge-intensive tasks, including entity linking (Cao et al., 2021) and question answering (Braslavski et al., 2017). Models like GENRE treat retrieval as a generation task by predicting entity names, while DSI and DSI-QG extend this idea to document retrieval using synthetic identifiers. Recent methods such as UGR (Chen et al., 2023a) and CorpusBrain further enhance the framework with prompt-based learning and corpus-level pre-training. Based on this, our work focuses on asking multi-modal clarifying questions in a multi-turn CS system and investigates whether it results in better retrieval performance.

3 Dataset Construction

We describe how we build ClariMM, our multi-turn multi-modal clarification dataset.

3.1 Data Collection

Our dataset builds upon Melon (Yuan et al., 2024), a single-turn dataset containing clarifying questions with images. We use Melon's topics and facets (user information needs), which originate from TREC Web Track 2009–2012 (Clarke et al., 2009, 2012), and the corresponding documents for each facet.

Multi-turn conversation synthesis. We construct multi-turn conversations by systematically combining QA pairs from Melon that share the same topic. For each topic, we exhaustively generate all possible combinations of single-turn QAs to create two-, three-, and four-turn conversations. Each turn retains its corresponding images from Melon, ensuring the diversity in clarification patterns.

Data sampling. The synthesis process generates extremely large subsets for two-, three-, and four-turn conversations, with the two-turn subset alone exceeding 1 million conversations. This vast dataset poses challenges for post-processing and analysis while also containing redundant and unnatural conversations. To address this issue, we randomly sample 10,000 conversations from each subset. This selection balances dataset size while maintaining diversity and relevance.

Data refinement. To enhance the naturalness of synthetic data and ensure more realistic conversations, we develop an automated refinement method

Algorithm 1 Multi-turn Conversation Refinement

Input: Conversation d with T turns, hidden intention F
Output: Refined conversation c

```
c ← {} // Initialize refined conversation
for t = 1 to T do
    if t == 1 then
        A'_t ← Θ_initial(Q_t, A_t, F) // Q_t, A_t denote the question-answer pair at turn t, Θ denotes the prompting strategy
    else if t < T then
        A'_t ← Θ_partial(Q_t, A_t, F)
    else
        A'_t ← Θ_final(Q_t, A_t, F)
    end if
    c ← c ∪ {(Q_t, A'_t)}
end for
```

using GPT-4o (Algorithm 1). While manual refinement would be ideal for ensuring conversation quality, it is impractical given our dataset scale. Our automated approach significantly reduces the required effort while maintaining high-quality dialogue refinement.

At the start of the conversation, we prompt GPT-4o to act as a user, interpreting the multi-modal conversational history and refining its responses without revealing the user's intent based on the given facet. This approach encourages a natural extension of the interaction, requiring additional exchanges to fully clarify the user's needs. As the conversation progresses, we iteratively refine responses to gradually unveil the hidden intent, effectively simulating the natural flow of the clarification phase. We apply this method to the filtered 30k dialogues, ensuring that the generated dialogues remain coherent and engaging while gradually revealing the hidden intent, preventing it from being disclosed too early. The detailed annotation pipeline and all prompts used are provided in Appendix A.

3.2 Quality Control

To validate the quality of our synthetic dataset, we conducted a human evaluation assessing four key metrics: *relevance*, *coherence*, *diversity*, and *intent reveal*. These metrics were chosen to evaluate critical aspects of our dataset construction process (detailed definition of the metrics see Appendix B). Given our dataset's scale, we randomly sampled 10% of the topics for annotation. Two of the authors independently evaluated 178 conversations using a 5-point Likert scale (1: poor to 5: excellent). Our human evaluation results (Table 1) demonstrate the effectiveness of our construction approach. Relevance scores show consistent improvement from Turn 1 (3.62, σ=1.29)

Metric	Mean	Std Dev	Median
Relevance (Turn 1)	3.62	1.29	3.00
Relevance (Turn 2)	3.56	1.24	3.00
Relevance (Turn 3)	3.78	1.09	3.00
Relevance (Turn 4)	4.11	0.97	4.00
Coherence	3.36	1.10	3.00
Diversity	4.01	0.97	4.00
Intent reveal	4.65	0.87	5.00

Table 1: Human evaluation scores for relevance, coherence, diversity, and intent reveal.

Metric	Value
# topics	298
# facets	1,070
# all questions	4,969
# images	14,869
# answers	33,477
# 2-Turn Conversations	7,782 (59.36%)
# 3-Turn Conversations	3,391 (25.86%)
# 4-Turn Conversations	1,935 (14.78%)

Table 2: Statistics of the ClariMM dataset.

to Turn 4 (4.11, σ=0.97), validating our GPT-4o refinement strategy for maintaining topical focus. While coherence (3.36, σ=1.10) indicates some minor inconsistencies, the strong diversity score (4.01, σ=0.97) confirms that our sampling strategy captured varied aspects of topics without repetition. Most notably, the high intent completion score (4.65, σ=0.87) validates our approach of gradually revealing user intent across turns (additional evaluation see Appendix E). These results prove that our data generation pipeline successfully produces well-structured and semantically rich multi-turn conversations, making ClariMM a valuable resource for training multi-turn multi-modal retrieval systems.

3.3 Dataset Statistics

Table 2 provides an overview of the basic statistics of ClariMM. The dataset comprises a total of 298 search topics and 1070 facets. It consists of 4,969 clarifying questions accompanied by 14,869 images, resulting in an average of 2.99 images per question. Additionally, the dataset includes 33,477 answers and every question has its corresponding answers. Overall, the dataset consists of over 7k two-turn conversations, 3k three-turn conversations, and 1k four-turn conversations.

