STYLE: Improving Domain Transferability of Asking Clarification Questions in Large Language Model Powered Conversational Agents

Yue Chen[♠◇]   **Chen Huang**[♠◇]   **Yang Deng**[♡]   **Wenqiang Lei**[♠◇∗]
Dingnan Jin[♣]   **Jia Liu**[♠]   **Tat-Seng Chua**[♡]

♠ College of Computer Science, Sichuan University, China
◇ Engineering Research Center of Machine Learning and Industry Intelligence,
Ministry of Education, China
♡ National University of Singapore, Singapore
♣ University of Electronic Science and Technology, China
chenyueeee24@gmail.com   wenqianglei@scu.edu.cn

Abstract

Equipping a conversational search engine with strategies regarding when to ask clarification questions is becoming increasingly important across various domains. Attributing to the context understanding capability of LLMs and their access to domain-specific sources of knowledge, LLM-based clarification strategies feature rapid transfer to various domains in a *post-hoc* manner. However, they still struggle to deliver promising performance on unseen domains, struggling to achieve effective domain transferability. We take the first step to investigate this issue and existing methods tend to produce one-size-fits-all strategies across diverse domains, limiting their search effectiveness. In response, we introduce a novel method, called STYLE, to achieve effective domain transferability. Our experimental results indicate that STYLE bears strong domain transferability, resulting in an average search performance improvement of ∼10% on four unseen domains.

1 Introduction

Recent research in conversational search systems has highlighted the potential of utilizing large language models (LLMs) to address ambiguities present in user queries (Kuhn et al., 2022; Zhang and Choi, 2023). A key focus has been on investigating the strategies regarding *when to ask* clarification questions during conversations (Aliannejadi et al., 2021). When developing a search system for a specific domain *without prior training*, its clarification strategy may be limited, since ambiguities are influenced by the domain-specific background knowledge (Ferrari et al., 2017). For example, financial terminology may be perceived as ambiguous by the conversational search systems that are not confident with the finance jargon. To address

this, recent studies (Zhang and Choi, 2023; Zhang et al., 2023) resort to LLMs to feature rapid transfer to various domains in a post-hoc manner, such as finance services (Deng et al., 2023a) and movie recommendation (Fan et al., 2023). This may be attributed to the context-understanding capability of LLMs and their access to domain-specific sources of knowledge (Zhang et al., 2023). However, empirical evidence suggests that the effectiveness of LLM-based methods remains suboptimal in unseen domains, with notable difficulties in achieving effective domain transferability (Deng et al., 2023b). This motivates us to rethink: **What impedes the domain transferability of LLM-based methods regarding the clarification strategy?**

To investigate this, we conduct an in-depth analysis to examine off-the-shelf LLM-based methods (detailed in Section 2). Our findings reveal that while LLM-based methods, armed with domain-specific knowledge sources, somewhat develop strategies that could work on unseen domains, they often produce a one-size-fits-all strategy, such as asking increasingly more questions as the conversation advances. Such strategies typically perform less effectively compared to tailored strategies trained on domain-specific data. This highlights the limited ability of LLM-based methods to adapt to unseen domains effectively. The challenge lies in the fact that strong domain transferability can not be achieved by training on single-domain data. Meanwhile, the mismatched distribution of domain-specific representations poses a significant obstacle to effective domain transfer. Affected by the above challenges, existing methods are accustomed to addressing ambiguities solely in a single domain (Rahmani et al., 2023; Aliannejadi et al., 2019), potentially leading to models becoming brittle when faced with unseen domains.

To this end, we propose a novel method,

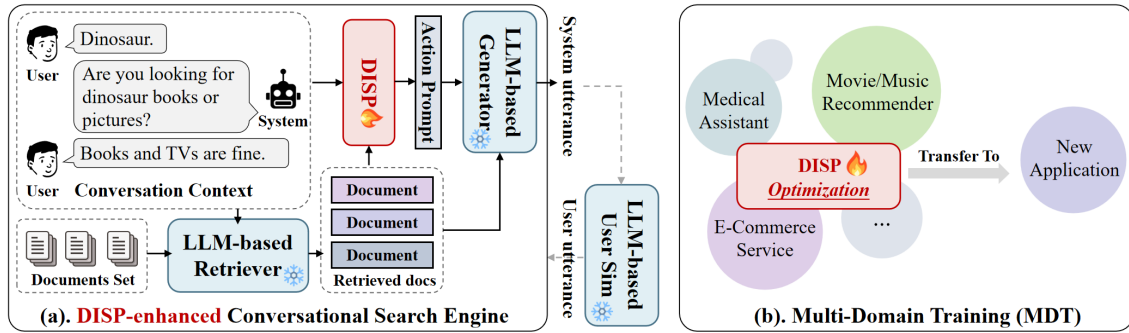
* Corresponding author.

10633

Figure 1: The **STYLE** contains domain-invariant strategy planner (DISP) and multi-domain training paradigm (MDT). The DISP extracts domain-invariant information and mitigates the swift of domain-specific distributions. The MDT encourages the domain transferability of DISP by population-based multi-domain training.

called STYLE, which features rapid tranSfer To previouslY unseen domains via tailored stratEgies in a post-hoc manner. STYLE comprises the domain-invariant strategy planner (DISP) in the conversational search engine and multi-domain training (MDT) paradigm. Specifically, the DISP is configured to extract domain-invariant information, mitigating the mismatch in the distribution of domain-specific representations and ensuring robustness across domains. We further leverage a retrieval-augmented paradigm to obtain documents that match the user queries in conjunction with their matching scores. The matching score reflects the retrieval quality and confidence of the retrieval module, which is highly correlated to the ambiguous level of the user queries and avoids the introduction of domain-specific semantic representation. Moreover, instead of relying on one solitary domain, MDT encourages the domain transferability of DISP by training it across multiple diverse domains. This is inspired by the population-based training (Long et al., 2023), which suggests that the generalization of a collaborative agent to held-out populations can be improved by training larger and more diverse populations (Charakorn et al., 2020). As such, STYLE enhances domain transferability and adapts its strategies for various domains.

We conduct experiments to evaluate STYLE using four domain-specific benchmark datasets in conversational search, including e-commerce, movies, and books. Our findings demonstrate that STYLE consistently surpasses existing LLM-based baselines in terms of domain transferability, resulting in a significant average performance improvement of ∼10%. Further analysis reveals that STYLE tailors its strategies to diversify them in different domains, which laid the foundation for its effectiveness. In summary, we analyze and address

the inadequacies of existing methods in determining when to ask clarification questions in scenarios involving previously unseen domains. This paves the way to advance both practical applications and academic research in this area. We present three key contributions:

- We verify and highlight that the one-size-fits-all strategies impedes the domain transferability of existing LLM-based methods when deciding on the timing to pose clarification questions.
- We present a new method STYLE to improve domain transferability in a post-hoc manner. It includes a domain-invariant strategy planner (DISP) in the search engine and a multi-domain training (MDT) paradigm.
- We experimentally show that STYLE bears strong domain transferability, resulting in an average search performance improvement of ∼10% on four unseen domains.

2 Preliminary Analysis

Previous analyses indicate that current methods' effectiveness of clarification strategies is compromised when applied in unseen domains (Deng et al., 2023a), yet the underlying cause remains unclear. This section aims to investigate the factors that may hinder the domain transferability of LLM-based methods. Our analysis involves two factors: search performance, and strategy characteristics.

