Retrieving Intent-covering Demonstrations for Clarification Generation in Conversational Search Systems

Ziliang Zhao
Gaoling School of Artificial Intelligence
Renmin University of China
Beijing, China
zhaoziliang@ruc.edu.cn

Changle Qu
Gaoling School of Artificial Intelligence
Renmin University of China
Beijing, China
changlequ@ruc.edu.cn

Zhicheng Dou[*]
Gaoling School of Artificial Intelligence
Renmin University of China
Beijing, China
dou@ruc.edu.cn

Haonan Chen
Gaoling School of Artificial Intelligence
Renmin University of China
Beijing, China
hnchen@ruc.edu.cn

Jiajie Jin
Gaoling School of Artificial Intelligence
Renmin University of China
Beijing, China
jinjiajie@ruc.edu.cn

Abstract

Search clarification is a critical user interface for open-domain conversational Web search, where generating high-quality facets for ambiguous or multi-facet queries significantly guides disambiguation and enhances the user's interaction experience. Recently, in-context learning with Large Language Models (LLMs) has emerged as a promising approach for facet generation by leveraging static or similarity-based demonstrations as prompts. However, existing methods predominantly rely on query similarity, failing to account for the multi-dimensional nature of query intents. This limitation can lead LLMs to generate incorrect or suboptimal facets misaligned with user needs. To address this challenge, we propose an intent-covering framework that improves clarification facet generation by selecting demonstrations that comprehensively cover the diverse intents underlying a given query. Specifically, we first train a generative model with beam search to predict potential intents and construct an intent-document graph to capture their semantic relationships. We then introduce a heuristic greedy algorithm that optimizes demonstration selection by maximizing intent coverage. Furthermore, since the order of demonstrations significantly affects generation quality, we develop a re-ranking model to optimize their sequence for better contextual alignment. Experiments demonstrate the superiority of our approach over strong baselines in various lexical and semantic evaluation metrics. Additionally, we conduct an in-depth analysis of how the number, order, and contextual relevance of demonstrations influence generation performance.

CCS Concepts

• **Information systems → Search interfaces**.

Keywords

Search Clarification, In-context Learning, Conversational Search

ACM Reference Format:
Ziliang Zhao, Changle Qu, Zhicheng Dou, Haonan Chen, and Jiajie Jin. 2025. Retrieving Intent-covering Demonstrations for Clarification Generation in Conversational Search Systems. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2 (KDD '25), August 3–7, 2025, Toronto, ON, Canada.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3711896.3737104

1 Introduction

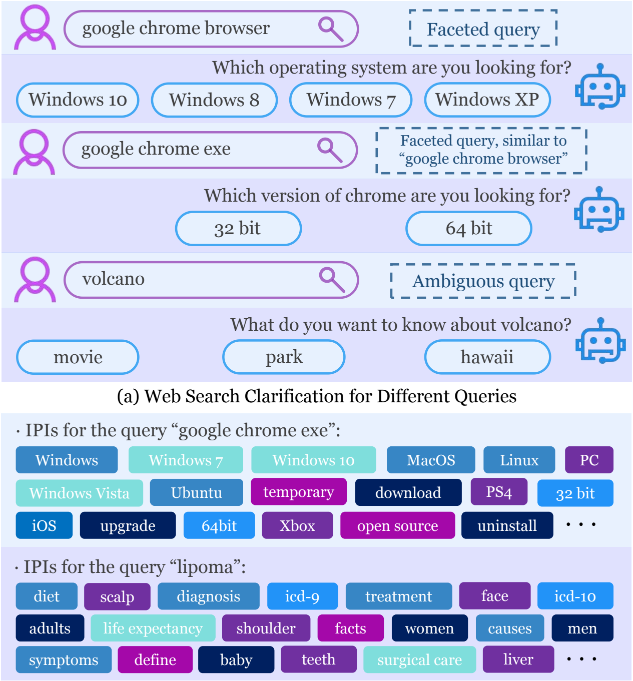
Search clarification is a fundamental component of conversational search and Web information retrieval (IR) systems [1, 3, 29, 49, 56]. When users issue ambiguous or faceted queries, the system can proactively generate clarifying questions and suggest candidate facets to refine the search intent. For instance, as illustrated in Figure 1(a), for the faceted query "Google Chrome browser", facets such as "Windows 10", "Windows 7", and "Windows XP" specify different operating systems. Similarly, for the ambiguous query "volcano", the system seeks to clarify whether the user is referring to the movie Volcano or a geographical entity like Volcano Park. Notably, even semantically similar queries can correspond to distinct user intents, underscoring the necessity of robust facet generation techniques. These facets not only serve as potential queries that users may resubmit but also play a critical role in intent exploration. Moreover, they enhance user engagement in conversational search interfaces [35] and improve search result quality through diversification [39], personalization [44], and exploratory search [25, 28].

Existing studies have explored various approaches for generating clarification aspect facets, demonstrating significant effectiveness. Early methods include rule-based approaches [7, 8] and machine-learning-based models [14, 15], which extract query facets from search result pages using manually crafted rules. With the advent of pre-trained language models (PLMs), several PLM-based methods

[*] Zhicheng Dou is the corresponding author.

KDD '25, August 3–7, 2025, Toronto, ON, Canada

ACM ISBN 979-8-4007-1454-2/2025/08
https://doi.org/10.1145/3711896.3737104


google chrome browser
Faceted query
Which operating system are you looking for?
Windows 10
Windows 8
Windows 7
Windows XP
google chrome exe
Faceted query, similar to "google chrome browser"
Which version of chrome are you looking for?
32 bit
64 bit
volcano
Ambiguous query
What do you want to know about volcano?
movie
park
hawaii

(a) Web Search Clarification for Different Queries

(b) Independent Potential Intents (IPIs) for Different Queries

Figure 1: Examples of query facets and our defined Independent Potential Intents (IPIs) of a query. The same color of IPIs means that they share the same topic.

have been introduced for facet generation, including NMIR [10], PINMIR [11], and BART-based models [27, 35]. More recently, Large Language Models (LLMs) have been leveraged for aspect facet generation through In-Context Learning (ICL) [26, 38], capitalizing on their extensive open-domain knowledge and strong intent comprehension capabilities. Notable examples include GPT-3 [35] and GPT-3.5 Turbo [27], which typically generate high-quality facets by incorporating static or similar demonstrations in the form of (*query*, *facets*) within a carefully constructed natural language prompt. These advancements highlight the growing role of LLMs in refining conversational search by improving intent disambiguation and enhancing user interaction quality.

However, using LLMs for facet generation with static in-context demonstrations often underperforms fine-tuned PLM-based models, as these demonstrations are not tailored to the specific query [27, 35]. As a result, LLMs primarily learn the format and structure of facet generation rather than adapting contextually to query-specific semantics, leading to mismatches between generated and ground-truth facets. To mitigate this issue, recent studies have explored demonstration selection strategies based on term- or semantic-level similarity [23], as well as diversity-based selection [17], in various natural language generation tasks. Additionally, some approaches focus on training retrievers to select and rank demonstrations based on LLM performance [21, 34]. However, it is particularly challenging that retrieving query-similar demonstrations does not guarantee facet alignment, as query facets often exhibit a **multi-dimensional nature**. Consequently, the selected demonstrations may still **contain irrelevant facets**, leading to suboptimal generation quality.