4 Our Method

4.1 Problem Formulation

Following Yuan et al. (2024), we consider a set of topics denoted as $T = \{t_1, t_2, \ldots, t_k\}$, serve as user queries. Each topic t_i is associated with a set of facets, defined as $F_i = \{f_i^1, f_i^2, \ldots, f_i^{n_i}\}$, where n_i represents the number of facets for t_i. Each facet f_i^j captures a distinct aspect of t_i, specifying a particular user information need. Given a topic t and an information need (facet) f, the user engages in a conversation C consisting of k turns. Each conversation comprises a sequence of **multi-modal** clarifying questions $Q = \{q_1, q_2, \ldots, q_k\}$ and their corresponding **text-only** answers $A = \{a_1, a_2, \ldots, a_k\}$. Each question q_i consists of text and may optionally include some images. At the end of each conversation, a set of documents D are retrieved and ranked based on the conversation. The goal is to identify the hidden facet (*i.e.*, user needs) f and learn a retrieval function $R(\cdot)$ that maps the conversation context and topic to a ranked list of documents, such that $R(C, t) \to D$.

4.2 Framework Overview

As shown in Figure 2, we propose a framework called Mario to retrieve relevant documents given the multi-modal conversational history (details see Section 4.3). The process begins with the system receiving the user's query as input. It then refines the query by incorporating the conversation history to generate an inferred query. Next, BM25 is applied for the first-phase retrieval, retrieving the top 100 most relevant documents. Then, we introduce a multi-modal generative re-ranking model that incorporates the inferred query to refine and re-rank the initial results. Inspired by (Yuan et al., 2024; Geigle et al., 2022), we train the model to generate keywords for the top relevant documents, leveraging both textual and visual information. By incorporating multi-modal information, the model effectively re-ranks the retrieved documents to enhance relevance.

4.3 Multi-modal Two-phase Retrieval

4.3.1 First-phase Retrieval

In the first phase of our retrieval process, we employ BM25 (Robertson and Zaragoza, 2009) to retrieve an initial set of relevant documents from the document base given the query t and conversational history context C. Since C is lengthy and might contain noise, we extract an inferred query Φ from C using GPT-4o (prompts see Appendix D). Given the inferred query Φ, the set of retrieved documents is obtained as:

$$D_{initial} = \text{BM25}(t, \Phi, D)\,, \quad (1)$$

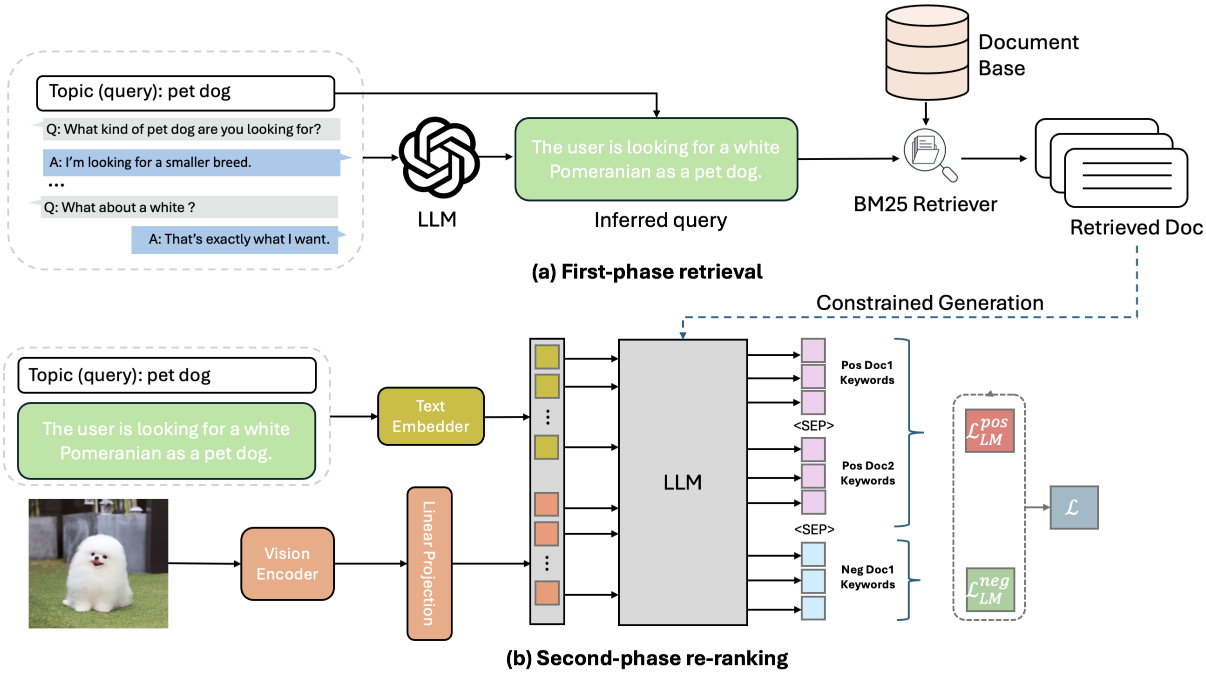

Figure 2: Overview of the Mario two-phase retrieval framework.

where D is the initial document set and $D_{initial}$ is the first-ranked result. The retrieved documents are then passed to subsequent stages for further refinement and re-ranking using multi-modal information with generative models.

4.3.2 Multi-modal Re-ranking

To integrate multi-modal information, we propose a generative re-ranking model based on a multi-modal LLM.

Image and text encoding. Our model encodes the input image I using the SigLIP (Zhai et al., 2023) vision encoder f_{img} to extract image feature $\mathbf{z}$: $\mathbf{z} = f_{img}(I)$. The image feature is then projected into the LLM's embedding space using a learned projection matrix W and concatenated with the text embedding τ, where τ is obtained from the text embedder f_{text}: $\tau = f_{text}(t, \Phi)$. The final output $\mathbf{e}$ is then computed as:

$$\mathbf{e} = f_{\text{LLM}}([W\mathbf{z}; \tau]) , \tag{2}$$

where f_{LLM} is the LLM responsible for generating the final re-ranking output.