2.1 Experimental Setup.

Baselines & Datasets. We compare the in-domain-trained model with LLM-based methods, including LLMs with zero-shot, few-shot, and CoT prompts. Here, the in-domain-trained model means the model is well-trained on data from the same domain as the test set, which contains a BERT-based
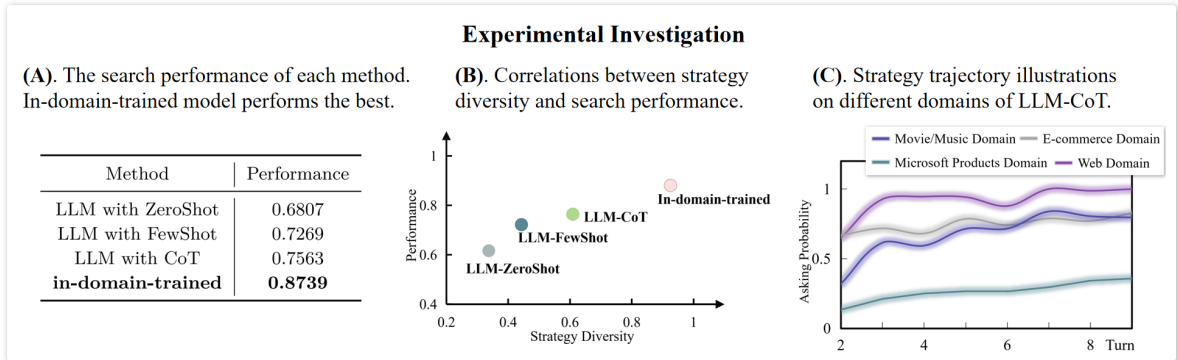
Figure 2: In-domain-trained model achieves the highest performance and strategy diversity. LLM-based methods often adopt a one-size-fits-all strategy with limited diversity across different domains (cf. Appendix A for details).

encoder and the two-layer fully connected network. We consider four domain-specific datasets, detailed in Appendix A.

Evaluation metrics. Following previous work (Chen et al., 2023), we evaluate search performance using SR@5. To analyze strategy characteristics, considering the variation of domain knowledge in different search domains, diverse clarification strategies may be required, we introduce strategy diversity as a metric. Specifically, as the strategy module decides whether to ask for clarification during each conversation turn, we analyze the diversity of strategies based on the strategy trajectories, each formed by multi-turn actions. Take *LLM with CoT* as an example, Figure 2 (C) reveals its strategy trajectories on different domains, represented as tr_1, tr_2, tr_3, tr_4. To evaluate its strategy diversity, we calculate the average similarities between pairs of trajectories. A lower similarity score indicates a greater variety of strategies. More information on strategy diversity can be found in Appendix A. It's important to note that the strategy diversity of the in-domain-trained model is assessed based on trajectories from their variances trained on its own specific domain.

Investigation Analysis. As illustrated in Figure 2(A), the in-domain-trained model outperforms LLM-based methods in terms of search performance. Additionally, in-depth analysis in Figure 2 (B)[1] reveals that the in-domain-trained model excels in both strategy diversity and search performance, whereas LLM-based methods often adopt a one-size-fits-all strategy with limited diversity across different domains (cf. Section 4.3 for details). Taking Figure 2 (C) for example, LLM with CoT shows a tendency to ask increasingly more questions as the conversation advances. This indi-

cates that the in-domain-trained model has diverse tailored strategies for various domains, while **LLM-based methods lack the flexibility to produce diverse strategies tailored to unseen domains**.

3 The Method

To achieve effective domain transferability when deciding on when to seek clarification, we propose STYLE. It includes a domain-invariant strategy planner (DISP) and multi-domain training paradigm (MDT). We provide the problem formulation in Section 3.1. The overall architecture of STYLE is introduced in section 3.2, with detailed explanations of DISP and MDT provided in section 3.3 and section 3.4, respectively.

3.1 Problem Formulation

Retrieval-based Conversational Search. We focus on the retrieval setting since it is one of the most common paradigms (Gao et al., 2022). Formally, for a user u_i, there exists a document d_i in the collection D that aligns with the user's intent. The interaction commences with the user's initial query q_1. At each turn t, when the user presents a query q_t, the conversation history $H_t = \{q_1, m_1, ..., q_{t-1}, m_{t-1}, q_t\}$ is formed, where q_{t-1} and m_{t-1} represent the user's query and the system's response at turn $t-1$. Given H_t, the system initially retrieves a subset of documents $D_t \subset D$. Subsequently, based on H_t and D_t, the system generates a response m_t by either posing a clarification question cq_t to the user or displaying the top x retrieved documents in D_t. This iterative process continues until the system presents d_i to the users or reaches the maximum number of turns T.

MDP Environment. The conversational search process is often formulated as a Markov Decision Process (MDP) (Chen et al., 2023). At turn t, con-

---

[1]More results are in Appendix A.

sidering q_t, H_t, and D_t, the system chooses an action $a_t \in \mathcal{A}$ from a set of clarification strategies $\mathcal{A}$. The goal is to learn a strategy π that maximizes the expected total rewards across the observed conversation episodes. This is formulated as: $\pi^* = \arg\max_{\pi \epsilon \Pi} \mathbb{E}\left[\sum_{t=0}^{T} r(s_t, a_t)\right]$, where s_t represents the state comprising H_t and D_t, $r(\cdot)$ is the immediate reward, denoted as r_t.

3.2 Overall Architecture

As illustrated in Figure 1, STYLE includes domain-invariant strategy planner (DISP) in the conversational search engine and multi-domain training paradigm (MDT). In Figure 1(b), STYLE initially trains DISP across various domains using MDT. Subsequently, the well-trained DISP can be introduced into unseen domains in a post-hoc manner. During the inference at conversation turn t, as depicted in Figure 1(a), the LLM-based retriever identifies documents D_t that closely match the user query within the conversation context H_t. Following this, based on H_t and D_t, the DISP decides whether to ask the user a clarification question by generating an action a_t using domain-invariant information. If the action a_t suggests asking, our conversational search engine utilizes the LLM-based generator to create a clarification question cq_{t+1} through few-shot CoT, considering the conversation context and the retrieved documents. Otherwise, our search engine presents x retrieved documents to the user.

3.3 Domain-Invariant Strategy Planner

To mitigate the discrepancy in the distribution of domain-specific representations, we propose DISP, implemented by a two-layer fully connected network. DISP is configured to extract domain-invariant representation that is general and structural, thereby enhancing its robustness to the domain transfer. The domain-invariant representation used in DISP is the concatenation of encoded conversation context and retrieved documents, and the ranking scores of retrieval results. This information captures the conversation state, the matching degree between user queries, the domain-specific knowledge from retrieved documents, and the confidence and quality of the retrieval module.

In particular, We utilize a fixed BERT (Devlin et al., 2019) to encode H_t and D_t into representations H_t and D_t, which remain unchanged during the training process. Then, we concatenate H_t and
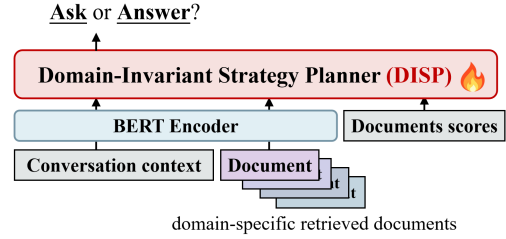

Figure 3: Domain-invariant strategy planner (**DISP**).

D_t with the ranking scores $score_t^{1:k}$ of k retrieved documents assigned to each document in D_t by the retrieval module since these scores can be indicative of the retrieval results' relevance or the retrieval system's confidence while relatively independent from the domain knowledge distributions.

Finally, such domain-invariant information plays the role of the state s_t and is fed into the DISP to yield action a_t, formulated below:

$$value = MLP\left(\mathbf{H}_t \oplus \mathbf{D}_t \oplus score_t^{1:k}\right), \quad (1)$$

$$a_t = \begin{cases} ask, & value \geq 0.5 \\ answer, & value < 0.5 \end{cases} \quad (2)$$

When DISP chooses to ask, the search engine generates and provides a clarification question to the user. Otherwise, the engine provides the x retrieved documents with the highest ranking scores as answers to the user.

3.4 Multi-Domain Training

To encourage the domain transferability of DISP, STYLE involves MDT. Inspired by the population-based training (Long et al., 2023), the MDT trains DISP on multiple domains (e.g., e-commerce, web), which enhances the generalization of the DISP to tailor its strategies to unseen domains.