We argue that the multi-dimensional nature of query facets is a key factor contributing to inaccuracies in LLM-generated predictions. A single query can correspond to multiple valid facet sets, yet evaluation datasets typically consider only one as ground truth. For example, as shown in Figure 1(b), the query "google chrome exe" can be refined along different dimensions: (1) [64-bit, 32-bit], (2) [Windows 7, Windows 8, Windows 10, Windows XP], and (3) [update, download, install, fix]. However, in MIMICS dataset [50], only the first dimension is treated as ground truth because **search clarification prioritizes most likely follow-up intents**—users searching "google chrome exe" are more likely to refine it with "32-bit" or "64-bit" than with other facets. Given this multi-dimensional nature, we propose that **demonstrations should maximize the coverage of potential query intents to encompass as many ground-truth facets as possible**. To achieve this, we introduce Independent Potential Intents (IPIs), a structured representation of a query's independent possible facets. These IPIs contain the ground-truth facets while maintaining diversity. As shown in Figure 1(b), Windows, install, and 64-bit are all IPIs of "google chrome exe", but they may not co-occur in a single ground-truth facet set due to coherency constraints [56]. IPIs thus serve as an effective foundation for generating coherent and comprehensive facet groups [54].

Given the criteria for demonstration selection to cover user intents, two key challenges arise: **obtaining IPIs** and **designing a selection algorithm**. To address the first challenge, we train a Seq2Seq intent prediction model, which differs from existing PLM-based approaches that directly generate facet groups. Instead, our model takes a query and relevant document snippets as input and outputs a single independent IPI per inference step. To obtain multiple IPIs, we apply beam search to extract the top-k IPIs with highest probabilities for each query. Based on the predicted IPIs, we introduce **RDS** (Rule-based Demonstration Selection), a heuristic algorithm that greedily selects optimal demonstrations by maximizing IPI coverage while leveraging co-occurrence relationships within top search result snippets. Additionally, recognizing that demonstration order significantly impacts generation quality, we propose **DRM** (Demonstration Re-ranking Model) to refine the sequence of selected demonstrations, further improving facet generation effectiveness.

Following previous studies, we obtain MIMICS-Click [50] as the demonstration set and apply MIMICS-Manual for evaluation. We implement various kinds of PLM- and LLM-based methods as baselines. Furthermore, we also compared our proposed methods with three common demonstration retrieval methods: random selection, selection with BM25 similarity, and selection with SBERT similarity. The experimental results show that, first, our proposed heuristic method RDS shows significant improvement compared with all PLM-based and LLM-based baseline methods. The re-ranker DRM further improves the generation effectiveness by rearranging the order of the demonstrations. We further conduct additional experiments to analyze (1) whether the input including in-context documents influences the experimental results, (2) how the generation effectiveness varies with the demonstration number, and (3) how the demonstration order affects the experimental results.

The contributions of this work include:

- We introduce a novel demonstration selection criterion for Web search clarification, maximizing IPI coverage for a given query.

- We fine-tune a BART model to generate IPIs and design a heuristic method (RDS) along with an effective re-ranking model (DRM) to optimize demonstration retrieval and ordering.
- Our approach achieves state-of-the-art performance on MIMICS, outperforming existing PLM- and LLM-based baselines.

2 Related Work

Clarification for Conversational Search. Aliannejadi et al. [1] first introduced the idea of clarifying questions in conversational search systems by selecting questions to elicit user responses. However, in Web search systems, query complexity poses a significant challenge, as a limited number of questions cannot capture the broad range of user intents. More recently, Zamani et al. [49, 51] highlighted the critical role of clarification in Web search and introduced the MIMICS dataset [42, 50], which facilitates research on automatically generating clarifications. In this setting, clarifying questions [45] and facet-based aspect items [7, 14, 15] are dynamically generated rather than selected [1], or predefined by rules as in conversational recommender systems [19]. This distinction reinforces the open-domain nature of search clarification. Meanwhile, domain-specific clarification methods [30, 31, 43, 47] have demonstrated effectiveness in question-answering systems, yet they remain inadequate for the vast diversity of Web search queries.

Other Clarification Scenarios. Clarification extends beyond Web search to community question-answering (CQA) platforms like StackExchange [30, 31] and Conversational Recommender Systems (CRS), which aim to infer user preferences through multi-turn interaction. Sun and Zhang [40] first formalized CRS by identifying key challenges. Subsequent studies have explored strategies to improve CRS by leveraging reinforcement learning, retrieval-based methods, and hybrid recommendation [2, 4, 16, 19, 22, 48, 52, 57, 58]. Beyond mainstream open-domain and closed-domain clarification scenarios, search clarification has also been applied to interactive classification tasks and multi-turn image guessing games such as the 20-Questions task [37, 46, 48, 53]. These emerging applications highlight the growing importance of adaptive clarification mechanisms across different domains.

Retrieving Demonstrations for LLMs. In-Context Learning (ICL) has become a cornerstone prompting strategy for guiding Large Language Models (LLMs) to generate high-quality outputs. In early studies, Liu et al. [23] demonstrated that retrieving semantically similar demonstrations significantly outperforms random or static selection. Consequently, most ICL-based approaches benchmark against random selection and similarity-based retrieval [6, 9, 20]. Beyond these fundamental strategies, several refinements have been proposed for task-specific optimization. For instance, Levy et al. [17] introduced a diversity-driven selection strategy in semantic parsing, ensuring that demonstrations cover a broad spectrum of potential outcomes. Kim et al. [13] proposed an LLM-generated demonstration strategy, where the model itself generates exemplars prior to inference. Additionally, demonstrations can be retrieved using a well-trained retriever, further improving selection quality and task performance [21, 34]. These advancements underscore the importance of intelligent demonstration retrieval in enhancing LLM effectiveness across various NLP tasks.

3 Methods

As discussed in Section 2, various ICL methods have been used to enhance LLM generation. However, without considering the query's multi-dimensional potential intents, LLMs can be misled by irrelevant metrics like semantic similarity [23] or diversity [17], leading to irrelevant facet generation. Based on this, we propose that demonstrations should cover as many high-probability potential intents as possible. To achieve this, we define Independent Potential Intents (IPIs) as a list of possible query facets and introduce a Seq2Seq model, BART-IPI, to obtain them. We then design a heuristic method RDS to select demonstrations from a demonstration pool by considering the coverage of IPIs and the co-occurrence relation between IPIs. Although RDS exceeds all PLM- and LLM-based baseline methods for facets generation, we also discover that the demonstration order influences the generation quality effectively. However, RDS only focuses on covering the IPIs greedily, which cannot provide a good demonstration order. Based on our observation, we further train a re-ranker DRM based on the LLM's feedback to re-rank the demonstrations for better generation.

3.1 Task Formulation

Our target is to predict a set of facets for a given user query. A **query** represents the user's information needs. When the query is sometimes ambiguous or faceted [49], each **facet** can correspond to a potential intent that help the user reformulate her **intents**, as shown in Figure 1(a) and 2(a).