Keyword extraction. Following previous work in generative retrieval (Tang et al., 2024; Li et al., 2023), we train the multi-modal LLM to generate a ranked sequence of document IDs. Each document d is identified by a unique keyword-based ID denoted as K_d, ensuring efficient retrieval and semantic relevance. Specifically, we extract five representative keywords per document using YAKE (Campos et al., 2020). These keywords serve as compact semantic descriptors that capture each document's core information.

Model training. We train the model to generate a ranked sequence of document IDs based on the multi-modal input x, refining the initial BM25 ranking D_{initial}. To improve the model's ability to distinguish between good and bad ranking results, we train it to generate keywords for relevant and irrelevant documents sequentially, with individual documents separated by a [SEP] token. Relevant and irrelevant samples are identified based on their overlap with the ground-truth labels in D_{initial}. For the loss function, we use a combination of the positive sample's language modeling loss and Margin Ranking Loss:

$$\mathcal{L} = \mathcal{L}_{\text{LM}}^{\text{pos}} + \lambda_{\text{rank}} \cdot \mathcal{L}_{\text{rank}} , \tag{3}$$

Here λ_{rank} is the weighting factor, and the ranking loss is defined as:

$$\mathcal{L}_{\text{rank}} = \max(0, m + \mathcal{L}_{\text{LM}}^{\text{pos}} - \mathcal{L}_{\text{LM}}^{\text{neg}}) , \tag{4}$$

where m is the margin, $\mathcal{L}_{\text{LM}}^{\text{pos}}$ and $\mathcal{L}_{\text{LM}}^{\text{neg}}$ are the language modeling loss for the relevant and irrelevant samples respectively. See detailed explanation in Appendix G.

Inference. During inference, to prevent the model from generating arbitrary tokens, we employ a constrained generation technique (Post and Vilar,

2018) to ensure that only valid keywords are selected and generated. That is, we restrict the model vocabulary to a predefined set of allowed keywords from $D_{initial}$. Specifically, at each decoding step t, let the current partial sequence be $y_{<t}$. We define the allowed set of tokens A_t as:

$$\{v \in \mathcal{V} \mid \exists z \in \mathcal{T}, \text{s.t. } y_{<t} \oplus v = \text{prefix}(z)\}\ , \quad (5)$$

where $\mathcal{V}$ is the vocabulary, $\mathcal{T}$ is the trie encoding for all valid keyword sequences, and $\oplus$ denotes sequence concatenation. By masking the probability distribution for the next token to consider only those in A_t, we ensure that the generated output adheres strictly to the allowed keywords (For detailed explanation of the constrained generation see Appendix F).

5 Experiments

5.1 Experimental Setup

We split ClariMM's facets into 80% for training, 10% for validation, and 10% for testing, and create the corresponding datasets accordingly. As a result, the training set consists of 9,688 conversations and 856 facets, while the validation and testing set each contains 672 conversations and 107 facets. To create the single-turn comparison set, we use the inferred query as input and the first turn of each conversation as input. We choose LLaVA-OneVision-7b as the base model for Mario. For retrieval evaluation, we employ Mean Reciprocal Rank (MRR), Precision (P@k), and Normalized Discounted Cumulative Gain (nDCG@k) where $k \in \{1, 3, 5\}$. The ground truth relevance documents are sourced from the TREC Web Track 2009-2012 (Clarke et al., 2009, 2012). All hyper-parameters are detailed in Appendix C. We report the performance of Oracle image selection. Our experiments are conducted using the PyTorch framework, with training and evaluation performed on one NVIDIA V100 and two NVIDIA A100 GPUs.

5.2 Compared Methods

We first adopt several uni-modal baselines by removing image information from the model input to simulate a text-only interaction.

BM25 (Robertson and Zaragoza, 2009) ranks documents based solely on the text input, without any re-ranking.

Bert (Devlin et al., 2019) re-ranks the BM25 results with Bert model. We adopt the implementation from MacAvaney et al. (2019).

T5 (Raffel et al., 2019) is trained to perform re-ranking by generating keyword sequences of relevant documents given a query. We use the `T5-base` version in our experiment.

Qwen-2 (Yang et al., 2024) ranks documents similar to T5, but uses `Qwen-2-7b` as the base model.

We also compare our method with several multi-modal baselines:

VisualBert (Li et al., 2019) is a multi-modal model with Bert structure and is trained with pairwise softmax loss for re-ranking.

VLT5 (Cho et al., 2021) takes multi-modal input and is trained to output the keyword of the documents with constrained generation.

5.3 Experimental Results

We report the performance of multiple baselines on multi-turn and single-turn settings in Table 3 and 4. We observe that language-model-based rankers such as T5 and Bert outperform the traditional lexical method, BM25. We further analyze the impact of incorporating images in the document retrieval task. Our findings indicate that using images enhances retrieval performance, particularly in multi-turn conversations, compared to models that rely solely on text. For instance, in the multi-turn case, VLT5 achieves a P@1 of 42.34%, outperforming its uni-modal counterpart T5, which records a P@1 of 41.30%. These results highlight the advantage of multi-modal information in capturing a more comprehensive user intent over longer conversational histories. However, this benefit diminishes in the single-turn scenario, where we see a 1.47% decrease in P@1 comparing Bert with VisualBert. This is because images are more likely to be misleading and have a negative impact in the first turn, as the model benefits less from visual information when contextual cues are limited. Results further show that all models perform notably better in multi-turn conversations than in single-turn ones, as added context helps capture user intent more effectively. Notably, Mario consistently outperforms the other baselines in the multi-turn and single-turn settings, achieving the highest scores across key metrics and emphasizing its superior ability to leverage contextual cues.