Taking inspiration from population-based training, we engage in training the DISP using a diverse set of domain datasets. With n distinct domain-specific datasets represented as $\mathbf{B} = \{B_1, B_2, ...B_n\}$, we randomly select a subset as the training data for each epoch. This training set is designed to expose the planner to an assortment of strategies relevant to different domains, thereby bolstering its capacity to tailor its strategy in novel scenarios. Upon completion of training, we retain the refined parameters of the planner, allowing it to make efficient inferences on any unseen domain B^* ($B^* \notin \mathbf{B}$) effectively. In MDT, we engage in interactive reinforcement learning using an LLM-based user simulator as described in prior research (Deng et al., 2023b). Each sample includes a user u_i seeking a specific document d_i along with intent

details d_i^*. We leverage d_i^* and role instructions to formulate the user prompt P_{user} (Details are in Appendix F.3). When the system presents a statement m_{t+1} to user u_i, the user responds with q_{t+1} as follows:

$$q_{t+1} = LLM\left(P_{user}\left(d_i^*\right), m_{t+1}, H_t\right). \quad (3)$$

Once we receive the response q_{t+1}, we calculate the reward r_t based on predefined criteria. Subsequently, we employ the dueling Q-network for training, which can be expressed as follows, where $y_t = Q^*(s_t, a_t)$ and Q^* indicates DISP in this case.

$$y_t = \mathbb{E}_{s_{t+1}}\left[r_t + \gamma \max_{a \in \mathcal{A}} Q^*(s_{t+1}, a_{t+1})|s_t, a_t\right]. \quad (4)$$

4 Experiments

We conduct experiments to evaluate the domain transferability of STYLE and analyze reasons behind its success. We study three research questions:

- **RQ1:** Can STYLE effectively transfer to the unseen domain without domain-specific training?
- **RQ2:** Does STYLE produce strategies tailored to different domains?
- **RQ3:** Why is STYLE effective in dealing with unseen domains?

4.1 Experimental Settings

Domain-specific Datasets. We evaluate STYLE using domain-specific benchmark datasets in conversational search across different domains. Building on prior research (Aliannejadi et al., 2019; Wang and Ai, 2022; Owoicho et al., 2023), we consider four datasets: **ClariQ** (Aliannejadi et al., 2021) which gathers ambiguous web search queries, **FaqAnt** (Chen et al., 2023), focusing on a conversational FAQ task in the financial domain, **MS-Dialog** (Qu et al., 2018), which records ambiguous queries about Microsoft products, **Opendialkg** (Moon et al., 2019), which contains ambiguous questions regarding movies and books. Data statistics[2] are in Table 1. Importantly, these datasets contain unambiguous queries that are utilized to assess the strategy module's ability to correctly determine when it is unnecessary to ask questions. To simulate unseen domains, we conduct held-out evaluation by training STYLE on three datasets and reserving one as the unseen domain dataset.

---

[2]Data processing details can be found in Appendix C.

Dataset	Domain	# Cases	Ambiguous
ClariQ	Web Track	721/153/120	0.60
FaqAnt	E-commerce	2197/591/592	0.52
MSDialog	Microsoft Products	1298/325/325	0.53
Opendialkg	Books & Movie	1008/271/228	0.50

Table 1: Data statistics. *Ambiguous* indicates the proportion of ambiguous queries. See Appendix C for details.

Baselines. We compare STYLE with the following two classes of baselines[3], which allow us to observe the in-depth insight brought by asking clarification questions. Implementation details for each method can be found in Appendix F.

- **Retreval-based conversational search models** always provide answers to users without asking clarification questions. This includes 1) BM25: a statistics-based method, 2) senBERT (Reimers and Gurevych, 2019) uses siamese and triplet BERT to encode the input, 3) monoBERT (Nogueira and Cho, 2019): a BERT-based cross-encoder re-ranker, and 4) ChatSearch (Sun et al., 2023), a ChatGPT-based retrieval method with the SOTA performance.

- **LLM-based methods** opt to present either the retrieved documents or clarification questions to the user. This includes 1) ClarSim (Zhang and Choi, 2023) determines when to inquire using uncertainty modeling through self-questioning, 2) CLAM (Kuhn et al., 2022) identifies when to ask and generates questions through few-shot in-context learning, 3) CLAM$_{zeroShot}$ utilizes similar prompt following (Kuhn et al., 2022) except using zero-shot learning, and 4) ProCoT (Deng et al., 2023a) detects ambiguity and generates questions using few-shot CoT.

Evaluation Metrics. Considering that good clarification strategies lead to better performance of the search engines, we mainly focus on evaluating the efficiency and effectiveness of the search engines. Following previous works (Lei et al., 2020; Deng et al., 2023b; Chen et al., 2023), We adopt the Recall@5, average turn (AvgT) and the success rate at turn k (SR@k) as the automatic evaluation metrics.

4.2 Evaluation on unseen domains (RQ1)

We evaluate the domain transferability by assessing the conversational search performance on unseen domains. The results are displayed in Table 2, with a comprehensive analysis provided below.

---

[3]Notably, we omit supervised methods (Zhang and Choi, 2023; Aliannejadi et al., 2020) as they fail to transfer to unseen domains in a post-hoc manner.

Method		Recall@5↑	SR@3↑	SR@5↑	AvgT↓	Recall@5↑	SR@3↑	SR@5↑	AvgT↓
		ClariQ				*FaqAnt*			
Retrieval-based Conversational Search w/o CQ	BM25	0.6050	0.6638	0.6639	5.3193	0.3533	0.4967	0.5400	6.5833
	senBERT (Reimers and Gurevych, 2019)	0.1261	0.2773	0.3277	8.6891	0.1167	0.2467	0.3600	8.4667
	monoBERT (Nogueira and Cho, 2019)	0.1849	0.2605	0.3277	8.8908	0.1100	0.2533	0.3200	8.7733
	ChatSearch (Sun et al., 2023)	<u>0.6387</u>	0.6874	0.7059	4.9321	0.4167	0.5400	0.6200	6.0500
LLM-based methods w/ CQ	ClarSim (Zhang and Choi, 2023)	<u>0.6387</u>	0.6807	0.7143	4.8571	0.4200	0.5567	0.6033	6.0933
	CLAM (Kuhn et al., 2022)	<u>0.6387</u>	0.7143	0.7269	4.8697	<u>0.4711</u>	<u>0.5783</u>	0.6300	5.8699
	CLAM$_{zeroShot}$ (Kuhn et al., 2022)	<u>0.6387</u>	0.6555	0.6807	5.1428	0.4167	0.4567	0.4933	7.1133
	ProCoT (Deng et al., 2023a)	<u>0.6387</u>	<u>0.7311</u>	<u>0.7563</u>	<u>4.4986</u>	<u>0.4711</u>	0.5511	<u>0.6578</u>	<u>5.5811</u>
STYLE		<u>0.6387</u>	**0.7647**	**0.8655**	**3.8403**	<u>0.4711</u>	**0.5955**	**0.7173**	**5.1800**
		MSDialog				*Opendialkg*			
Retrieval-based Conversational Search w/o CQ	BM25	0.4300	0.5850	0.6200	5.9600	0.3964	0.4713	0.5330	6.5683
	senBERT (Reimers and Gurevych, 2019)	0.1533	0.2833	0.3500	8.4567	0.0970	0.2291	0.3304	8.4713
	monoBERT (Nogueira and Cho, 2019)	0.1667	0.3233	0.4133	8.0067	0.1850	0.3436	0.4273	7.5638
	ChatSearch (Sun et al., 2023)	0.4922	<u>0.6100</u>	<u>0.6378</u>	<u>5.6167</u>	0.4504	0.5749	0.6344	<u>5.4844</u>
LLM-based methods w/ CQ	ClarSim (Zhang and Choi, 2023)	<u>0.4950</u>	0.5817	0.6083	5.8783	0.4493	<u>0.5771</u>	<u>0.6564</u>	5.5507
	CLAM (Kuhn et al., 2022)	<u>0.4950</u>	0.5700	0.5933	6.0417	<u>0.4515</u>	0.5573	0.6189	5.6586
	CLAM$_{zeroShot}$ (Kuhn et al., 2022)	0.4633	0.5200	0.5300	6.7700	0.4478	0.5110	0.5595	6.5110
	ProCoT (Deng et al., 2023a)	<u>0.4950</u>	0.6067	0.6233	5.8067	0.4478	0.5653	0.6446	5.6858
STYLE		**0.4956**	**0.6144**	**0.6511**	**5.5678**	**0.4559**	**0.6157**	**0.7004**	**5.2632**

Table 2: Evaluation on unseen domains. We mark best results in **bold** and underline the second-best ones. We perform multiple runs to ensure the variance of each metric being less than 0.01. The runtime of each method is presented in the Appendix G.