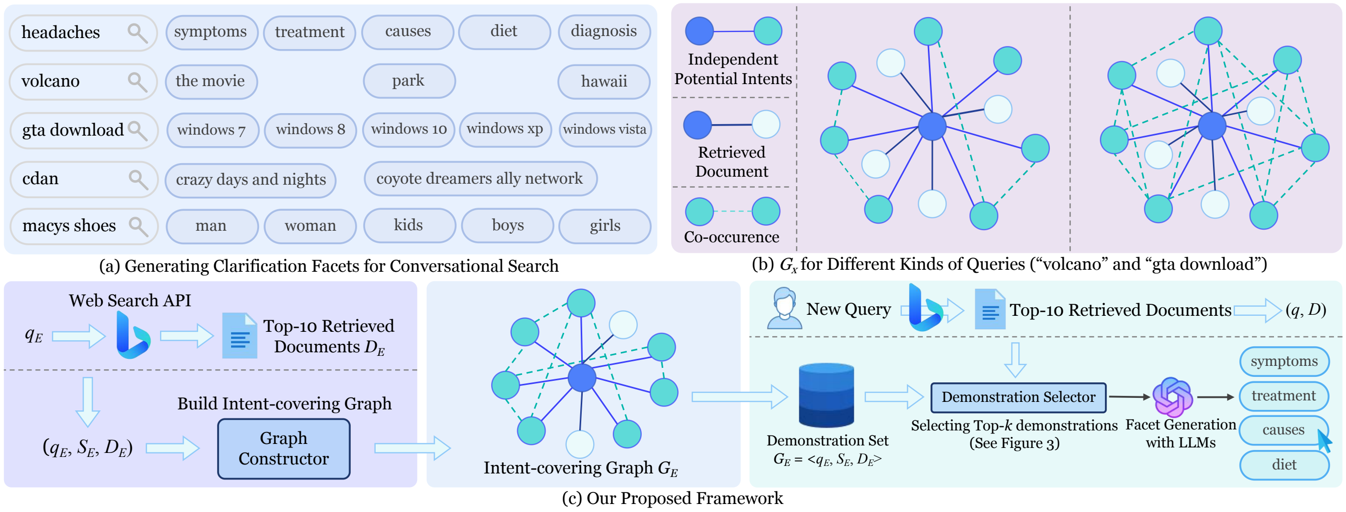
Given a training set $T = (q_i, S_i)_{i \in [1,N]}$ containing query-facets pairs and a test query q, our goal is to select k pieces of examples from T as $E = (q_i, S_i)_{i \in [1,k]}$, where $k << n$. These demonstrations are filled into a natural language prompt as P which is then concatenated with the test query $[P; q]$ as the pre-context of an LLM. Finally, the LLM outputs the predicted facets set $\hat{S}$. Figure 2(c) shows the whole process. The task can be formulated as:

$$E = R(q) = [(q_i, S_i,)]; \;\; P = prompt(E); \;\; \hat{S} = L([P; q]), \quad (1)$$

where $R(\cdot)$ is a demonstration selection method to select k pieces of demonstration, $prompt(\cdot)$ is the process of forming a natural language prompt automatically, and $L(\cdot)$ is an LLM.

3.2 BART-IPI

Web search clarification aims to generate a clarifying question Q and several candidate facets S given an ambiguous or faceted query q. These facets form a high-coherency set representing an aspect that the user may be interested in [51]. However, existing methods are difficult to generate a coherent set S in one try [35, 55]. Therefore, for either PLM-based methods or some LLM-based ICL methods [27, 35], the effectiveness is still limited. To solve this problem, we propose that we can learn to generate only a single facet at a time, and then use sampling strategies like *beam search* to sample and select the top k facets with the highest probability [54], as shown in Figure 1(b). It can be seen that the generated beams are very likely to contain ground truth facets, but unlike generating a set of S at once, the independent facets generated in this way have no *grouping information*. In fact, according to the co-occurrence information in MIMICS, we can put each facet into a different group [54]. For example in Figure 1, the same-color facets are prone

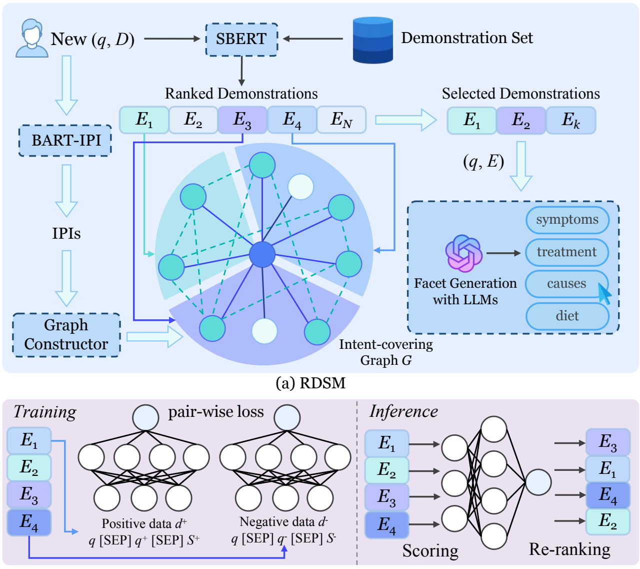


(a) Generating Clarification Facets for Conversational Search

(b) G_x for Different Kinds of Queries ("volcano" and "gta download")

(c) Our Proposed Framework

Figure 2: In this figure, we show (1) several example of search clarification generation, (2) query-facet graphs for different kinds of queries, and (3) the inference flow of our framework.



(a) RDSM

(b) SDRM

Figure 3: The framework of our proposed RDS and DRM.

to be in a good clarification pane. For the query "google chrome exe", different operating systems are in the same group, and different Windows version operating systems are in another group as well.

Even in the absence of explicit grouping information, the top-k candidate facets generated via beam search are highly likely to contain ground-truth facets [54]. These independently generated facets, with high recall, serve as valuable candidates for identifying the most relevant facets for a given query. In this paper, we define these independent facets as **Independent Potential Intents (IPIs)** that are highly likely to include ground-truth facets of the query. To generate IPIs, we designed a Seq2Seq model BART-IPI based on BART [18]. Unlike previous approaches that train BART to generate entire facet sets in a single pass [35, 55], we split the $(q, D \rightarrow S)$ data into multiple $(q, D \rightarrow s)$, each s representing a facet in S. In this way, for 400k pieces of $(q, D \rightarrow S)$ data in the MIMICS data set, after splitting, we get 1.25M pieces of $(q, D \rightarrow s)$ data. We use 90% of this data to train the BART-IPI and use 10% for validation. At inference, we apply beam search to retrieve the top-k highest-probability IPIs. Specifically, the BART-IPI model is trained to minimize the following objective function:

$$L = -\frac{1}{N}\sum_{i=1}^{N} \log P(f_\theta(x_i)|f_\theta(x_{<i})) \tag{2}$$

where $f_\theta(\cdot)$ is the BART-IPI model parameterized by θ.

Additionally, we apply data processing to mitigate noise in the MIMICS dataset. To address singular-plural inconsistencies, we use NLTK to normalize plural facets to their singular forms, ensuring the facets' consistency. Next, we deduplicate the generated facets. We then incorporate local and global statistics: facets are weighted based on document frequency and facet occurrence frequency in MIMICS-Click (the training dataset). After processing, we generated the corresponding top-100 IPIs for each test query in the MIMICS-Manual dataset. By conducting statistical analysis, we find that 51.7% of the ground truth facets in MIMICS-Manual appear in the top-50 high-probability IPIs corresponding to the query, significantly outperforming the Recall values of existing models evaluated on the MIMICS dataset.

3.3 RDS

Given the predicted IPIs for a query, we propose a heuristic **Rule-based Demonstration Selection (RDS)** method that iteratively traverses the demonstration candidate set to greedily maximize coverage of the IPIs and their co-occurrence relations. Specifically, **first**, for a given query q, we construct a graph G comprising nodes including query q, its top-retrieved document snippets D, and its predicted IPIs I, as illustrated in Figure 2. To capture potential semantic dependencies, we introduce edges between IPI nodes whenever two IPIs co-occur within the same document. This graph encodes both the possible user intents underlying the query and the relational structure among these intents.

Next, since semantic similarity is deemed crucial in demonstration selection [17], we rank these demonstrations based on $\text{sim}((q \text{ [SEP] } D), (q_E \text{ [SEP] } D_E))$ where q is the test query and q_E is one of the demonstration queries. D and D_E means the corresponding document snippets of q and q_E retrieved by Bing search engine API respectively. We use BERT-base to get the embedding of the text. All demonstration candidates for the query q are then sorted according to this similarity score.