	Img.	MRR	P@1	P@3	P@5	nDCG@1	nDCG@3	nDCG@5
BM25	✗	50.74	39.62	36.16	36.03	25.80	23.39	24.56
Bert	✗	56.36	46.08	41.50	41.37	35.70	33.82	34.01
T5	✗	52.15	41.30	37.64	38.63	41.30	38.82	39.39
Qwen-2	✗	46.48	42.26	39.72	39.23	40.08	37.96	36.88
VisualBert	✓	56.50	46.57	46.24	44.02	35.33	36.65	36.28
VLT5	✓	53.22	42.34	38.83	39.43	42.34	39.90	40.26
Mario	✓	**59.36**	**48.10**	**47.09**	**45.48**	**46.90**	**45.77**	**43.98**

Table 3: Experimental results (%) on multi-turn conversations.

	Img.	MRR	P@1	P@3	P@5	nDCG@1	nDCG@3	nDCG@5
BM25	✗	42.94	32.07	30.81	30.37	20.39	20.15	21.02
Bert	✗	49.34	39.22	37.42	36.27	29.66	29.42	29.13
T5	✗	41.37	28.08	28.97	28.88	28.08	29.16	31.92
Qwen-2	✗	44.30	40.56	37.68	35.97	38.40	35.94	33.68
VisualBert	✓	45.95	37.75	33.50	32.55	28.43	25.83	25.20
VLT5	✓	43.18	30.46	28.92	28.94	30.46	29.69	30.42
Mario	✓	**53.24**	**46.54**	**43.48**	**40.02**	**41.85**	**39.56**	**38.68**

Table 4: Experimental results (%) on single-turn conversations.

6 Extensive analysis

6.1 Impact on different turns

We further report the retrieval performance under the different number of turns for VLT5, VisualBert, and Mario in Figure 3. As shown in the figure, VLT5 indicates only a modest improvement from 38.59 (two-turn) to 41.30 (four-turn), indicating limited gains from the additional turns. VisualBert's performance even declines as the conversation length increases, starting at 45.58 for two-turn data and dropping to 40.19 for four-turn data. This suggests that VisualBert struggles to leverage the increasing context effectively in longer conversations. In contrast, Mario demonstrates consistent and substantial improvements with each additional turn, with P@5 increasing from 43.60 (two-turn) to 48.12 (four-turn). This significant gain confirms that Mario excels in multi-turn conversational retrieval and outperforms VLT5 and VisualBert in longer interactions. This highlights the model's ability to effectively capture the evolving intent and incorporate context across turns making it particularly well-suited for handling long conversations.

6.2 Impact on different topics

We further evaluate the performance of various models on seen and unseen topics to evaluate their robustness and generalization capabilities. We re-

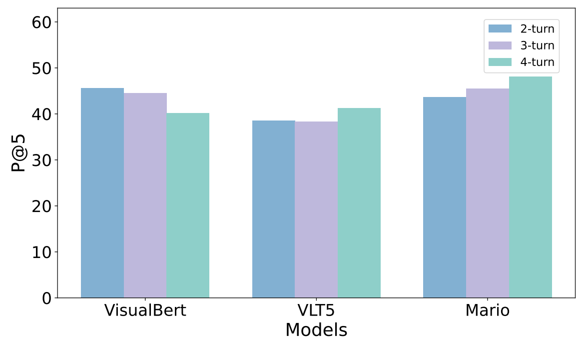

Figure 3: P@5 scores under different turn counts in ClariMM.

split the ClariMM dataset into training, unseen, and seen testing sets. The unseen testing set consists of 10% of all topics, entirely excluded from the training process. In contrast, the seen testing set includes topics that are also present in the training set. As shown in Table 6, Bert-based models (*i.e.*, VisualBert & Bert) and our model demonstrate a relatively consistent performance across both seen and unseen topics, with minimal differences in evaluation metrics. T5-based models (*i.e.*, VLT5 & T5), however, show a more significant decline between the seen and unseen sets, which suggests greater sensitivity to new topic distributions. Furthermore, we observe that the impact of using images in the unseen topics is more noticeable than in the seen topics. We can see a 4.8% increase in MRR when

Idx	Topic	Facet	Turn Num	Inferred Query	Image	Image Effect
1	Teddy bears	Find giant teddy bears	multi-turn	Looking for giant teddy bears		+0.2
			single-turn	Exploring options related to teddy bears		0
2	Hobby Stores	Where can I buy radio-controlled planes?	multi-turn	Places to buy radio-controlled planes		+0.8
			single-turn	Finding a new hobby		+0.2
3	Wilson's Disease	What are the symptoms of Wilson's disease?	multi-turn	Understanding symptoms of Wilson's disease		+0.2
			single-turn	Understanding the condition of Wilson's disease		-0.4

Table 5: Case study on Mario. A positive Image Effect indicates an increase in performance after adding the image, while a negative effect indicates a performance drop.

Method	Seen		Unseen	
	MRR	P@5	MRR	P@5
Bert	54.55	40.31	51.50	34.00
T5	53.12	34.23	38.55	24.16
VisualBert	53.53	39.46	51.85	35.17
VLT5	55.31	34.46	43.35	25.49
Mario	**58.68**	**46.17**	**57.79**	**43.23**

Table 6: Comparison between seen and unseen topics.

comparing T5 and VLT5 on unseen data, however, this difference is smaller (2.29%) on the seen domain. This suggests that incorporating visual information provides a greater advantage when dealing with unfamiliar topics.

6.3 Case study

To demonstrate the effect of adding images to the multi-turn and single-turn conversations, we perform a case study in Table 5. In most cases, including images provides valuable contextual information that enhances the model's performance. Notably, adding images in multi-turn conversations tends to have a more significant positive effect compared to single-turn cases. For example, in case 2, adding an image in the multi-turn setting improves the P@5 score by 0.8, whereas adding an image in the single-turn scenario only boosts P@5 by 0.2. However, there are instances where images can negatively impact performance. In case 3, the inferred query from the single-turn conversation focuses on understanding the condition of Wilson's disease. Unfortunately, due to the insufficiency of the inferred query, the returned image fails to align with the user's hidden intent, as it includes treatment-related information. The user is primarily interested in learning about the symptoms of this disease, not its treatment, and this image leads to a negative impact on the P@5 score. By contrast, in the multi-turn scenario, the image displays symptoms, thereby providing valuable information that enhances the model's performance.