STYLE exhibits superior performance on unseen domains, showcasing strong domain transferability. As illustrated in Table 2, STYLE achieves the most accurate search results in the fewest turns across unseen domains. On average, it outperforms the leading LLM-based baseline (ProCoT) by a significant ∼10% margin in SR@5 across all domains. With the highest SR@5 scores, STYLE also maintains a lead of over ∼5% in AvgT compared to the baselines in most domains. Furthermore, STYLE performs well even in domains where clarification questions play a less crucial role. Notably, on MSDialog, ChatSearch outperforms all LLM-based methods, suggesting that asking clarification questions may not be essential. Nevertheless, STYLE still surpasses ChatSearch, underscoring its robust transferability in unseen domains.

4.3 Strategy characterises analysis (RQ2)

This section verifies whether STYLE produces tailored strategies necessary for effective transfer to unseen domains. To achieve this, we compare STYLE with its counterpart STYLE$_{inDomain}$[4], which is trained on the corresponding in-domain dataset. Importantly, strategies contained in STYLE$_{inDomain}$ are carefully tailored to the specific domain via customized training. Subsequently, we visualize the strategies of various methods in Figure 4 and evaluate how closely they align with
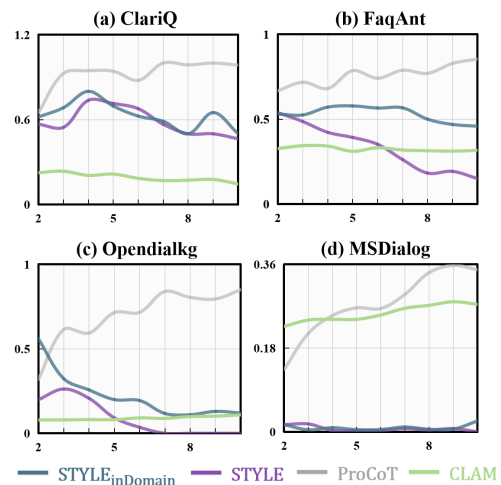
STYLE$_{inDomain}$ in Table 3.

Figure 4: Strategy trajectory illustration on two best LLM-based methods. The X-axis indicates the conversation turns. The Y-axis indicates the probability of asking. The strategy diversities is as follows STYLE: **0.9187**, ProCoT: 0.6079, CLAM: 0.4459.

LLM-based baselines prefer size-fits-all strategies across various domains. As depicted in Figure 4, the trajectory of strategies for ProCoT remains consistent across various domains, showing a tendency to ask more questions as the conversation progresses. Similarly, CLAM follows a uniform strategy, maintaining a consistent likelihood of asking questions at each turn across all domains. Quantitatively speaking, as shown in Table 3, the strategies of LLM-based baselines exhibit

---

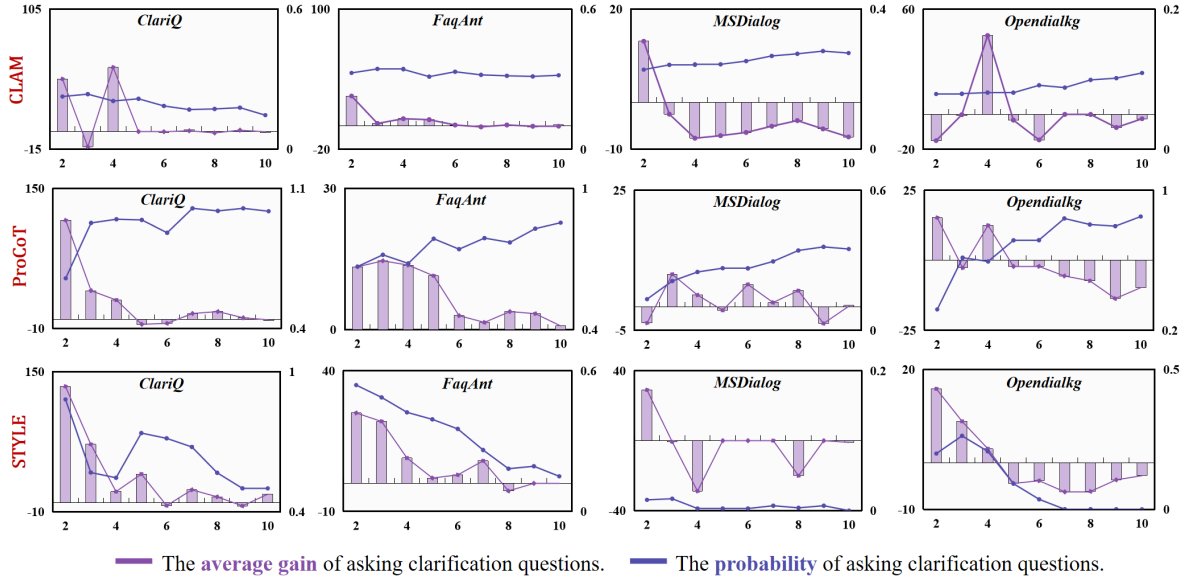[4]We report its performance in Appendix D and Table 6.

Figure 5: Illustration on the average gain and the probability of asking clarification questions. The X-axis indicates the conversation turns. The Y-axis (left) indicates the average asking gain at each turn, while the Y-axis (right) indicates the probability of asking.

Method	$DTW_{inDomain} \downarrow$			
	ClariQ	FaqAnt	MSDialog	Opendialkg
CLAM	3.8850	2.2735	2.4270	1.9885
ProCoT	2.5955	2.4427	2.4432	5.1715
STYLE	**0.5904**	**1.4819**	**0.0518**	**1.2939**

Table 3: The DTW similarities to $\text{STYLE}_{inDomain}$. Lower DTW corresponds to a better alignment with the strategy used in $\text{STYLE}_{inDomain}$.

limited alignment with the in-domain strategies of $\text{STYLE}_{inDomain}$, failing to tailor their strategies to different domains.

STYLE produces diverse and tailored strategies to different domains. As shown in Figure 4, STYLE demonstrates the highest level of strategy diversity compared to LLM-based baselines. It also showcases clarification strategies that closely align with those of $\text{STYLE}_{inDomain}$. For instance, on Opendialkg, both STYLE and $\text{STYLE}_{inDomain}$ tend to introduce clarification questions early in the conversation, gradually reducing the frequency of asking as the conversation progresses. This trend is consistent across the other datasets as well. The quantitative findings in Table 3 further confirm the superior alignment of STYLE with $\text{STYLE}_{inDomain}$. In conclusion, STYLE customizes strategies to meet diverse requirements across different domains.

4.4 Characteristics of STYLE (RQ3)

In this section, we investigate reasons behind the effective domain transferability of STYLE. To accomplish this, we measure the benefits of ask-

ing clarification questions at each turn (we term it asking benefits). In particular, a good clarification question posed at an appropriate time would assist the search module in retrieving documents. Thus, we calculate the ranking change of target documents after the user answers this clarification question[5]. The outcomes are presented in Figure 5.

The domain transferability of STYLE stems from its diverse strategies tailored to suit different domain needs. As depicted in Figure 5, all methods demonstrate diverse gains from asking questions across various domains. Given the fluctuating asking benefits across domains, an effective approach must adapt its strategy to meet the distinct requirements of each domain. In particular, CLAM sticks to a consistent probability of asking, regardless of the conversation turn. When the asking benefits show a notable decrease after the 2nd turn on MSDialog and FaqAnt, CLAM still performs a consistent probability and fails to adjust its strategy to adapt to the fluctuations of asking benefits. In contrast, STYLE showcases precise control over its strategy, adjusting to fluctuations in gains on a turn-by-turn basis. Notably, its asking benefits decrease gradually on ClariQ, prompting a reduction in the probability of asking questions as the benefits diminish. On MSDialog, where the asking benefits remain notably low (around -20), STYLE strategically limits its asking probability to a minimum level across all turns. Consequently, STYLE

---
[5]More details can be found in Appendix B.

