Large Language Model based Long-tail Query Rewriting in Taobao Search

Wenjun Peng*
pengwj@mail.ustc.edu.cn
pengwenjun.pwj@taobao.com
University of Science and Technology of China
Hefei, Anhui, CHN

Guiyang Li
Yue Jiang
Zilong Wang
{liguiyang.lgy,jy270069,huanshi.wzl}@taobao.com
Taotian Group
Hangzhou, Zhejiang, CHN

Dan Ou†
Xiaoyi Zeng
{oudan.od,yuanhan}@taobao.com
Taotian Group
Hangzhou, Zhejiang, CHN

Tong Xu†
Enhong Chen
{tongxu,cheneh}@ustc.edu.cn
University of Science and Technology of China
Hefei, Anhui, CHN

ABSTRACT

In the realm of e-commerce search, the significance of semantic matching cannot be overstated, as it directly impacts both user experience and company revenue. Query rewriting serves as an important technique to bridge semantic gaps inherent in the semantic matching process. However, existing query rewriting methods often struggle to effectively optimize long-tail queries and alleviate the phenomenon of *"few-recall"* caused by semantic gap. In this paper, we present **BEQUE**, a comprehensive framework that **B**ridges the s**E**mantic gap for long-tail **QUE**ries. BEQUE comprises three stages: multi-instruction supervised fine tuning (SFT), offline feedback, and objective alignment. Specifically, we first construct a rewriting dataset based on rejection sampling, and mix it with multiple auxiliary tasks data to fine tune our large language model (LLM) in a supervised fashion during the first stage. Subsequently, with the well-trained LLM, we employ beam search to generate multiple candidate rewrites, which would be fed into Taobao offline system to simulate the retrieval process and obtain the partial order. Leveraging the partial order of candidate rewrites, we introduce a contrastive learning method to highlight the distinctions between rewrites and align the model with the Taobao online objectives. Offline experiments prove the effectiveness of our method in enhancing retrieval performance. Online A/B tests reveal that our method can significantly boost gross merchandise volume (GMV), number of transaction (#Trans) and unique visitor (UV) for long-tail queries. BEQUE has been deployed on Taobao, one of most popular online shopping platforms in China, since October 2023.

*This work was done when the first author was an intern at Taobao Main Search.
†Corresponding Authors

TheWebConf'24 Companion, May 13 - 17, 2024, Singapore

ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/XXXXXXX.XXXXXXX

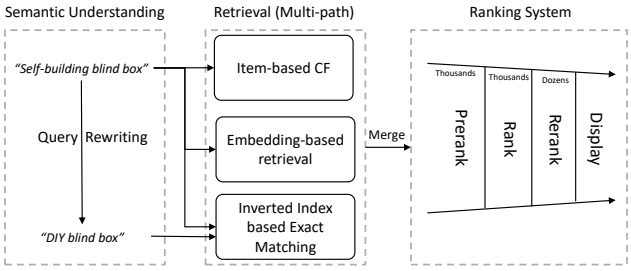

Figure 1: Framework of Taobao search engine.

CCS CONCEPTS

• **Information systems** → **Query reformulation**; • **Computing methodologies** → **Natural language processing**.

KEYWORDS

Query reformulation, large language models, semantic matching

ACM Reference Format:
Wenjun Peng, Guiyang Li, Yue Jiang, Zilong Wang, Dan Ou, Xiaoyi Zeng, Tong Xu†, and Enhong Chen. 2023. Large Language Model based Long-tail Query Rewriting in Taobao Search. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation emai (TheWebConf'24 Companion).* ACM, New York, NY, USA, 9 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Over the past several decades, the growth of e-commerce has been exceptionally rapid. Leading e-commerce companies such as Taobao, JD and Amazon have amassed hundreds of millions of users, generating billions of gross merchandise volume (GMV) annually. To facilitate the quick retrieval of related products for these users, a well-established search paradigm has been proposed. As illustrated in Figure 1, this paradigm involves several steps, including *"semantic understanding - retrieval - rank"*. Semantic understanding serves as the foundation of the entire system, ensuring accurate matching of user intent. However, due to the variations