7 Conclusion

We investigate the novel task of asking multi-modal clarifying questions in multi-turn CS systems. To enable research in this domain, we introduce a large-scale dataset named ClariMM, which contains over 13k multi-turn multi-modal interactions and 33k question-answer pairs, accompanied by 14k images. We also propose a multi-modal query clarification framework named Mario, which adopts a two-phase retrieval strategy by combining initial BM25 ranking with a multi-modal generative re-ranking model. We further compare Mario with state-of-the-art models. Experimental results show that multi-turn multi-modal interactions significantly help users refine their queries, leading to improved retrieval performance.

Limitations

Several limitations remain for future work. First, we synthetically developed our dataset from Melon, which, despite our best efforts to refine it for realism, may not fully capture the spontaneity and complexity of true user interactions. Future work could address this limitation by leveraging techniques like data augmentation or reinforcement learning from human feedback (RLHF) to bridge the gap between synthetic and natural interactions. Additionally, it remains an open question how much images truly enhance the user experience in the MMCQ task. Since the effectiveness of visual information can depend heavily on its contextual relevance and the specific user intent, our current approach might not optimally handle ambiguous or noisy visual inputs. Future work should explore methods to better integrate and disambiguate visual data to maximize their contribution to the overall user experience.

Moreover, our work primarily focuses on the retrieval task and does not explicitly address query reformulation. While inferred queries are used as part of the retrieval pipeline, an investigation into multi-modal query reformulation remains unexplored. We consider this a crucial future direction that could enhance the quality of conversational systems.

Ethical Statement

All images and user topics in our dataset are sourced from publicly available datasets, ensuring that no private or sensitive information is included. The collection and use of these resources strictly comply with the terms of use and licensing agreements set by the original dataset providers. We have diligently verified that all materials originate from public sources, conducting our research with the highest regard for data privacy and ethical integrity.

References

Mohammad Aliannejadi, Julia Kiseleva, Aleksandr Chuklin, Jeff Dalton, and Mikhail Burtsev. 2020. Convai3: Generating clarifying questions for open-domain dialogue systems (clariq). *CoRR*, abs/2009.11352.

Mohammad Aliannejadi, Julia Kiseleva, Aleksandr Chuklin, Jeffrey Dalton, and Mikhail Burtsev. 2021. Building and evaluating open-domain dialogue corpora with clarifying questions. *Preprint*, arXiv:2109.05794.

Mohammad Aliannejadi, Hamed Zamani, Fabio Crestani, and W. Bruce Croft. 2019. Asking clarifying questions in open-domain information-seeking conversations. *CoRR*, abs/1907.06554.

Pavel Braslavski, Denis Savenkov, Eugene Agichtein, and Alina Dubatovka. 2017. What do you mean exactly? analyzing clarification questions in cqa. In *Proceedings of the 2017 Conference on Conference Human Information Interaction and Retrieval*, CHIIR '17, page 345–348, New York, NY, USA. Association for Computing Machinery.

Ricardo Campos, Vítor Mangaravite, Arian Pasquali, Alípio Jorge, Célia Nunes, and Adam Jatowt. 2020. Yake! keyword extraction from single documents using multiple local features. *Information Sciences*, 509:257–289.

Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. 2021. Autoregressive entity retrieval. *Preprint*, arXiv:2010.00904.

Jiangui Chen, Ruqing Zhang, Jiafeng Guo, Maarten de Rijke, Yiqun Liu, Yixing Fan, and Xueqi Cheng. 2023a. A unified generative retriever for knowledge-intensive language tasks via prompt learning. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '23, page 1448–1457. ACM.

Jiangui Chen, Ruqing Zhang, Jiafeng Guo, Yiqun Liu, Yixing Fan, and Xueqi Cheng. 2022. Corpusbrain: Pre-train a generative retrieval model for knowledge-intensive language tasks. In *Proceedings of the 31st ACM International Conference on Information amp; Knowledge Management*, CIKM '22, page 191–200. ACM.

Xiaoyang Chen, Yanjiang Liu, Ben He, Le Sun, and Yingfei Sun. 2023b. Understanding differential search index for text retrieval. *Preprint*, arXiv:2305.02073.

Jaemin Cho, Jean-Baptiste Alayrac, Elena Buchatskaya, Ivan Laptev, Josef Sivic, Cordelia Schmid, and Joao Carreira. 2021. Vlt5: Vision-language transformers for vision-and-language tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Charles Clarke, Nick Craswell, and Ian Soboroff. 2009. Overview of the trec 2009 web track.

Charles L. A. Clarke, Nick Craswell, and Ellen M. Voorhees. 2012. Overview of the TREC 2012 web track. In *Proceedings of The Twenty-First Text REtrieval Conference, TREC 2012, Gaithersburg, Maryland, USA, November 6-9, 2012*, volume 500-298 of *NIST Special Publication*. National Institute of Standards and Technology (NIST).

Anni Coden, Daniel F. Gruhl, Neal Lewis, and Pablo N. Mendes. 2015. Did you mean a or b? supporting clarification dialog for entity disambiguation. In *SumPre-HSWI@ESWC*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *North American Chapter of the Association for Computational Linguistics*.

Gregor Geigle, Jonas Pfeiffer, Nils Reimers, Ivan Vulić, and Iryna Gurevych. 2022. Retrieve fast, rerank smart: Cooperative and joint approaches for improved cross-modal retrieval. *Preprint*, arXiv:2103.11920.

Braden Hancock, Antoine Bordes, Pierre-Emmanuel Mazaré, and Jason Weston. 2019. Learning from dialogue after deployment: Feed yourself, chatbot! *CoRR*, abs/1901.05415.