Method	SR@5↑	AvgT↓	SR@5↑	AvgT↓	SR@5↑	AvgT↓	SR5↑	AvgT↓
	ClariQ		FaqAnt		MSDialog		Opendialkg	
STYLE	**0.8655**	**3.8403**	0.7173	5.1800	**0.6511**	**5.5678**	**0.7004**	**5.2632**
(a) - w/o DISP planner	0.7563	4.4986	0.6578	5.5811	0.6233	5.8067	0.6446	5.6858
(b) - w/ 1 domain	0.8291	4.0111	0.7133	5.1867	0.6407	5.6320	0.6799	5.3759
(c) - w/ 2 domains	0.8488	3.9188	0.6889	5.4222	0.6433	5.5933	0.6578	5.4479
(d) - w/o documents	0.8151	4.1639	**0.7317**	**5.0950**	0.6417	5.6250	0.6394	5.5707
(e) - w/o doc scores	0.7647	4.3908	0.6434	5.6350	0.6484	5.5750	0.6410	5.4956
(f) - w/o CoT	0.8319	4.0210	0.7167	5.2167	0.6456	5.5978	0.6806	5.4449

Table 4: Ablation evaluation. DISP is the key predictor for domain transferability of STYLE. Domain variability of training dataset and domain-invariant input also matters. The contribution of CoT is minimal.

exhibits more tailored strategies compared to other baselines, demonstrating better transferability by employing diverse strategies customized to meet specific domain needs. This enables STYLE to maximize the benefits of asking clarification questions in various domains, thereby enhancing its performance in unseen domains.

4.5 Ablation Study

We conduct ablations to ascertain the contribution of each module in STYLE. The results are provided in Table 4. Detailed findings are outlined below.

DISP. Table 4, row(a), shows that not including the DISP leads to the biggest decrease. This highlights its significance in STYLE.

Training Sources of MDT. A reduction in the variety of datasets employed for training, as shown in rows (b-c) of Table 4, leads to a noteworthy reduction in STYLE's performance. Considering the diverse origins of these datasets, the results reinforce the necessity of training the STYLE on sufficiently diverse domains to ensure transferability.

Domain-invariant Input of DISP. The removal of the input information for the DISP, documented in rows (d-e) of Table 4, provides further insights. Specifically, row (d) reveals that excluding retrieved documents D_t from the input diminishes performance across most domains. Additionally, as shown in row (e), the absence of document scores $score_t^{1:k}$ undermines the performance in all tested domains. The retrieval score is domain-invariant information and potentially indicative of both the retrieval model's confidence and the relevance of the documents retrieved, thereby aiding DISP in making more informed decisions.

Prompt Design. The CoT utilized by the LLM-based Generator indirectly influences the LLM-based Retriever's outputs by shaping the conversational context, thereby also impacting DISP's effectiveness. In row (f), the prompt design for generating questions was altered from CoT to a more straightforward in-context learning approach. The comparative analysis indicates that while utilizing CoT is beneficial, STYLE still retains superior performance even after CoT's removal, affirming the robustness of the STYLE.

5 Related Work

Determining when to ask clarification questions is critical in conversational search engines (Rahmani et al., 2023; Wang and Ai, 2021; Deng et al., 2022; Aliannejadi et al., 2021), which resolves the ambiguous user query (Aliannejadi et al., 2020). Supervised methods rely on extensive data annotation and training to adapt to specific domain requirements (Keyvan and Huang, 2022; Rahmani et al., 2023). Building on the success of Large Language Models (LLMs), developers have created strategies for determining when to ask questions based on LLMs. These strategies use LLMs to identify if a user query is ambiguous and generate questions through in-context learning (Kuhn et al., 2022; Zhang and Choi, 2023) and chain-of-thought methods (Deng et al., 2023a). This approach enables efficient development across various domains in a post-doc manner. However, their performance is still not satisfactory (Deng et al., 2023b,a), especially when applied to unseen domains. For the first time, We reveal that they struggle to establish tailored strategies when transferred to various unseen domains and propose the use of STYLE to address this challenge.

6 Conclusion

In this paper, we verify and highlight that one-size-fits-all strategies impede the transferability of existing LLM-based methods in unseen domains. To

tackle this limitation, we present STYLE, featuring a domain-invariant strategy planner (DISP) and a multi-domain training paradigm (MDT), to execute tailored strategies in conversational search engines in a post-hoc manner. We conduct a comprehensive set of experiments utilizing four benchmark datasets, each representing distinct domains. The empirical evidence from the results confirms the effectiveness of STYLE. Through detailed analysis, we ascertain that tailored strategies form the basis for efficient domain transferability, elucidating the efficacy of STYLE. With STYLE, we lay the foundation for research on effective model customization.

Limitation

First, owing to the multifaceted nature of conversational search, which encompasses question answering (QA), retrieval, and recommendation scenarios, a thorough analytical study of the clarification question module across all search settings would provide a more comprehensive understanding. However, such an expansive investigation would significantly amplify the experimental workload and diverge from the central research question. Consequently, our focus is confined to conversational retrieval, to extend our research to encompass other forms of conversational search in future work. In addition, since we focus on verifying whether STYLE performs well in unseen domains, multiple unseen datasets are required as test sets for each out-domain trained model. However, due to the heavy experimental workload, for each out-of-domain training model, we only use one dataset at a time as the unseen domain test set. In the future, we will consider using multiple datasets simultaneously as the test set.

Acknowledgements

References

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

Mohammad Aliannejadi, Julia Kiseleva, Aleksandr Chuklin, Jeff Dalton, and Mikhail Burtsev. 2021. Building and evaluating open-domain dialogue corpora with clarifying questions. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4473–4484, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Mohammad Aliannejadi, Hamed Zamani, Fabio Crestani, and W Bruce Croft. 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*, pages 475–484.

Rujikorn Charakorn, Poramate Manoonpong, and Nat Dilokthanakul. 2020. Investigating partner diversification methods in cooperative multi-agent deep reinforcement learning. In *Neural Information Processing: 27th International Conference, ICONIP 2020, Bangkok, Thailand, November 18–22, 2020, Proceedings, Part V 27*, pages 395–402. Springer.

Yue Chen, Dingnan Jin, Chen Huang, Jia Liu, and Wenqiang Lei. 2023. Travel: Tag-aware conversational faq retrieval via reinforcement learning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3861–3872.

Yang Deng, Wenqiang Lei, Wenxuan Zhang, Wai Lam, and Tat-Seng Chua. 2022. PACIFIC: Towards proactive conversational question answering over tabular and textual data in finance. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6970–6984, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yang Deng, Lizi Liao, Liang Chen, Hongru Wang, Wenqiang Lei, and Tat-Seng Chua. 2023a. Prompting and evaluating large language models for proactive dialogues: Clarification, target-guided, and non-collaboration. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10602–10621, Singapore. Association for Computational Linguistics.

Yang Deng, Wenxuan Zhang, Wai Lam, See-Kiong Ng, and Tat-Seng Chua. 2023b. Plug-and-play policy planner for large language model powered dialogue agents. In *The Twelfth International Conference on Learning Representations*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Wenqi Fan, Zihuai Zhao, Jiatong Li, Yunqing Liu, Xiaowei Mei, Yiqi Wang, Jiliang Tang, and Qing Li. 2023. Recommender systems in the era of

large language models (llms). *arXiv preprint arXiv:2307.02046*.

Alessio Ferrari, Beatrice Donati, and Stefania Gnesi. 2017. Detecting domain-specific ambiguities: An nlp approach based on wikipedia crawling and word embeddings. In *2017 IEEE 25th International Requirements Engineering Conference Workshops (REW)*, pages 393–399.

Joseph L Fleiss and Jacob Cohen. 1973. The equivalence of weighted kappa and the intraclass correlation coefficient as measures of reliability. *Educational and psychological measurement*, 33(3):613–619.

Jianfeng Gao, Chenyan Xiong, Paul Bennett, and Nick Craswell. 2022. Neural approaches to conversational information retrieval. *arXiv preprint arXiv:2201.05176*.

Chen Huang, Peixin Qin, Wenqiang Lei, and Jiancheng Lv. 2023. Reduce human labor on evaluating conversational information retrieval system: A human-machine collaboration approach. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10876–10891, Singapore. Association for Computational Linguistics.

Kimiya Keyvan and Jimmy Xiangji Huang. 2022. How to approach ambiguous queries in conversational search: A survey of techniques, approaches, tools, and challenges. *ACM Computing Surveys*, 55(6):1–40.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2022. Clam: Selective clarification for ambiguous questions with generative language models. *arXiv preprint arXiv:2212.07769*.