in how users express their preferences for products, a semantic gap often exists between their queries and the product keywords. This issue becomes even more pronounced with long-tail queries, where the retrieval system may fail to provide any relevant products, significantly compromising the user's shopping experience. For instance, when a user intends to search for a *"DIY blind box"*, the query might inadvertently deviate to *"self-building blind box"* due to users' expression habits, which prevents term matching methods such as inverted index to match the semantic of *"DIY"* and non-customary term *"self-building"*. Therefore, optimizing the semantic gap for long-tail queries and addressing the problem of *"few-recall"* in e-commerce platform pose a significant necessity.

To address this problem, certain approaches[12, 18, 31, 39] have introduced the *"embedding-based retrieval"* paradigm within the process of semantic matching. In these approaches, the query and product are initially mapped to a common semantic space. Subsequently, the approximate nearest neighbor (ANN) is employed to retrieve the K most related products to the query. These methods demonstrate an efficient ability to identify a set of related products for each query, irrespective of the presence or absence of such a set. Nevertheless, it is worth noting that the retrieval outcomes obtained from these approaches are challenging to interpret and often give rise to suboptimal scenarios that cannot be improved. To enhance the controllability of retrieval outcomes, a group of researchers has employed the *"query rewriting - exact match"* paradigm. In certain research works[17, 40], query rewriting is regarded as a retrieval process aimed at finding similar rewrites from a document or query reformulation set. These rewrites are subsequently utilized to search for relevant products using sparse retrieval. Although this approach effectively expands the semantic of hot queries, it is not adequately optimized for long-tail queries, often leading to situations where no related rewrite can be generated. To tackle this issue, generative-based query rewriting methods[23, 34] have been proposed, which involve supervised training on query pair data to empower the model with rewriting capabilities. Furthermore, a alignment process[1, 21] is incorporated to enhance metric preference of the generation model. While these methods partially address the semantic gap problem, they typically rely on small generative models with limited comprehension of long-tail queries. Furthermore, the design of rewards in these methods does not fully consider the challenges associated with long-tail queries retrieval. Consequently, the rewriting capability of these methods is significantly constrained. Other approaches[2, 14, 32, 33] utilize LLMs as backbones and aim to enhance the semantic understanding by generating pseudo documents. However, these methods lack specific training for rewriting and do not undergo an learning process to align with the goals of e-commerce search.

In this paper, we propose BEQUE, a framework for long-tail query rewriting. The framework consists of three stages: multi-instruction supervised fine-tuning (SFT), offline system feedback, and objective alignment. In the first stage, we utilize the rejection sampling to collect <*query, rewrite*> pairs with desired quality distribution. We combine these <*query, rewrite*> pairs with data from quality classification, product title prediction and chain of thought (CoT) tasks to construct the multi-instruction rewriting dataset for fine-tuning our LLM. Next, with the well-trained LLM, we employ beam search to generate multiple candidate rewrites for each query. These candidate rewrites are fed into the Taobao offline system to retrieve a collection of related products. We calculate the quality score of retrieved products for rewrites and use them as rewards for candidates ranking. To highlight the differences between the candidate rewrites and maximize the probability of generating rewrites with high rankings, we introduce a Bradley-Terry based contrastive learning method that considers the partial order among the candidate rewrites. Ultimately, the model training objective is aligned with the online goal of the Taobao search, ensuring that the generated rewrites yield the desired search results.

The main contributions of this work are listed as follows:

- We have analyzed long-tail query in e-commerce search and identified the semantic gap problem associated with such queries.
- We propose a three-stage framework called BEQUE to address the issue of semantic gap in long-tail queries. This framework is designed to generate rewrites that align with the objectives of Taobao search.
- The effectiveness of our model is demonstrated through both offline and online experiments, showcasing its ability to significantly improve e-commerce revenue.