After that, we select top-N demonstrations and iterate these demonstrations. In this paper, we set $N = 100$. Our aim is to select demonstrations that **cover as many IPIs as possible**. Specifically, for the i-th demonstration, if the demonstration E_i covers more IPIs than one demonstration E_x that has been selected, then we replace E_x with E_i. On the other hand, if a new demonstration covers the same IPIs but covers more edges in G, we also replace the old demonstration with the new one. Since in each iteration, various demonstrations can satisfy to be selected, we record all satisfying demonstrations and fetch the one in which *the facets have the highest average position in the IPIs of q*. This process is iterated until k demonstrations are selected. In brief, we try to find k demonstrations that can maximize the coverage of the edges and nodes in graph G. An illustration can be found in Figure 3(b) and the whole detailed process of RDS is shown in Algorithm 1.

3.4 DRM

The order of demonstrations significantly impacts generation effectiveness. However, RDS, introduced in Section 3.3, lacks the capability to further modify the ranking of selected demonstrations, instead relying on a greedy approach to cover the query-IPI graph G. To address this limitation, we propose re-ranking the selected demonstrations. To validate the importance of demonstration order, we conduct an experiment using 4 demonstrations per query, generating 24 possible permutations for each. By inputting these permutations into the LLM and evaluating the resulting facet predictions using the term match F1 metric, we identify the best and worst permutations for each query. The aggregated results across all test queries, presented in Table 1 as RDS_{min} and RDS_{max}, demonstrate the substantial impact of demonstration ordering.

Our analysis focuses on two key metrics: term overlap F1 and exact match F1. The results reveal that RDS (0.1861, 0.1700) significantly outperforms RDS_{min} (0.1074, 0.1007) but underperforms RDS_{max} (0.2838, 0.2559). This indicates two critical findings: First, demonstration order substantially impacts facet generation effectiveness, with a performance gap of approximately 0.18 in term overlap F1 and 0.15 in exact match F1 between optimal and worst arrangements. Second, RDS's current implementation leaves room for improvement as it lacks consideration for demonstration ordering. Motivated by these observations, we propose the supervised **Demonstration Re-ranking Model (DRM)**, which processes queries and facets from demonstrations as input, learns scoring patterns, and incorporates LLM feedback through pair-wise training.

Algorithm 1: RDS

```
input  : Query $q$, Demonstration number $k$
output : A set of facets $S$
- Get top-retrieved snippets $D$ with Bing search API;
- Generate IPIs $I$ using BART-IPI;
- Build intent-covering graph $G$ for the query $q$;
- Obtain demonstration set $G_E =< q_E, S_E, D_E >$;
- Rank demonstrations with BERT similarity, select top-100;
- Set selected demonstration set $E = []$, facets set $F = \{\}$;
for $i \leftarrow 1$ to $len(G_E)$ do
    if Count($G_E[i], I$) == 0: Continue;
    else
        if Count($G_E[i], F$) == $len(G_E[i])$: Continue;
        for $j \leftarrow 1$ to $len(E)$ do
            if Count($G_E[i], E$) == $len(G_E[i])$: Continue;
        for $j \leftarrow 1$ to $len(E)$ do
            if Count($E[j], I$) < Count($G_E[i], I$)
                $G_E[i]$.replace($E[j]$); $update(F)$;
        for $j \leftarrow 1$ to $len(E)$ do
            if Count($E[j], I$) == Count($G_E[i], I$) and
             Edge($E[j], I$) < Edge($G_E[j], I$)
                $G_E[i]$.replace($E[j]$); $update(F)$;
        foreach $e$ in $E$ do
            if Count($G_E[i], F$) == 0
                $E$.append($G_E[i]$); $update(F)$;
- Build a prompt and apply the LLM to generate the facets;
```

To this end, we first sample 25k pieces of data from MIMICS-Click [50]. We then get the top-4 demonstrations using the BERT similarity for each query. After that, we utilize 24 kinds of permutations of demonstrations to generate 24 prompts for a query. We calculate the Term Match F1 score over the 24 results, and select the permutation with the highest score as the optimal sequences. Then, we use the optimal sequences generated by these 25,000 queries to train a pair-wise ranking model. We further fetch 5,000 pieces of randomly sampled data from MIMICS dataset [50] as the evaluation set. The training process is to optimize the pair-wise loss:

$$L_p = -\log(\text{Sigmoid}(s(d^+) - s(d^-))) \tag{3}$$

where $s(\cdot)$ is an encoder to score the demonstration d composed of the issued query q, the demonstration query q_E, and the demonstration facets S_E in the form of "q [SEP] q_E [SEP] S_E" as shown in Figure 3. We apply BERT-base as the encoder. After training the ranking model, we apply point-wise strategy to inference the score of each demonstration in MIMICS-Manual for evaluation.

4 Experiments

4.1 Data

In this paper, we focus on Web search clarification, one of the most prominent open-domain search clarification scenarios. To the best of our knowledge, MIMICS [42, 50] is the only publicly available dataset for training and evaluation in this domain. Following prior studies [10, 11, 35, 55], we use MIMICS-Click for training (demonstration pool for selecting in-context examples in this paper) and MIMICS-Manual for evaluation. Each instance in MIMICS consists of a query q, a set of facets S, and a clarifying question Q. Additionally, we retrieve the top-10 search result snippets from Bing[1] as contextual documents D for each query. Furthermore, we train BART-IPI on MIMICS-Click using the mapping $(q, D) \rightarrow S$ and perform inference on MIMICS-Manual with the trained model.

4.2 Evaluation Metrics

We evaluate generation quality by comparing the lexical and semantic similarity between generated and ground-truth facets. The ground-truth facets in the MIMICS dataset are based on real-world search logs, reflecting users' actual needs. Therefore, although facet selection is not directly associated with more intuitive metrics like search accuracy or user satisfaction, we argue that evaluating facets can serve as an indirect but meaningful reflection of these metrics [35, 36]. Following previous studies [10, 11, 35], we use three lexical similarity metrics: (1) Term overlap (P, R, and F1) to measure term-level similarity, (2) Exact match (P, R, and F1) to assess exact matches between generated and ground-truth items, and (3) Set BLEU (1 to 4) to calculate n-gram overlap. Additionally, we apply (4) Set BERT (P, R, and F1) to evaluate the semantic similarity, which make up for the limitation of the previous three metrics that can just measure the lexical similarity.

4.3 Baseline Methods

Many studies have focused on generating clarifying facets for queries, including rule-based methods [7, 14, 15], PLM- and LLM-based methods [27, 35], etc. In this paper, we implement the known state-of-the-art approaches for Web search clarification and categorize existing methods into three baseline groups.

PLM-based methods. PLM-based methods mainly indicate models based on BERT [5] and BART [18]. These methods including several facets obtaining paradigms like Labeling, Classification, Extraction (BERT-based), and Generation (BART-based), which are all trained with MIMICS-Click data. Read [35] for more details.

LLM + static demonstrations. The second group is LLM with static demonstrations. In other words, the demonstrations are the same for different queries. We fetch the results from two existing studies which use GPT-3 and GPT-3.5 Turbo as the base model respectively [27, 35]. In this two, the demonstrations stay consistent with their paper and will not be modified according to the query.

LLM + demonstration retrieval. We implement three basic demonstration selection strategies which are widely applied in existing in-context learning studies: (1) **Random**: The k demonstrations are randomly selected from the demonstration set. (2) **BM25** [33]: The demonstrations in the demonstration set are ranked by the BM25 score between the test data (q_x, D_x) and each demonstration, then the top-k demonstrations are selected. (3) **SBERT** [32]:The demonstrations in the demonstration set are ranked by the BERT similarity between the test data (q_x, D_x) and each demonstrations, then the top-k demonstrations are selected. (4) **BM25-d** and **SBERT-d**: It is shown that the diversification of demonstration is effective in some NLP tasks [17]. Therefore, we also apply this strategy as baseline to BM25 and SBERT. Specifically, when we choose demonstration, in addition to considering similarity, we also keep the facets of each demonstration from overlapping, thereby improving the diversity of facets. To this end, we first sort the demonstrations by similarity, and then, starting from the first demonstration, we record the facets set of the selected demonstrations. If the new demonstration facets have duplicates with this set, the demonstration is skipped.