Wenqiang Lei, Gangyi Zhang, Xiangnan He, Yisong Miao, Xiang Wang, Liang Chen, and Tat-Seng Chua. 2020. Interactive path reasoning on graph for conversational recommendation. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2073–2083.

Weifan Long, Taixian Hou, Xiaoyi Wei, Shichao Yan, Peng Zhai, and Lihua Zhang. 2023. A survey on population-based deep reinforcement learning. *Mathematics*, 11(10):2234.

Seungwhan Moon, Pararth Shah, Anuj Kumar, and Rajen Subba. 2019. Opendialkg: Explainable conversational reasoning with attention-based walks over knowledge graphs. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pages 845–854.

Rodrigo Nogueira and Kyunghyun Cho. 2019. Passage re-ranking with bert. *arXiv preprint arXiv:1901.04085*.

Paul Owoicho, Ivan Sekulic, Mohammad Aliannejadi, Jeffrey Dalton, and Fabio Crestani. 2023. Exploiting simulated user feedback for conversational search: Ranking, rewriting, and beyond. In *Proceedings of*

the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 632–642.

Chen Qu, Liu Yang, W Bruce Croft, Johanne R Trippas, Yongfeng Zhang, and Minghui Qiu. 2018. Analyzing and characterizing user intent in information-seeking conversations. In *The 41st international acm sigir conference on research & development in information retrieval*, pages 989–992.

Hossein A. Rahmani, Xi Wang, Yue Feng, Qiang Zhang, Emine Yilmaz, and Aldo Lipani. 2023. A survey on asking clarification questions datasets in conversational systems. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2698–2716, Toronto, Canada. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. 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)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Ivan Sekulić, Mohammad Aliannejadi, and Fabio Crestani. 2022. Evaluating mixed-initiative conversational search systems via user simulation. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*, pages 888–896.

Weiwei Sun, Lingyong Yan, Xinyu Ma, Shuaiqiang Wang, Pengjie Ren, Zhumin Chen, Dawei Yin, and Zhaochun Ren. 2023. Is ChatGPT good at search? investigating large language models as re-ranking agents. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 14918–14937, Singapore. Association for Computational Linguistics.

Brandon Tran, Maryam Karimzadehgan, Rama Kumar Pasumarthi, Michael Bendersky, and Donald Metzler. 2019. Domain adaptation for enterprise email search. In *Proceedings of the 42nd international ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 25–34.

Xiaolei Wang, Xinyu Tang, Xin Zhao, Jingyuan Wang, and Ji-Rong Wen. 2023. Rethinking the evaluation for conversational recommendation in the era of large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10052–10065, Singapore. Association for Computational Linguistics.

Zhenduo Wang and Qingyao Ai. 2021. Controlling the risk of conversational search via reinforcement learning. In *Proceedings of the web conference 2021*, pages 1968–1977.

Zhenduo Wang and Qingyao Ai. 2022. Simulating and modeling the risk of conversational search. *ACM*

Transactions on Information Systems (TOIS), 40(4):1–33.

Hamed Zamani, Johanne R Trippas, Jeff Dalton, Filip Radlinski, et al. 2023. Conversational information seeking. *Foundations and Trends® in Information Retrieval*, 17(3-4):244–456.

Michael JQ Zhang and Eunsol Choi. 2023. Clarify when necessary: Resolving ambiguity through interaction with lms. *arXiv preprint arXiv:2311.09469*.

Zihan Zhang, Meng Fang, Ling Chen, Mohammad-Reza Namazi-Rad, and Jun Wang. 2023. How do large language models capture the ever-changing world knowledge? a review of recent advances. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 8289–8311, Singapore. Association for Computational Linguistics.

Appendix

A   Details of the Experimental Investigation

In this section, we present the details of the experimental investigation as shown in section 2.

Baselines & Dataset Firstly, we select three representative LLM-based methods as baselines, which are *LLM with ZeroShot*, *LLM with FewShot* (CLAM) (Kuhn et al., 2022), and *LLM with CoT* (ProCoT) (Deng et al., 2023a). Then, We consider four domain-specific datasets, which are: *ClariQ* in the Web domain (Aliannejadi et al., 2021), *FaqAnt* in the e-commerce domain (Chen et al., 2023), *MS-Dialog* in the Microsoft products domain (Qu et al., 2018), and *Opendialkg* in the Movies/Books domain (Moon et al., 2019).

Strategy Diversity. Firstly, we introduce the approach for quantifying strategy diversity. As the strategy module decides whether to ask a clarification question during each conversation turn, we analyze the diversity of strategies based on the *strategy trajectories*, each formed by multi-turn actions.

Take *LLM with CoT* as an example, Figure 6 reveals its strategy trajectories on different domains, represented as tr_1, tr_2, tr_3, tr_4. To evaluate its strategy diversity, we calculate the average similarities between pairs of trajectories. To this end, we employ Dynamic Time Warping (DTW) distance, a widely recognized metric for gauging the similarity of sequences. A greater DTW distance between trajectories indicates a lower sequence similarity, leading to a higher level of strategic diversity. Consequently, the strategy diversity of *LLM with CoT*
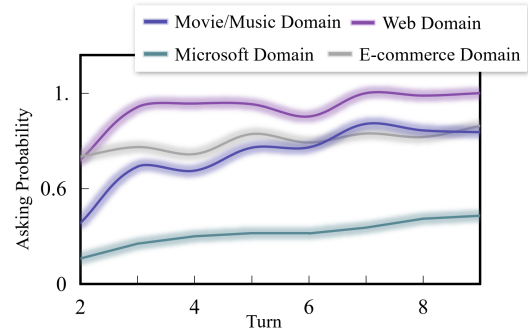

Figure 6: The strategy trajectories of *LLM with CoT* on four domains. The X-axis indicates conversation turns. The Y-axis indicates the probability of asking clarification questions.

is quantified by the average DTW distance between pairs of the four trajectories:

$$\frac{dtw\,(tr_1, tr_2) + dtw\,(tr_1, tr_3) + ... + dtw\,(tr_3, tr_4)}{6}. \quad (5)$$

It's important to note that the strategy diversity of the in-domain-trained model is assessed based on trajectories from their variances trained on its own specific domain.

Overall Analysis. We obtain the strategy diversity and performance of each method on four datasets across different domains, as shown in Table 5. Table 5 reveals that the in-domain-trained model outperforms LLM-based methods in search performance on all domains. Meanwhile, the in-domain-trained model excels in both strategy diversity and search performance, whereas LLM-based methods often adopt a one-size-fits-all strategy with limited diversity across different domains as shown in Table 5. Additionally, there is a strong correlation (Pearson's correlation fluctuates between 0.9060 and 0.9923.) between strategy diversity and search performance in all domains.

This indicates that the in-domain-trained model has diverse tailored strategies for various domains, while **LLM-based methods lack the flexibility to produce diverse strategies tailored to unseen domains**.

B   Asking Benefit Calculation

In this section, we present the details of calculating the benefit of asking a clarification question cq_t at turn t. Considering that a good clarification question helps the search engine retrieve the user's desired information, we calculate the ranking change of the user's desired document d_i after the user answers the question cq_t. Ideally, if the

Method	Strategy Diversity	Performance (SR@5)			
		ClariQ	FaqAnt	MSDialog	Opendialkg
LLM with ZeroShot	0.3362	0.6807	0.4933	0.5300	0.5595
LLM with FewShot	0.4459	0.7269	0.6300	0.5933	0.6189
LLM with CoT	0.6079	0.7563	0.6578	0.6233	0.6446
In-domain-trained model	0.9200	0.8739	0.7233	0.6400	0.7269

Table 5: We calculate the performance of each method in different datasets and the strategy diversity of each method. The **pearson correlation coefficient** between the strategy diversity and the performance on each domain is ClariQ: **0.9923**, FaqAnt: **0.9060**, MSDialog: **0.9355**, Opendialkg: **0.9869**.

system asks the user a good question at an appropriate time, the user's answer to this question will assist the search module in retrieving d_i accurately. We formulate the benefit of the question cq_t as:

$$gain_{cq_t} = rank_t(d_i) - rank_{t+1}(d_i), \qquad (6)$$

where d_i indicates the user's desired document. $rank_t(d_i)$ indicates the rank of d_i at turn t and $rank_{t+1}(d_i)$ indicates its rank at turn $t+1$ after the user answer cq_t. After calculating the benefit of each clarification question, we can measure the average benefit per question that can be achieved at each turn of the conversation.