2 RELATED WORKS

2.1 Query Rewriting

Query rewriting, also known as query expansion or query reformulation, plays a pivotal role in e-commerce search technology and has a profound impact on the user's shopping experience and the revenue of e-commerce platforms. This technique can be broadly categorized into discriminative and generative methods.

Discriminative methods treat query rewriting as a retrieval process that expand the semantics of the original query by selecting appropriate terms from the candidate set. For example, pseudo-relevance [6, 28, 35] selects the top k documents from the initial retrieval as semantic extensions. These approaches, however, often pose challenges in effectively controlling the semantic scope and ensuring retrieval relevance. To address these challenges, one potential solution is to utilize a well-built thesaurus [5, 20] as a candidate rewrite set. However, it is important to note that the effectiveness of these methods highly depends on the quality of the thesaurus. Inadequate quality may result in query semantic drift, where the intended meaning of the query is compromised. Furthermore, alternative approaches [3, 8, 17, 19] involve generating candidate rewrites based on search logs, incorporating similar terms from users' search history as extensions. Unfortunately, due to the Matthew effect, search logs naturally exhibit a bias towards popular queries. Consequently, the training data collected through this approach may not meet the optimization requirements for long-tail queries, which are less frequently searched for.

Generative methods [15, 23, 26, 34] treat the rewriting task as a generative process, where candidate terms for the original query are produced by a transformer-like model. Furthmore, some methods [1, 21] incorporate reinforcement learning or contrastive learning to match human preferences or offline metrics. These methods have the ability to generate related rewrites for each query. However, they usually employ a model with limited number of parameters, which are prone to processing long-tail queries due to it pool

capability of semantic understanding. In addition, the rewrites they generate are often inconsistent with the optimization goals of the actual search engine. LLM-based rewriting methods [2, 14, 32, 33] provide a deeper understanding and can generate appropriate expansion for long-tail queries. Nevertheless they usually lack ad-hoc training on fine-tuning and goal alignment, which may introduce illusions and noise to the original query.

2.2 Preference Alignment

In recent years, with the increase in number of parameters, language models [10, 22, 30, 36] have demonstrated incredible semantic understanding and zero-shot capabilities on one hand. On the other hand, they have also faced the challenges of model hallucinations and ethical issues. These models can fabricate facts and mislead users by leveraging their extensive background knowledge. To align the model's outputs with human morals and preferences, reinforcement learning (RL) [7, 27] have introduced to force the model to learn the partial order among different outputs. OpenAI made a groundbreaking application [22] of RL to the training process of large language models, specifically with ChatGPT, which received tremendous attention. LLama2 [30] designed a multi-objective reward function that not only ensures the safety of model outputs but also enhances their helpfulness. Unfortunately, these methods often rely on complex training processes and are difficult to tune due to the abundance of hyperparameters. Therefore, rejection sampling-like methods [4, 9] have been proposed, which continue training the model by collecting outputs with high rewards from the previous rounds to align with human preferences. Moreover, contrastive learning methods [24, 25, 37] directly rank the outputs based on their rewards and utilize ranking loss to adjust the output probabilities, explicitly learning the partial order of outputs. Inspired by these methods, we combine them with our designed taobao metric to align with taobao's online objectives.