4.4 Demonstration Features

In addition to demonstration selection, we are also interested in the impact of three important characteristics of demonstrations:

- *Demonstration Order*: The order of demonstrations significantly influence the generation quality. So we arrange the permutation of all demonstrations and calculate the results correspondingly.
- *Demonstration Context*: In some BART-based models [35], considering top-retrieved document snippets to generate facets is better than using the query only [35]. Since LLM naturally has powerful natural language understanding capabilities, we want to explore whether providing these contextual documents to LLM will have a greater impact on the experimental results.
- *Demonstration Number*: The number of demonstration may have an impact on the experimental results. We set the demonstration number k from 1 to 5 and set $k = 4$ in our main experiment.

The related experiments are conducted and analyzed in Section 4.7, 4.8, and 4.9, respectively.

4.5 Implementation Details

For the LLM base model, we apply the GPT-3.5 Turbo[2]. We apply the API to implement the model. The prompt we use is shown in Figure 4. We select BART-base for BART-IPI and BERT-base as the encoder for training DRM. The two models are implemented with Transformers and PyTorch. We select the AdamW optimizer with a learning rate of 1 × 10e-4. To control the learning progress of BART-IPI, we observe the proportion of BART-IPI covering facets on a held-out set with 20k queries after each epoch training on MIMICS-Click, and stop training when the proportion begins to decrease. When processing the IPIs, we use Stanza[3] for stemming and deduplication. For demonstration number k, we select k from 1 to 5 to observe the changes in the experimental results.

4.6 Experimental Results and Analysis

The experimental results are shown in Table 1. For all baseline methods and our proposed methods (except for RDS_{min} and RDS_{max}), we denote the best result for each metric as **bold** and the second best result for each metric as underlined. We also conduct significance test (t-test) to measure whether our proposed methods can outperform strong baselines significantly (denoted with "†"). We can conclude form the table from multiple dimensions that:

- **Performance of PLM-based Methods**: In our analysis of PLM-based approaches, we observe varying performance across

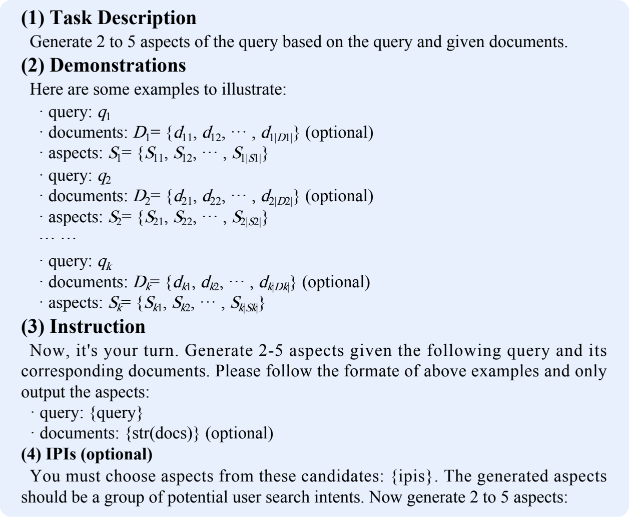
[1] MIMICS Dataset: https://github.com/microsoft/MIMICS

[2] GPT-3.5-turbo: https://platform.openai.com/playground?models=gpt-3.5-turbo

[3] https://stanfordnlp.github.io/stanza/

Table 1: Evaluation results for aspect items generation (set $k = 4$). The best result for each metric is marked in bold and the second best result for each metric is underlined. "†" denotes that the proposed method (RDS and DRM) achieves significant improvement compared with all baseline methods with $p < 0.05$.

		Term Overlap			Exact Match			Set BLEU				Set BERT		
Type	Model	Prec	Recall	F1	Prec	Recall	F1	1-gram	2-gram	3-gram	4-gram	Prec	Recall	F1
PLM	Labeling	0.1071	0.1436	0.1169	0.0859	0.1328	0.1008	0.2423	0.1448	0.1076	0.0946	0.5987	0.5973	0.5979
	Classification	0.0547	0.0572	0.0539	0.0476	0.0545	0.0489	0.1025	0.0613	0.0482	0.0409	0.4942	0.4925	0.4933
	Extraction	0.0411	0.0645	0.0469	0.0269	0.0491	0.0339	0.1914	0.0918	0.0479	0.0363	0.4601	0.4599	0.4598
	Generation-q	0.0683	0.0548	0.0577	0.0577	0.0459	0.0492	0.1932	0.0884	0.0549	0.0463	0.5315	0.5423	0.5384
	Generation-qD	0.1706	0.1599	0.1570	0.1489	0.1325	0.1350	0.2857	0.1833	0.1446	0.1283	0.6131	0.6133	0.6130
LLM	GPT-3	0.0713	0.1199	0.0842	0.0548	0.0861	0.0648	0.2157	0.1225	0.0853	0.0721	0.5553	0.5551	0.5550
	GPT-3.5-Turbo	0.0629	0.1102	0.0746	0.0356	0.0546	0.0420	0.2242	0.1222	0.0792	0.0664	0.5861	0.5865	0.5861
Baselines	Random	0.0623	0.0987	0.0717	0.0431	0.0667	0.0507	0.2222	0.1209	0.0777	0.0644	0.5788	0.5781	0.5783
	BM25	0.1076	0.1535	0.1201	0.0903	0.1266	0.1020	0.2597	0.1649	0.1217	0.1061	0.5931	0.5924	0.5926
	BM25-d	0.1275	0.1693	0.1382	0.1095	0.1437	0.1201	0.2770	0.1816	0.1388	0.1234	0.6162	0.6148	0.6154
	SBERT	0.1095	0.1541	0.1223	0.0942	0.1293	0.1057	0.2637	0.1671	0.1253	0.1078	0.6012	0.6004	0.6006
	SBERT-d	0.0999	0.1367	0.1099	0.0832	0.1114	0.0922	0.2578	0.1576	0.1142	0.0998	0.6025	0.6015	0.6019
Our	**RDS**	0.1702	0.2256†	0.1861†	0.1537	0.2041†	0.1700†	0.3086†	**0.2190**†	0.1804†	0.1617†	**0.6177**	**0.6162**	**0.6168**
	RDS$_{\text{min}}$	0.0944	0.1396	0.1074	0.0886	0.1263	0.1007	0.2558	0.1619	0.1195	0.1014	0.5798	0.5786	0.5791
	RDS$_{\text{max}}$	0.2680	0.3261	0.2838	0.2379	0.2947	0.2559	0.3316	0.2586	0.2279	0.2102	0.5932	0.5917	0.5924
	DRM	**0.1841**†	**0.2418**†	**0.2011**†	**0.1698**†	**0.2267**†	**0.1814**†	**0.3095**†	0.2175†	**0.1823**†	**0.1639**†	0.6165	0.6137	0.6144



(1) Task Description

Generate 2 to 5 aspects of the query based on the query and given documents.