C   Data Processing

We perform the experiments on four datasets: ClariQ (Aliannejadi et al., 2021), FaqAnt (Chen et al., 2023), MSDialog (Qu et al., 2018) and Opendialkg (Moon et al., 2019). To obtain data that is challenging and fits our setting, we perform the data processing on these datasets. We built our data in a format that fit our setting and increased the proportion of ambiguous queries in our data. More specially, we need to obtain the data in the format of $(u_i, d_i, d_i^*, q_i^{ini})$. u_i is the user ID. d_i indicates the ground truth document that matches the user intent. d_i^* is the intent information of the user and q_i^{ini} is the first-turn query of the user. In this section, we explain the process of data processing.

ClariQ contains conversations between the user and the search agent. Each conversation features an initial query q_i^{ini} of user u_i, an ambiguity classification label of the query (1: ambiguous/0: not ambiguous), a clarification question cq_i posed to the user, and a corresponding facet that aligns with the user's intent. Initially, we consider this facet to be the user's ground truth document d_i. Subsequently, we construct the intent information d_i^*. We employ ChatGPT to rephrase d_i, modifying certain key terms and enriching the context to obtain d_i^*. Finally, to augment the complexity of our task,

we removed a portion of conversations where the ambiguity label of query q_i^{ini} is 0 to ensure a preponderance of ambiguous queries. After the above process, we obtain a set of 1000 conversations in ClariQ for the experiment.

FaqAnt contains conversations between the user and a finance customer service agent. Each conversation includes an initial query q_i^{ini} of user u_i, an ambiguity label of the query (1: ambiguous/0: not ambiguous), and the FAQ question-answer pair which matches the user intent. We first take the user's desired question-answer pair as the ground truth document d_i. We then construct the refined intent information d_i^* by applying ChatGPT to paraphrase d_i. Finally, to ensure that our task is challenging and contains enough ambiguous queries, we removed a portion of conversations following our operation on ClariQ. After the above process, we obtain a set of 3380 conversations.

MSDialog encompasses question-answering conversations sourced from the Microsoft forum, featuring discussions with multiple participants. This includes the user u_i with their initial query q_i^{ini}, as well as responses from various Microsoft human agents. First, we need the ground truth document d_i that matches the user intent. Fortunately, in MSDialog, there is a binary label for each turn in the conversation marking whether this turn is acknowledged as the right answer to the user's intent. Following prior research (Wang and Ai, 2022), we determine the ground truth document d_i by selecting the response with the highest vote count. Then, we construct the intent information d_i^* by utilizing ChatGPT to rephrase d_i into d_i^*. Finally, to maintain a dataset rich in ambiguous queries, we removed a portion of conversations that the d_i can be simply retrieved by BM25 given the first turn query q_i^{ini}. This yields an experimental set comprising 1948 conversations.

Opendialkg consists of dialogues wherein a user seeks a recommendation or opinion from an agent

Method	SR@5↑	AvgT↓	SR@5↑	AvgT↓	SR@5↑	AvgT↓	SR5↑	AvgT↓
	ClariQ		*FaqAnt*		*MSDialog*		*Opendialkg*	
senBERT$_{inDomain}$	0.6975	5.0672	0.6067	6.0567	0.6000	6.2600	0.5242	6.6167
monoBERT$_{inDomain}$	0.6555	5.5462	0.6643	5.6710	0.5934	6.3700	0.5286	6.4669
STYLE$_{inDomain}$	**0.8739**	**3.6303**	**0.7233**	**5.1733**	<u>0.6400</u>	<u>5.6133</u>	**0.7269**	**5.1277**
STYLE	<u>0.8655</u>	<u>3.8403</u>	<u>0.7173</u>	<u>5.1800</u>	**0.6511**	**5.5678**	<u>0.7004</u>	<u>5.2632</u>

Table 6: In-domain training analysis. The subscript $inDomain$ indicates that this method was trained on the same unseen domain where the evaluation is performed. We mark the values indicating the best performance in **bold** and the second-best performance in <u>underline</u>.

on topics such as movies, music, or books, initiated with a query q_i^{ini}. Originally, Opendialkg was utilized for conversational reasoning and knowledge graph entity prediction tasks. Aligning with the methodologies of prior studies (Wang and Ai, 2021, 2022), our approach concentrates on conversation retrieval without leveraging knowledge graph-centric models. To establish the ground truth document d_i, each conversation undergoes human review. Subsequently, we generate the intent information d_i^* by utilizing ChatGPT to rearticulate d_i into a rephrased version, d_i^*. This method results in an experimental dataset of 1507 conversations.

D In-domain Training Analysis

In this section, we first demonstrate whether our method can outperform existing supervised methods in unseen domains when training data for unseen domains are available for both our method and existing methods. Then, we assess the capability of our method to maintain superior performance over existing methods when training data for unseen domains is only accessible to the supervised methods.

By conducting the comparison against well-trained supervised baselines, we save the need to compare with numerous domain adaptation methods (Tran et al., 2019), considering that supervised baselines with domain-specific training data surpass domain adaptation methods that lack such data. It is worth noticing that we only consider the training data (query-document pairs) for the document retrieval model to be available, as annotating the training data for supervised clarification question models does not match our setting.

As shown in Table 6, given sufficient training data in unseen domains, our method STYLE$_{inDomain}$ significantly outperforms the supervised retrieval method. This illustrates that **our method has a higher performance upper bound when extensive in-domain training data is at**

hand. Moreover, to our supervised, our method surpasses existing supervised methods in search performance, even when in-domain training data is only available for existing methods. This indicates that **our method does not rely on domain-specific training and has robust transferability**, which enables its efficient application across unseen domains.

E Human Evaluation of Clarification Question

To rigorously evaluate the quality of the clarification questions produced by each method, we performed a human evaluation. Our comparison focused on STYLE and the top-performing LLM-based method, i.e., ProCoT and CLAM (Deng et al., 2023a; Kuhn et al., 2022). From the clarification questions generated by each method, we randomly selected 100 instances, ensuring that the conversation context, the documents retrieved, and the user intent information were included in the sample. For the evaluation, we enlisted three independent raters to assess the sampled questions based on two specific criteria:

- Helpfulness. This criterion gauges if the posed question is informative and has the potential to elicit valuable information from the user.
- Intent Consistency. This measures the degree to which the question includes elements (such as keywords) that are pertinent to the user's intent.

To ascertain the level of consensus among the evaluators, we calculated the inter-rater reliability using Fleiss' Kappa (Fleiss and Cohen, 1973). The results revealed a Kappa score of 0.517 for Helpfulness and 0.782 for Intent Consistency, indicating a moderate to substantial agreement among the raters. Figure 7 presents the outcomes of our
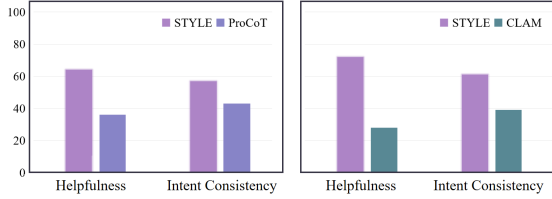
Figure 7: The human evaluation results of the quality of clarification questions. The y-axis represents the number of samples preferred by human judges.

human evaluation, which demonstrate that the clarification questions conceived by STYLE outperform those from ProCoT and CLAM in terms of both Helpfulness and Intent Consistency. This suggests that STYLE's generated questions are superior in both providing informative content and aligning with the user's intended purpose.

F Implementation Details

F.1 Parameters of our method

We split the dataset by 6:1:1 for training, validation, and testing. During the training, we randomly sample data from datasets in multiple domains. We set the maximum turn T as 10 and the number of training episodes as 1800. We set the size of the experience buffer in DQN as 10000 and the sample size as 32. We set the learning rate as 1e-4 with Adam's optimizer. The discount factor γ is 0.99. We adopt the reward of a successful search as 1.0 and the punishment of exceeding the maximum turns as -0.5. We set the number of presented documents x to users as 5. We set the number of layers in the BERT-based encoder in section 3.3 as 3.