Chao-Wei Huang, Chen-Yu Hsu, Tsu-Yuan Hsu, Chen-An Li, and Yun-Nung Chen. 2023. Converser: Few-shot conversational dense retrieval with synthetic data generation. *Preprint*, arXiv:2309.06748.

Johannes Kiesel, Arefeh Bahrami, Benno Stein, Avishek Anand, and Matthias Hagen. 2018. Toward voice query clarification. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '18, page 1257–1260, New York, NY, USA. Association for Computing Machinery.

Liunian Harold Li, Haoyang Yin, Pengchuan Li, Xiaowei Hu, Lei Zhang, Lijuan Yang, Houdong Wang, and Jianfeng Gao. 2019. Visualbert: A simple and performant baseline for vision and language. In *arXiv preprint arXiv:1908.03557*.

Yongqi Li, Nan Yang, Liang Wang, Furu Wei, and Wenjie Li. 2023. Learning to rank in generative retrieval. *Preprint*, arXiv:2306.15222.

Sean MacAvaney, Andrew Yates, Arman Cohan, and Nazli Goharian. 2019. Cedr: Contextualized embeddings for document ranking. *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*.

Yuya Nakano, Seiya Kawano, Koichiro Yoshino, Katsuhito Sudoh, and Satoshi Nakamura. 2022. Pseudo ambiguous and clarifying questions based on sentence structures toward clarifying question answering system. In *Proceedings of the Second DialDoc Workshop on Document-grounded Dialogue and Conversational Question Answering*, pages 31–40, Dublin, Ireland. Association for Computational Linguistics.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. *Preprint*, arXiv:1912.01703.

Matt Post and David Vilar. 2018. Fast lexically constrained decoding with dynamic beam allocation for neural machine translation. *Preprint*, arXiv:1804.06609.

Filip Radlinski and Nick Craswell. 2017. A theoretical framework for conversational search. *Proceedings of the 2017 Conference on Conference Human Information Interaction and Retrieval*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *CoRR*, abs/1910.10683.

Hossein A. Rahmani, Xi Wang, Mohammad Aliannejadi, Mohammadmehdi Naghiaei, and Emine Yilmaz. 2024. Clarifying the path to user satisfaction: An investigation into clarification usefulness. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 1266–1277, St. Julian's, Malta. Association for Computational Linguistics.

Sudha Rao and Hal Daumé III. 2018. Learning to ask good questions: Ranking clarification questions using neural expected value of perfect information. *CoRR*, abs/1805.04655.

Stephen Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389.

Clemencia Siro, Zahra Abbasiantaeb, Yifei Yuan, Mohammad Aliannejadi, and Maarten de Rijke. 2025. Do images clarify? a study on the effect of images on clarifying questions in conversational search. In *CHIIR '25: ACM SIGIR Conference on Human Information Interaction and Retrieval, Melbourne, Australia, March 24 - 28, 2025*. ACM.

Clemencia Siro, Yifei Yuan, Mohammad Aliannejadi, and Maarten de Rijke. 2024a. AGENT-CQ: automatic generation and evaluation of clarifying questions for conversational search with llms. *CoRR*, abs/2410.19692.

Clemencia Siro, Yifei Yuan, Mohammad Aliannejadi, and Maarten de Rijke. 2024b. Agent-cq: Automatic generation and evaluation of clarifying questions for conversational search with llms. *Preprint*, arXiv:2410.19692.

Sathya Krishnan Suresh, Wu Mengjun, Tushar Pranav, and Eng Siong Chng. 2025. Diasynth: Synthetic dialogue generation framework for low resource dialogue applications. *Preprint*, arXiv:2409.19020.

Yubao Tang, Ruqing Zhang, Jiafeng Guo, Maarten de Rijke, Wei Chen, and Xueqi Cheng. 2024. Listwise generative retrieval models via a sequential learning process. *ACM Trans. Inf. Syst.*, 42(5).

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023a. Self-instruct: Aligning language models with self-generated instructions. *Preprint*, arXiv:2212.10560.

Zhenduo Wang, Yuancheng Tu, Corby Rosset, Nick Craswell, Ming Wu, and Qingyao Ai. 2023b. Zero-shot clarifying question generation for conversational search. In *Proceedings of the ACM Web Conference 2023*, WWW '23, page 3288–3298, New York, NY, USA. Association for Computing Machinery.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Huggingface's transformers: State-of-the-art natural language processing. *Preprint*, arXiv:1910.03771.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jianxin Yang, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Xuejing Liu, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, Zhifang Guo, and Zhihao Fan. 2024. Qwen2 technical report. *Preprint*, arXiv:2407.10671.

Yifei Yuan, Clemencia Siro, Mohammad Aliannejadi, Maarten de Rijke, and Wai Lam. 2024. Asking multimodal clarifying questions in mixed-initiative conversational search. *Preprint*, arXiv:2402.07742.

Hamed Zamani, Susan T. Dumais, Nick Craswell, Paul N. Bennett, and Gord Lueck. 2020. Generating clarifying questions for information retrieval. In *WWW '20: The Web Conference 2020, Taipei, Taiwan, April 20-24, 2020*, pages 418–428. ACM / IW3C2.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. 2023. Sigmoid loss for language image pre-training. *Preprint*, arXiv:2303.15343.

Jie Zou, Evangelos Kanoulas, and Yiqun Liu. 2020. An empirical study on clarifying question-based systems. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, CIKM '20, page 2361–2364, New York, NY, USA. Association for Computing Machinery.

A Dataset Creation and Prompts

We use a multi-step refinement process, as shown in Figure 4, to address the unnaturalness of synthetic data. We first prompt GPT-4o to determine if two QA convey similar information in a single conversation, then we remove entries identified as having duplicate QA structures using Prompt A in Table 8. This step helps detect and remove redundant or highly similar QAs.