(2) Demonstrations

Here are some examples to illustrate:

· query: q_1
· documents: $D_1 = \{d_{11}, d_{12}, \cdots, d_{1|D1|}\}$ (optional)
· aspects: $S_1 = \{S_{11}, S_{12}, \cdots, S_{1|S1|}\}$
· query: q_2
· documents: $D_2 = \{d_{21}, d_{22}, \cdots, d_{2|D2|}\}$ (optional)
· aspects: $S_2 = \{S_{21}, S_{22}, \cdots, S_{2|S2|}\}$
$\cdots\ \cdots$
· query: q_k
· documents: $D_k = \{d_{k1}, d_{k2}, \cdots, d_{k|Dk|}\}$ (optional)
· aspects: $S_k = \{S_{k1}, S_{k2}, \cdots, S_{k|Sk|}\}$

(3) Instruction

Now, it's your turn. Generate 2-5 aspects given the following query and its corresponding documents. Please follow the formate of above examples and only output the aspects:

· query: {query}
· documents: {str(docs)} (optional)

(4) IPIs (optional)

You must choose aspects from these candidates: {ipis}. The generated aspects should be a group of potential user search intents. Now generate 2 to 5 aspects:

Figure 4: Our prompt for generating facets. The (4) IPIs are optional and are not used in our experiments.

different methods. While simple classification and extraction methods show limited effectiveness, specific techniques such as labeling and generation-q demonstrate considerably stronger results. Notably, the Generation-qD method, which solely utilizes concatenated query and snippet texts for item generation, achieves the best outcomes among the PLM-based methods. This suggests that integrating both query and snippet contexts directly into the generation process can significantly enhance the PLM's capability to produce relevant and accurate aspect items.

• **Performance of LLM-based Baselines**: Our analysis reveals that although LLMs possess extensive open-domain knowledge, they still exhibit underwhelming performance in the task of facets generation. However, we observe a significant enhancement in performance when transitioning from static demonstrations to demonstration retrieval methods. This improvement underscores the crucial role of in-context learning abilities of LLMs and their significant reliance on the provided demonstrations. Additionally, the BM25-d method outperforms the standard BM25, suggesting that beyond semantic similarity, diversity in demonstration selection is vital. This insight provides crucial guidance for selecting demonstrations, indicating that a balanced consideration of both semantic relevance and diversity can significantly enhance the effectiveness of LLMs in the facets generation task.

Table 2: Experimental results when different LLMs are applied as the base model in DRM (set $k = 4$).

Base Model	TF	EF	BLEU-2	BF
Mistral-7B	0.1426	0.1173	0.1841	0.6131
LLaMA2-7B	0.1377	0.1154	0.1868	0.6090
DeepSeek-V3	0.1774	0.1659	0.2173	0.6140
DeepSeek-R1	0.1853	0.1709	0.2188	0.6129
GPT-3.5-Turbo	0.1861	0.1700	0.2190	**0.6168**
GPT-4-Turbo	**0.1905**	**0.1732**	**0.2210**	0.6159

• **Comparison Between PLM-based and LLM-based Methods**: In comparing PLM-based and LLM-based methods, it becomes evident that LLM-based approaches generally exhibit lower performance across most evaluation metrics when compared to finely-tuned PLM-based methods. This observation underscores the effectiveness of PLM-based models which, through fine-tuning, have been specifically optimized for particular tasks, leading to higher precision and effectiveness. In contrast, despite their broad knowledge base and inherent flexibility, existing LLM-based methods struggle to match this level of specialized performance, particularly in structured tasks like aspect item generation. This contrast highlights the trade-offs between the generalizability of LLMs and the targeted efficiency of PLM-based approaches.

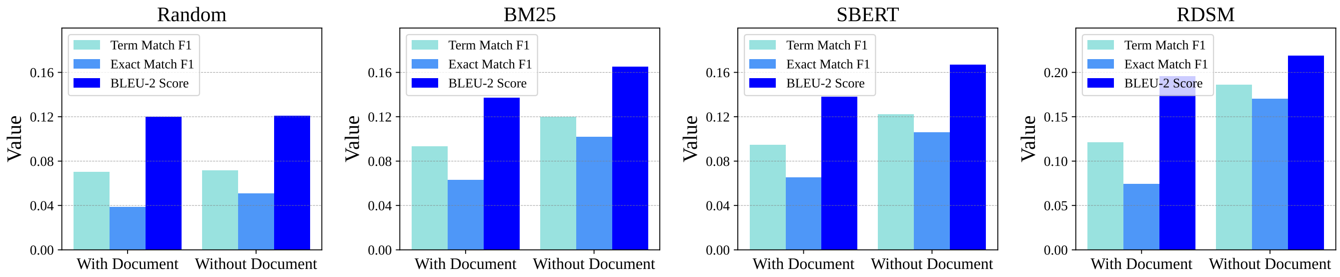


Figure 5: Comparison of whether have in-context documents.

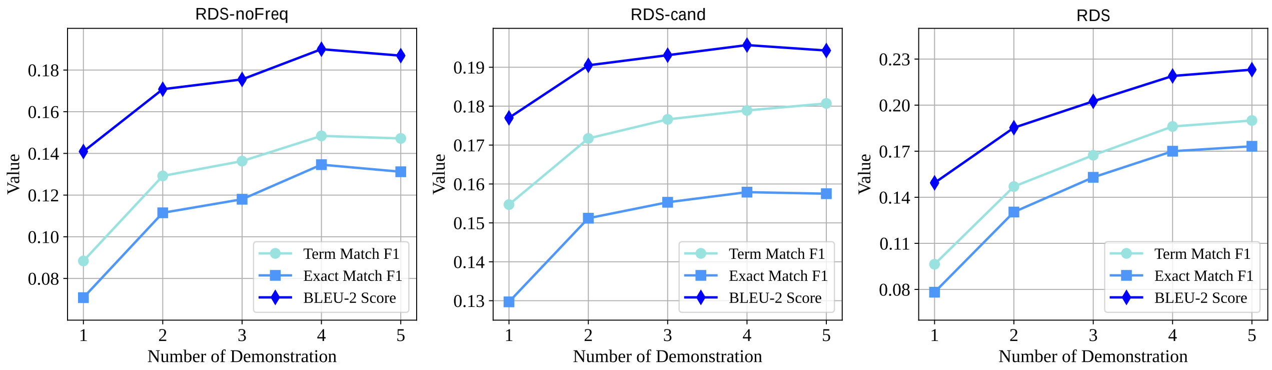


Figure 6: Comparison for different number of demonstrations.

DeepSeek-V3: 0.1774, 0.1659, 0.2173, 0.6140 DeepSeek-R1: 0.1853, 0.1709, 0.2188, 0.6129 • **Our Methods**: We can observe that our proposed method RDS demonstrates superior performance over existing baselines across nearly all evaluation metrics. This demonstrates the effectiveness of RDS, which can be attributed to the fact that RDS selects demonstrations based on the coverage and co-occurrence of IPIs. In this way, RDS can address the multi-dimensional nature of query facets by optimizing the demonstration inputs, thereby significantly enhancing the LLM's capability in facets generation.

Furthermore, the comparison between RDS_{min} and RDS_{max} reveals a significant variance in results, underscoring the sensitivity of LLMs to the order of provided demonstrations. Our subsequent model, DRM, capitalizes on this observation by re-ranking demonstrations, which leads to further improvements in performance. This refinement highlights the importance of demonstration order in enhancing the effectiveness of LLMs, ensuring that the most contextually relevant and beneficial demonstrations are prioritized.

• **Different Base LLMs**: Since different base LLMs may have different influence on the generation results, we choose three open-source LLMs (Mistral-7B, LLaMA2-7B, and DeepSeek-V3), two close-source LLMs (GPT-3.5-Turbo and GPT-4-Turbo), and one cutting-edge Large Reasoning Model DeepSeek-R1, as the base models for generation in DRM. We test the results on Term Match F1 (TF), Exact Match F1 (EF), BLEU-2, and BERT F1 (BF) as shown in Table 2. It can be seen that, the generation ability of the closed-source model is significantly better than that of the open-source model in almost all evaluation metrics. A potential solution to improve the performance of the open-source models is to apply Supervised Fine-Tuning (SFT) or instruction tuning with clarification data in MIMICS dataset. Besides, there is a trend that the generation ability can be improved with the increase of the model size.