F.2 Baseline Implementation

Retreval-based conversational search models. We consider the representative retrieval-based baselines including BM25, sentenceBERT (Reimers and Gurevych, 2019), monoBERT (Nogueira and Cho, 2019), and ChatSearch (Sun et al., 2023). The BERT-based baselines (sentenceBERT and monoBERT) are initialized by the publicly available checkpoint pre-trained from the huggingface[6] based on an open-domain corpus, such as MS MARCO[7]. We then fine-tune these methods on the same training source as our method, which is not in the same domain as the test set. During training, we set the learning rate as 5e-5 and utilized

---

[6]https://huggingface.co/OpenMatch/cocodr-base-msmarco
[7]https://microsoft.github.io/MSMARCO-Question-Answering/

the same number of training data following our method. We set the epoch to 15 and the batch size to 16 using the AdamW optimizer For the LLM-based retrieval method (ChatSearch), we use the permutation generation prompt as described in the corresponding paper (Sun et al., 2023).

LLM-based methods. We construct the LLM-based method based on *gpt-3.5-turbo*. For the implementation of CLAM, we adhere to the prompts outlined in the paper (Kuhn et al., 2022) and undertake the task of clarification need prediction and question generation utilizing few-shot in-context learning, as demonstrated in their work. In the case of ClarSim, we diverge from the original approach described in the original paper (Zhang and Choi, 2023) because their method necessitates the entropy information from the output of the decoder and the Intended Interpretation, which our models and dataset do not provide. Alternatively, we apply the Self-Ask strategy as indicated in their publication, prompting LLMs to make a decision—represented by the outputs "Yes" or "No"—on whether to pose a question. For ProCoT (Deng et al., 2023a), it depends on a grounded document, which our dataset lacks. To adjust for this, we replace the grounded document with documents we have retrieved and execute a similar inquiry strategy using few-shot Chain of Thought (CoT) prompts. Lastly, when it comes to $\text{CLAM}_{zeroShot}$, we dispense with the few-shot examples and employ the same prompts used in CLAM but through a zero-shot in-context learning framework. The prompts of LLM-based methods are presented in Figure 8 & 9.

F.3 Implementation of User Simulators

Evaluating a conversational system presents a significant challenge due to the multi-turn nature of user-system conversations (Huang et al., 2023; Zamani et al., 2023). In this paper, we resort to the user simulator. The simulated users are expected to play their assigned role as information seekers to communicate with the search engines. Previous studies have leveraged LLMs as user simulators and demonstrated their good performance (Deng et al., 2023b; Sekulić et al., 2022; Wang et al., 2023). Therefore, we utilize the ChatGPT to construct the user simulator based on *gpt-3.5-turbo*. Given a user u_i with the intent information d_i^*, we utilize d_i^* and some role instructions to formulate the user prompt P_{user} as shown in Figure 10. When the search engine asks the user a clarification ques-

tion, we send P_{user} into the ChatGPT and obtain the user's answer to this question. When the search engine provides the retrieved documents to users, we simply ask the user to give a positive or negative response depending on whether the user's desired document d_i^* is in the provided document.

G Runtime Analysis

In this section, we conduct runtime analysis to verify that our method STYLE does not incur additional runtime overhead. Specifically, we evaluate the average running time of different methods within each turn. As shown in Table 7, STYLE requires less time per turn. This is because we use the lightweight model (DISP) as the strategy module, instead of the parameter-intensive LLMs commonly employed in prior approaches. Consequently, our added component significantly reduces the time needed during execution.

Method	Runtime Per Turn ↓
ClarSim	3.3355s
CLAM$_{zeroShot}$	2.1435s
CLAM	1.8269s
ProCoT	2.6375s
STYLE	**1.5773s**

Table 7: The runtime analysis. STYLE takes only 1.5773 seconds on average per turn, which is less than other methods.

This bot determines whether the given query is ambiguous or not.

Query: What was the name of Elvis Presley's home?
The query is ambiguous: False.

Query: Find me map of USA.
The query is ambiguous: True.

Query: Find me information about the Ritz Carlton Lake Las Vegas.
The query is ambiguous: False.

Query: Find me information on sat.
The query is ambiguous: True.

Query: {query}
The query is ambiguous: {}

(a) Few Shot Ambiguous Query Detection (CLAM)

Given the ambiguous user query, ask a clarifying question to the user. Your clarifying question should always contains some options in format of 'Like option1, option2...'.

Query: Tell me about Obama family tree.
Question: What do you want to know about the Obama family tree? Like his parents and grandparents, biographical information and TIME magazine photo essay.

Query: How should I use the septic system design?
Question: What do you want to know about septic system design? Like how it works, the required maintenance, alternatives for the design and the contractor.

Query: {query}
Question: {}

(b) Few Shot Clarification Question Generation (CLAM)

Given the user query and retrieved documents, first identify whether the query is ambiguous or not. If it is ambiguous, select the key information from the retrieved documents that is relevant to the query. Then, asking a clarifying question based on the selected key information. Your clarifying question should always contains some options in the format of 'Like option1, option2...'. Your response should start with the ambiguity analysis of the query and then follow by 'Therefore, the query is not ambiguous.' or 'Therefore, the query is ambiguous. The selected key information is ... The clarifying question is ...'.

Query: Barack Obama's parents and grandparents.
Retrieved Documents: [1]. Where did Barack Obama's parents and grandparents come from?; [2]. Find biographical information on Barack Obama's mother.; [3]. Find the TIME magazine photo essay 'Barack Obama's Family Tree'.......
Output: There is one document that highly matches the user query than other documents, that is [1]. Therefore, the query is not ambiguous.;

Query: Where can I find cheap internet?
Retrieved Documents: [1]. I want to find cheap DSL providers.; [2]. What are some low-cost broadband internet providers?; [3]. Where can I find cheat codes for PlayStation 2 games?.......
Output: First, there are no document that highly match the user query. It is uncertain which document is referred to. Therefore, the query is ambiguous. Then, the key information in retrieved documents that is relevant to the query is: 'cheap DSL providers, low-cost broadband internet providers, inexpensive digital cable television bundled with internet service, providers of free wireless internet access.'. Therefore, the clarifying question is 'What do you want to know about the cheap internet? Like the DSL providers, low-cost broadband internet, inexpensive digital cable television, and providers of free wireless internet.

Query: {query}
Retrieved Documents: {retrieved documents}
Output: {}

(c) Few Shot Chain-of-Thought Clarification Question Generation (ProCoT, ClarSim, STYLE)

Figure 8: The Prompting Design of Clarification Question Module (1)

Given the user query and five documents retrieved by the query, determine if the query is ambiguous or not. If the query is ambigous, output 'Yes'. Otherwise, output 'No'.

Query: {query}
Retrieved Documents: {retrieved documents}
Output: {}

(a) Zero Shot Ambiguous Query Detection (CLAM$_{zeroShot}$)

Given the clarification, will I know the correct answer? Please answer "Yes" or "No".

Query: {query}
Retrieved Documents: {retrieved documents}
Clarification: {clarification question}
Output: {}

(b) Self Ask Clarification (ClarSim)

Figure 9: The Prompting Design of Clarification Question Module (2)

Now enter the role-playing mode. In the following conversation, you will play as the information seeker to communicate with the system.
Your information need is {information need}.
The conversation history is {conversation history}.
The system's current question is {question}.
You should answer the system's current question by selecting options of this question based on your information need and the conversation history. Your answer should follow these rules:
1 .Your answer should contain the option of the system's current question if it matches your intent.
2. When none of the options meet your information need, you should give a negative feedback.
3. Your answer shouldn't be too long.

Your answer is: {}

(a) User Simulator Feedback

This is RankGPT, an intelligent assistant that can rank passages based on their relevancy to the query.
The following are {num} passages, each indicated by number identifier []. I can rank them based on their relevance to query: {query}
[1] {passage_1}
[2] {passage_2}
(more passages) ...
The search query is: {query}.
I will rank the {num} passages above based on their relevance to the search query. The passages will be listed in descending order using identifiers, and the most relevant passages should be listed first, and the output format should be [] > [] > etc, e.g., [1] > [2] > etc.
The ranking results of the {num} passages (only identifiers) is:

(b) Document ReRank

Figure 10: Prompting Design of Use Simulator and Retrieval Module