Next, we prompt GPT-4o to analyze each conversation turn and identify whether the hidden facet intention is revealed prematurely using prompt B in Table 8. This prompt assesses whether the hidden facet intention is revealed too early. It judges whether a provided answer can be interpreted as the same as the facet intention. For instance, if the conversation has four turns and the hidden intention is revealed in the second turn, we extract those two turns and add them to the two-turn dataset.

As illustrated in the figure, the four-turn data undergoes the most rigorous filtering process compared to the two-turn and three-turn data, which explains its lower count in Table 2. Consequently, the amount of available data decreases as the number of turns increases because, in most cases, the intention is revealed prematurely.

Finally, we introduce an additional refinement step using Algorithm 1 to ensure the conversational flow is as realistic as possible. In this algorithm, we employ three prompts, Θ_{initial}, Θ_{partial}, and Θ_{final}, using 2-shot learning. In Table 8, we show that these prompts iteratively refine responses to gradually unveil the hidden intent to effectively simulate the natural progression of the clarification phase. The entire refinement process is performed only once before training.

This approach is aligned with other recent efforts in the community where high-quality synthetic datasets are constructed using LLMs like AGENT-CQ (Siro et al., 2024b), Self-Instruct (Wang et al., 2023a), CONVERSER (Huang et al., 2023) and DiaSynth (Suresh et al., 2025).

B Quality Control Metric

The following metrics were used to assess the quality of ClariMM during human evaluation: **Relevance**: Each turn's alignment with the original topic (assessed per turn); **Coherence**: Logical flow between combined QA pairs (assessed per dialogue); **Diversity**: Variation in responses and avoidance of redundancy (assessed per dialogue);

and **Intent reveal**: Effectiveness of progressive intent revelation (assessed per dialogue). **Average Image Relevance**: The alignment of each turn's images with the question and answer. We take the average score for 3 images per turn.

C Hyperparameter Settings

Our code is based on PyTorch (Paszke et al., 2019) and Huggingface Transformers (Wolf et al., 2020). For Llava-OneVision, we use the 7b pretrained version, 1e-4 as the learning rate and 2 for the batch size. For the loss function, we set the margin to 2.0 and λ_{rank} to 0.75. For generation, we set the number of beams to 10. For first-phase document retrieval, we retrieved the top 100 documents using BM25.

D Inferred Query Extraction

To capture the user's intent from a multi-turn conversation, we employ a summarization step using GPT-4o that focuses on what the user is actually interested in. It compresses the conversation into a short "inferred query" discarding irrelevant details such as off-topic remarks. By isolating only the essential user request, the system can more effectively guide subsequent retrieval, ensuring that the user's primary goal remains at the forefront. This step is performed only once on the dataset before training to generate inferred queries, which are then used during training.

Prompt
Extract the user's intent based on the conversation. Only mention what they are interested in. Conversation: {conversation}

Table 7: Prompts used for dataset creation.

E Additional Quality Control

Two other authors independently evaluated 30 samples from the dataset to further assess its quality. Table 9 shows the human evaluation scores. As shown, the results are consistent with our original findings, demonstrating strong performance in relevance, diversity, and intent reveal across turns.

Metric	**Mean**	**Std Dev**	**Median**
Relevance (Turn 1)	3.5	0.63	3.00
Relevance (Turn 2)	3.17	0.65	3.00
Relevance (Turn 3)	3.67	0.71	4.00
Relevance (Turn 4)	3.97	0.72	4.00
Coherence	3.83	0.95	4.00
Diversity	3.37	0.91	4.00
Intent reveal	4.53	0.63	5.00

Table 9: Additional human evaluation scores for relevance, coherence, diversity, and intent reveal.

To assess the multi-modal grounding quality of our dataset, they also evaluated the relevance of the images presented in each conversational turn. Since each turn included three images, we report the average image relevance per turn. Ratings were collected on a 5-point scale, similar to before, across 30 data samples. Table 10 summarizes the mean, standard deviation, and median of the average image relevance scores for each turn. The helpfulness of images is studied in section 6.3.

Metric	**Mean**	**Std Dev**	**Median**
Avg Image Relevance (Turn 1)	4.50	0.33	4.50
Avg Image Relevance (Turn 2)	4.13	0.21	4.00
Avg Image Relevance (Turn 3)	4.89	0.18	5.00
Avg Image Relevance (Turn 4)	5.00	0.00	5.00

Table 10: Human evaluation scores for average image relevance.

F Constrained Generation

In our framework, each document is uniquely represented by a sequence of five keywords, which serve as its compact semantic identifier. During inference, the model must generate this exact sequence to successfully retrieve the corresponding document from our document pool. If unconstrained generation were used, the model could either mix keywords from different documents, producing sequences that don't correspond to any real document from the pool, or generate tokens outside the keyword vocabulary, which would result in invalid or out-of-scope outputs. To prevent these issues, we employ constrained generation via a TRIE-based decoding mechanism, which ensures that only valid keyword sequences (those corresponding to actual documents in the collection) are generated. This guarantees that the model's outputs remain semantically meaningful and retrievable, which is critical for the effectiveness of the re-ranking step.

Our approach aligns with prior work in generative retrieval, such as GenRE (Cao et al., 2021)

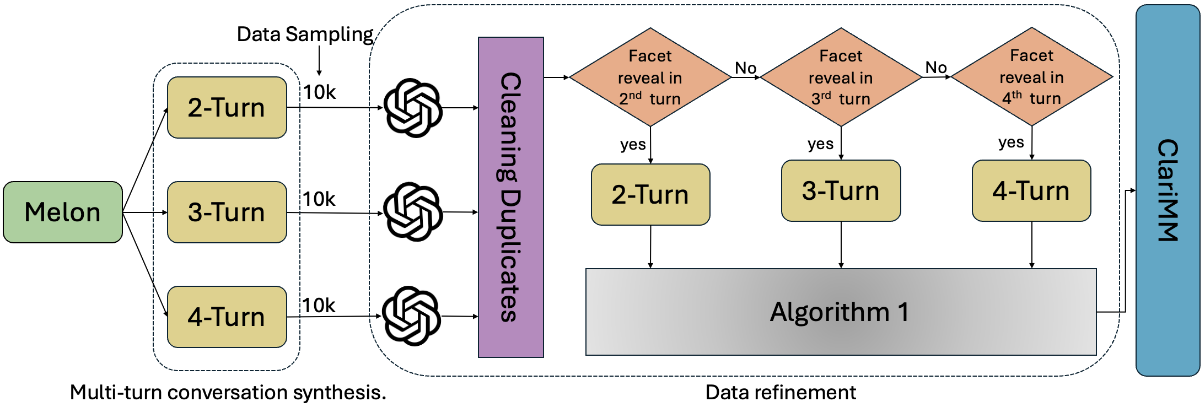

Figure 4: Dataset creation pipeline.