4.7 Impact of Demonstration Order

Considering that LLMs may respond differently depending on the order of demonstrations presented in the prompt [12, 24, 41], we also conduct a full permutation of the order of demonstrations within the prompt. Analyzing the impact of demonstration order, we observe an interesting phenomenon that RDS_{max} consistently outperforms RDS_{min} across various metrics. This suggests that organizing demonstrations based on their maximum relevance to the task at hand leads to better facets generation performance. The higher scores in RDS_{max} indicate the effectiveness of this approach in leveraging relevant information effectively.

4.8 Impact of In-context Documents

As illustrated in Figure 5, the analysis across three metrics reveals distinctive patterns in performance with and without the inclusion of documents. For the Random method, there is a negligible difference between scenarios with and without documents, indicating that the presence of documents does not significantly influence the outcomes. However, for BM25, SBERT, and particularly our proposed method RDS, the performance is notably better without including documents. Specifically, RDS shows a significant improvement in results when documents are not included. This may be attributed to the irrelevant information within the documents, which could interfere with the ability of LLMs. This observation suggests that providing only the query and corresponding aspects as demonstrations to the LLM is more effective for this task. By focusing demonstrations strictly on relevant query-aspect pairs, the LLM can generate facets more effectively, avoiding the noise and distractions presented by lengthy documents.

4.9 Impact of Demonstration Number and Specific Heuristics for IPI Coverage

We further study how the experimental results vary with the demonstration number k increases from 1 to 5, and how the heuristics for IPI coverage can influence the results. To this end, we run RDS and two variants (RDS-noFreq which do not consider statistics mentioned in Section 3.3, and RDS-cand which adds IPIs as a part of the prompt shown in Figure 4) when the demonstration number is from 1 to 5 respectively. We record the term match F1, exact match F1, and Set BLEU-2 score for each setting of k. The experimental results are shown in Figure 6. It can be seen that, with the increase of k, all three metrics show an overall upward trend, but the rate of increase gradually slows down. This observation suggests a diminishing return on additional demonstrations beyond a certain point, highlighting the importance of optimizing the number of demonstrations to strike a balance between performance gains, computational efficiency, and model responsiveness.

5 Discussion

This paper focuses on in-context learning specifically for the Web search clarification task, demonstrating superior performance on the MIMICS dataset compared to general in-context learning methods. It should be emphasized that our primary objective is not to develop a universal in-context learning approach. Although the proposed method may not directly generalize to other tasks, its core concept can be adapted and transferred to other domains with appropriate modifications. For instance, in conversational recommender systems, the method could be similarly applied to predict which product attributes should be inquired about in the next user interaction [4, 22]. However, given the fundamental differences in task objectives, such applications fall outside the scope of this paper and are better suited as directions for future research.

6 Conclusion

Search clarification serves as a crucial component in conversational Web search systems. This work studies enhanced in-context demonstration selection for LLMs to improve their capability in generating clarification facets. Our approach introduces several key innovations: First, departing from conventional static or similarity-based demonstration selection methods, we define Independent Potential Intents (IPIs) for queries using our proposed BART-IPI model. Building upon this foundation, we develop two methods: (1) a heuristic-based Relevance-Driven Selection Model (RDS) that employs a greedy algorithm to select demonstrations covering the IPIs of the current query, and (2) a supervised Demonstration Reranking Model (DRM) that applies pair-wise learning-to-rank to refine RDS-selected demonstrations. Experimental results on the MIMICS dataset demonstrate that our methods significantly outperform both existing PLM-based and LLM-based approaches in clarification facet generation. Furthermore, we conduct comprehensive additional experiments to validate the effectiveness and robustness of our proposed methods.

Acknowledgments

This work was supported by Beijing Municipal Science and Technology Project No. Z231100010323009, National Science and Technology Major Project No. 2022ZD0120103, National Natural Science Foundation of China No. 62272467, Beijing Natural Science Foundation No. L233008, the fund for building world-class universities (disciplines) of Renmin University of China, and Public Computing Cloud, Renmin University of China. The work was partially done at the Beijing Key Laboratory of Research on Large Models and Intelligent Governance and Engineering Research Center of Next-Generation Intelligent Search and Recommendation, MOE.

References

[1] Mohammad Aliannejadi, Hamed Zamani, Fabio Crestani, et al. 2019. Asking clarifying questions in open-domain information-seeking conversations. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 475–484.

[2] Qibin Chen, Junyang Lin, Yichang Zhang, et al. 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)*. 1803–1813.

[3] Jeffrey Dalton, Sophie Fischer, Paul Owoicho, et al. 2022. Conversational Information Seeking: Theory and Application. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 3455–3458.

[4] Yang Deng, Yaliang Li, Fei Sun, et al. 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.

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, et al. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. 4171–4186.

[6] Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, and Zhifang Sui. 2022. A survey on in-context learning. *arXiv preprint arXiv:2301.00234* (2022).

[7] Zhicheng Dou, Sha Hu, Yulong Luo, et al. 2011. Finding dimensions for queries. In *Proceedings of the 20th ACM international conference on Information and knowledge management*. 1311–1320.

[8] Zhicheng Dou, Zhengbao Jiang, Sha Hu, et al. 2015. Automatically mining facets for queries from their search results. *IEEE Transactions on knowledge and data engineering* 28, 2 (2015), 385–397.

[9] Andrew Drozdov, Honglei Zhuang, Zhuyun Dai, Zhen Qin, Razieh Rahimi, Xuanhui Wang, Dana Alon, Mohit Iyyer, Andrew McCallum, Donald Metzler, et al. 2023. PaRaDe: Passage Ranking using Demonstrations with Large Language Models. *arXiv preprint arXiv:2310.14408* (2023).

[10] Helia Hashemi, Hamed Zamani, and W Bruce Croft. 2021. Learning multiple intent representations for search queries. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. 669–679.

[11] Helia Hashemi, Hamed Zamani, and W Bruce Croft. 2022. Stochastic Optimization of Text Set Generation for Learning Multiple Query Intent Representations. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. 4003–4008.

[12] Yupeng Hou, Junjie Zhang, Zihan Lin, Hongyu Lu, Ruobing Xie, Julian McAuley, and Wayne Xin Zhao. 2024. Large language models are zero-shot rankers for recommender systems. In *European Conference on Information Retrieval*. Springer, 364–381.

[13] Hyuhng Joon Kim, Hyunsoo Cho, Junyeob Kim, Taeuk Kim, Kang Min Yoo, and Sang-goo Lee. 2022. Self-generated in-context learning: Leveraging auto-regressive language models as a demonstration generator. *arXiv preprint arXiv:2206.08082* (2022).

[14] Weize Kong and James Allan. 2013. Extracting query facets from search results. In *Proceedings of the 36th international ACM SIGIR conference on Research and*

development in information retrieval. 93–102.

[15] Weize Kong and James Allan. 2014. Extending faceted search to the general web. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management.* 839–848.

[16] Wenqiang Lei, Xiangnan He, Yisong Miao, et al. 2020. Estimation-action-reflection: Towards deep interaction between conversational and recommender systems. In *Proceedings of the 13th International Conference on Web Search and Data Mining.* 304–312.

[17] Itay Levy, Ben Bogin, and Jonathan Berant. 2023. Diverse Demonstrations Improve In-context Compositional Generalization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers).* 1401–1422.

[18] Mike Lewis, Yinhan Liu, Naman Goyal, et al. 2020. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics.* 7871–7880.

[19] Raymond Li, Samira Ebrahimi Kahou, Hannes Schulz, et al. 2018. Towards deep conversational recommendations. *Advances in neural information processing systems* 31 (2018).

[20] Rui Li, Guoyin Wang, and Jiwei Li. 2023. Are Human-generated Demonstrations Necessary for In-context Learning? *arXiv preprint arXiv:2309.14681* (2023).

[21] Xiaonan Li, Kai Lv, Hang Yan, Tianyang Lin, Wei Zhu, Yuan Ni, Guotong Xie, Xiaoling Wang, and Xipeng Qiu. 2023. Unified Demonstration Retriever for In-Context Learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers).* 4644–4668.

[22] Zujie Liang, Huang Hu, Can Xu, et al. 2021. Learning Neural Templates for Recommender Dialogue System. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing.* 7821–7833.

[23] Jiachang Liu, Dinghan Shen, Yizhe Zhang, William B Dolan, Lawrence Carin, and Weizhu Chen. 2022. What Makes Good In-Context Examples for GPT-3?. In *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures.* 100–114.

[24] Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics* 12 (2024), 157–173.

[25] Wenhan Liu, Ziliang Zhao, Yutao Zhu, and Zhicheng Dou. 2024. Mining Exploratory Queries for Conversational Search. In *Proceedings of the ACM on Web Conference 2024.* 1386—-1394.

[26] Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. 2022. Fantastically Ordered Prompts and Where to Find Them: Overcoming Few-Shot Prompt Order Sensitivity. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers).* 8086–8098.

[27] Shiyu Ni, Keping Bi, Jiafeng Guo, and Xueqi Cheng. 2023. A Comparative Study of Training Objectives for Clarification Facet Generation. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region.* 1–10.

[28] Enrico Palumbo, Andreas Damianou, Alice Wang, Alva Liu, Ghazal Fazelnia, Francesco Fabbri, Rui Ferreira, Fabrizio Silvestri, Hugues Bouchard, Claudia Hauff, et al. 2023. Graph Learning for Exploratory Query Suggestions in an Instant Search System. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management.* 4780–4786.

[29] Filip Radlinski and Nick Craswell. 2017. A theoretical framework for conversational search. In *Proceedings of the 2017 conference on conference human information interaction and retrieval.* 117–126.

[30] Sudha Rao and Hal Daumé III. 2018. Learning to Ask Good Questions: Ranking Clarification Questions using Neural Expected Value of Perfect Information. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers).* 2737–2746.

[31] Sudha Rao and Hal Daumé III. 2019. Answer-based Adversarial Training for Generating Clarification Questions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers).* 143–155.

[32] Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing* (2019).

[33] Stephen Robertson, Hugo Zaragoza, et al. 2009. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends® in Information Retrieval* 3, 4 (2009), 333–389.

[34] Ohad Rubin, Jonathan Herzig, and Jonathan Berant. 2022. Learning To Retrieve Prompts for In-Context Learning. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies.* 2655–2671.

[35] Chris Samarinas, Arkin Dharawat, and Hamed Zamani. 2022. Revisiting Open Domain Query Facet Extraction and Generation. In *Proceedings of the 2022 ACM SIGIR International Conference on Theory of Information Retrieval.* 43–50.

[36] Ivan Sekulić, Mohammad Aliannejadi, and Fabio Crestani. 2021. User engagement prediction for clarification in search. In *Advances in Information Retrieval: 43rd European Conference on IR Research, ECIR 2021, Virtual Event, March 28–April 1, 2021, Proceedings, Part I 43.* Springer, 619–633.

[37] Zhengxiang Shi, Yue Feng, and Aldo Lipani. 2022. Learning to execute actions or ask clarification questions. In *Findings of the Association for Computational Linguistics: NAACL 2022.* 2060–2070.

[38] Taylor Sorensen, Joshua Robinson, Christopher Rytting, Alexander Shaw, Kyle Rogers, Alexia Delorey, Mahmoud Khalil, Nancy Fulda, and David Wingate. 2022. An Information-theoretic Approach to Prompt Engineering Without Ground Truth Labels. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers).* 819–862.

[39] Zhan Su, Zhicheng Dou, Yutao Zhu, and Ji-Rong Wen. 2022. Knowledge Enhanced Search Result Diversification. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining.* 1687–1695.

[40] Yueming Sun and Yi Zhang. 2018. Conversational recommender system. In *The 41st international acm sigir conference on research & development in information retrieval.* 235–244.

[41] Raphael Tang, Xinyu Zhang, Xueguang Ma, Jimmy Lin, and Ferhan Ture. 2023. Found in the middle: Permutation self-consistency improves listwise ranking in large language models. *arXiv preprint arXiv:2310.07712* (2023).

[42] Leila Tavakoli, Johanne R Trippas, Hamed Zamani, et al. 2022. MIMICS-Duo: Offline & Online Evaluation of Search Clarification. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval.* 3198–3208.

[43] Jan Trienes and Krisztian Balog. 2019. Identifying unclear questions in community question answering websites. In *European Conference on Information Retrieval.* Springer, 276–289.

[44] Shuting Wang, Zhicheng Dou, Jing Yao, Yujia Zhou, and Ji-Rong Wen. 2023. Incorporating Explicit Subtopics in Personalized Search. In *Proceedings of the ACM Web Conference 2023.* 3364–3374.

[45] Zhenduo Wang, Yuancheng Tu, Corby Rosset, et al. 2023. Zero-shot Clarifying Question Generation for Conversational Search. In *Proceedings of the ACM Web Conference 2023.* 3288–3298.

[46] Julia White, Gabriel Poesia, Robert Hawkins, et al. 2021. Open-domain clarification question generation without question examples. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing.* 563–570.

[47] Jingjing Xu, Yuechen Wang, Duyu Tang, et al. 2019. Asking clarification questions in knowledge-based question answering. 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).* 1618–1629.

[48] Lili Yu, Howard Chen, Sida I Wang, et al. 2020. Interactive Classification by Asking Informative Questions. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics.* 2664–2680.

[49] Hamed Zamani, Susan Dumais, Nick Craswell, et al. 2020. Generating clarifying questions for information retrieval. In *Proceedings of The Web Conference 2020.* 418–428.

[50] Hamed Zamani, Gord Lueck, Everest Chen, et al. 2020. Mimics: A large-scale data collection for search clarification. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management.* 3189–3196.

[51] Hamed Zamani, Bhaskar Mitra, Everest Chen, et al. 2020. Analyzing and Learning from User Interactions for Search Clarification. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval.* 1181–1190.

[52] Yiming Zhang, Lingfei Wu, Qi Shen, et al. 2022. Multiple Choice Questions based Multi-Interest Policy Learning for Conversational Recommendation. In *Proceedings of the ACM Web Conference 2022.* 2153–2162.

[53] Zhiling Zhang and Kenny Zhu. 2021. Diverse and specific clarification question generation with keywords. In *Proceedings of the Web Conference 2021.* 3501–3511.

[54] Ziliang Zhao and Zhicheng Dou. 2024. Generating Multi-turn Clarification for Web Information Seeking. In *Proceedings of the ACM on Web Conference 2024.* 1539–1548.

[55] Ziliang Zhao, Zhicheng Dou, Yu Guo, Zhao Cao, and Xiaohua Cheng. 2023. Improving Search Clarification with Structured Information Extracted from Search Results. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining.* 3549–3558.

[56] Ziliang Zhao, Zhicheng Dou, Jiaxin Mao, et al. 2022. Generating clarifying questions with web search results. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval.* 234–244.

[57] Kun Zhou, Wayne Xin Zhao, Shuqing Bian, et al. 2020. Improving conversational recommender systems via knowledge graph based semantic fusion. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining.* 1006–1014.

[58] Kun Zhou, Yuanhang Zhou, Wayne Xin Zhao, et al. 2020. Towards Topic-Guided Conversational Recommender System. In *Proceedings of the 28th International Conference on Computational Linguistics.* 4128–4139.