Type	Prompt Content
Prompt A	I will provide you with two pairs of questions and answers. Determine if these two question-answer pairs contain similar information. Output "yes" or "no" and explain why. Question 1: {question1} Answer 1: {answer1}, Question 2: {question2} Answer 2: {answer2}
Prompt B	I will provide you a pair of question-answer and a facet (user's hidden intention). Judge whether the answer aligns with the facet intention. If yes, generate: "intention reached". Facet intention: *facet_intention*, Question: *question*, Answer: *answer*
Prompt Θ_{initial}	**Examples:** Example 1: **Facet:** How to fix a car engine. **Question:** Do you want to buy a car? **Answer:** No, I am not looking to buy a car. Example 2: **Facet:** Find coffee shops near me. **Question:** Would you like to make a cup of coffee? **Answer:** No, thank you, I want to buy one. I provided you with some examples above. Now, modify the following answer so that it is connected to the question and doesn't reveal the hidden intention of the facet like in the examples. Ensure your answer doesn't violate the facet. **Prompt:** Imagine you are a user answering an agent question. Modify this answer without revealing any hidden intention of the facet and without violating the facet. Facet: {facet}, Question 1: {question1}, Answer 1: {answer1}, {examples}
Prompt Θ_{partial}	**Examples:** Example 1: **Facet:** The user wants to buy a red car. **Question:** Are you looking for a specific color? **Answer:** I am considering a color, but I haven't decided fully yet. Example 2: **Facet:** I'm looking for the car-part.com website. **Question:** Do you want to sell used car parts? **Answer:** For now, I am mainly focused on finding a website. I provided you with some examples above. Now, modify the following answer to reveal only a partial abstract of the hidden intention (facet) and hint at the user's interests without revealing the full intention **Prompt:** Imagine you are a user answering an agent question. Modify the following answer to reveal only a partial abstract of the hidden intention (facet). Do **NOT** reveal the full hidden intention. Facet: {facet} Question 3: {question2} Answer 3: {answer2} {examples}
Prompt Θ_{final}	**Examples:** Example 1: **Facet:** The user wants to buy a red car. **Question:** Are you looking for a specific color? **Answer:** Yes, I am looking for a red car to buy. Example 2: **Facet:** I'm looking for the car-part.com website. **Question:** Do you want to sell used car parts? **Answer:** No, I am just looking for the car-part.com website. I provided you with some examples above. Now, modify the following answer to fully reveal the hidden intention in a clear and direct manner, and ensure that the answer reflects the facet without ambiguity. **Prompt:** Imagine you are a user answering an agent question. Modify the following answer to fully reveal the hidden facet. Ensure that the answer clearly reflects the facet. Facet: {facet}, Question 3: {question3}, Answer 3: {answer3}, {examples}

Table 8: Prompts used for dataset creation.

and CorpusBrain (Chen et al., 2022), and we use constrained generation to tightly couple generation and retrieval. These works emphasize that without such mechanisms, generation-based retrieval risks semantic drift and retrieval failures. By integrating constraints, we ensure that the generation process remains grounded in the structure of the document collection.

G Loss Function

To train our generative retriever effectively, we adopt a hybrid loss function that combines a standard language modeling loss with a margin-based ranking loss. This combination encourages the model not only to generate correct outputs but also to assign higher confidence to relevant targets compared to irrelevant ones.

The total loss is defined as:

$$\mathcal{L} = \mathcal{L}_{\text{LM}}^{\text{pos}} + \lambda_{\text{rank}} \cdot \mathcal{L}_{\text{rank}} \,, \tag{6}$$

where $\mathcal{L}_{\text{LM}}^{\text{pos}}$ is the language modeling loss for the positive (i.e., relevant) sample, $\mathcal{L}_{\text{rank}}$ is a margin-based ranking loss, and λ_{rank} is a hyperparameter controlling the strength of the ranking component.

Language Modeling Loss. We formulate the generation task as conditional language modeling. The loss is computed as:

$$\mathcal{L}_{\text{LM}} = -\sum_{t=1}^{T} \log P_\theta(y_t \mid y_{<t}, x) \tag{7}$$

where x denotes the input, y_t is the t-th token of the target sequence, $y_{<t}$ is the prefix sequence before step t, and θ represents model parameters. This objective guides the model to produce the correct sequence token-by-token, such as a document identifier, keyword, or entity name relevant to the input.

Margin Ranking Loss. To improve the model's discriminative ability, we add a ranking loss that enforces a margin between the losses of relevant and irrelevant targets:

$$\mathcal{L}_{\text{rank}} = \max(0, m + \mathcal{L}_{\text{LM}}^{\text{pos}} - \mathcal{L}_{\text{LM}}^{\text{neg}}) \tag{8}$$

Here, $\mathcal{L}_{\text{LM}}^{\text{neg}}$ is the language modeling loss on a negative (irrelevant) target, and m is a fixed margin. This encourages the model to generate lower loss (i.e., higher confidence) for positive samples compared to negative ones by at least m. If the margin is not met, a penalty is applied. This design improves ranking fidelity among retrieved candidates.