3 METHOD

3.1 Framework Overview

Long-tail query rewriting aims to expand the original query semantics to address the problem of semantic gap while ensuring relevance. To this end, as shown in Figure 2, we propose a three-stage rewriting framework, which consists of: multi-instructions supervised fine-tuning (SFT), offline feedback and objective alignment. 1) First, with rejeciton sampling, we constructed a multi-instructions SFT dataset based on online logs that focuses on rewriting tasks, mixed with quality classification tasks, query correction and chain of thought (CoT) to train rewriting-specific LLMs. 2) After that, we use the well-trained LLM obtained in the first stage to generate multiple candidate rewrites for each sampled query. In order to obtain the partial order of these candidate rewrites, we construct a taobao offline system to obtain search results for these rewrites. The quality scores of the search results are used to rank the candidates. 3) Based on the partial order of candidate rewrites, we calibrate the generation probability of these rewrites using Bradley-Terry based contrastive learning to maximize the probability of rewrites that can obtain the desired search results.

3.2 Multi-instruction SFT

Given that no publicly available LLMs are specifically designed for e-commerce query rewriting, direct utilization of these models to address the long-tail query semantic gap issue is likely to introduce inaccuracies and noise. Consequently, we have embarked on an approach wherein we gather various rewriting-related tasks to fine tune LLMs, enhancing their ability to comprehend and rewrite e-commerce queries effectively.

Query Rewriting Dataset: we initially source rewrites from Taobao previous-generation rewriting policy. This process yields the initial rewriting dataset. Specifically, when a user initiates a query x in Taobao search, old rewriting policy generates a list of rewrites queries $Y = \{y_1, y_2, \cdots, y_n\}$. From this list, we select the top-ranked y_1 as the gold standard candidate to include in our initial rewriting dataset $\mathcal{D}$ represented as:

$$\mathcal{D} = \left\{ \left(x^i, y^i\right)\Big|_{i=1}^{N} \text{ such that } x^i \sim p(x), y^i \sim \pi_{old}\left(y \mid x = x^i, \theta_{old}\right) \right\}, \tag{1}$$

where $p(x)$ denotes the query distribution in Taobao search engine, π_{old} and θ_{old} is the previous-generation rewriting policy of Taobao and it's parameters.

It's important to highlight that e-commerce query rewriting differs from other text generation tasks. In this context, semantic similarity between query and rewrite does not necessarily guarantee retrieval of similar sets of products. What we aim to achieve is a high relevance between the products retrieved by rewrite y and the original query x. To attain this, we apply a relevance filter to $\mathcal{D}$ through rejection sampling:

$$\mathcal{D}_r = \left\{ \left(x^i, y^i\right)\Big|_{i=1}^{N_r} \text{ such that } x^i, y^i \in \mathcal{D}, rele(x^i, y^i) > \tau^{rele} \right\}, \tag{2}$$

where $rele(\cdot)$ and τ^{rele} denote the relevance method and its threshold of *<query, rewrite>* pair. The detail of the function $rele(\cdot)$ is discussed in the Section 3.3

Furthermore, Taobao's previous-generation of rewriting models primarily lacks optimization for long-tail queries. As we work on the new generation of rewriting models, our goal is to maintain the relevance of retrievals while expanding the semantics. This expansion is aimed at alleviating the issue of long-tail queries leading to *"few-recall"* results. As a result, we utilize rejection sampling once again to filter $\mathcal{D}_r$ by considering the retrieval increment. Additionally, we include the most recent interacted product title of x as supplementary information to better address long-tail queries:

$$\begin{aligned} \mathcal{D}_{sft} = \{&(concat(x^i, \mathcal{E}_x), y^i)|_{i=1}^{N_{sft}} \\ &\text{such that } x^i, y^i \in \mathcal{D}_r, incr(x^i, y^i) > \tau^{incr}\}, \end{aligned} \tag{3}$$

where, $\mathcal{E}_x$ is the interacted product title list of query x, $incr(\cdot)$ and τ^{incr} denote the increment method and its threshold of query-rewrite pair, respectively. The detail of the function $incr(\cdot)$ is discussed in the Section 3.3.

Auxiliary Task Datasets: In order to further enhance LLMs' ability to comprehend long-tail queries, we have gathered three high-related task datasets in the context of query rewriting. These tasks encompass quality classification, product title prediction, and CoT.

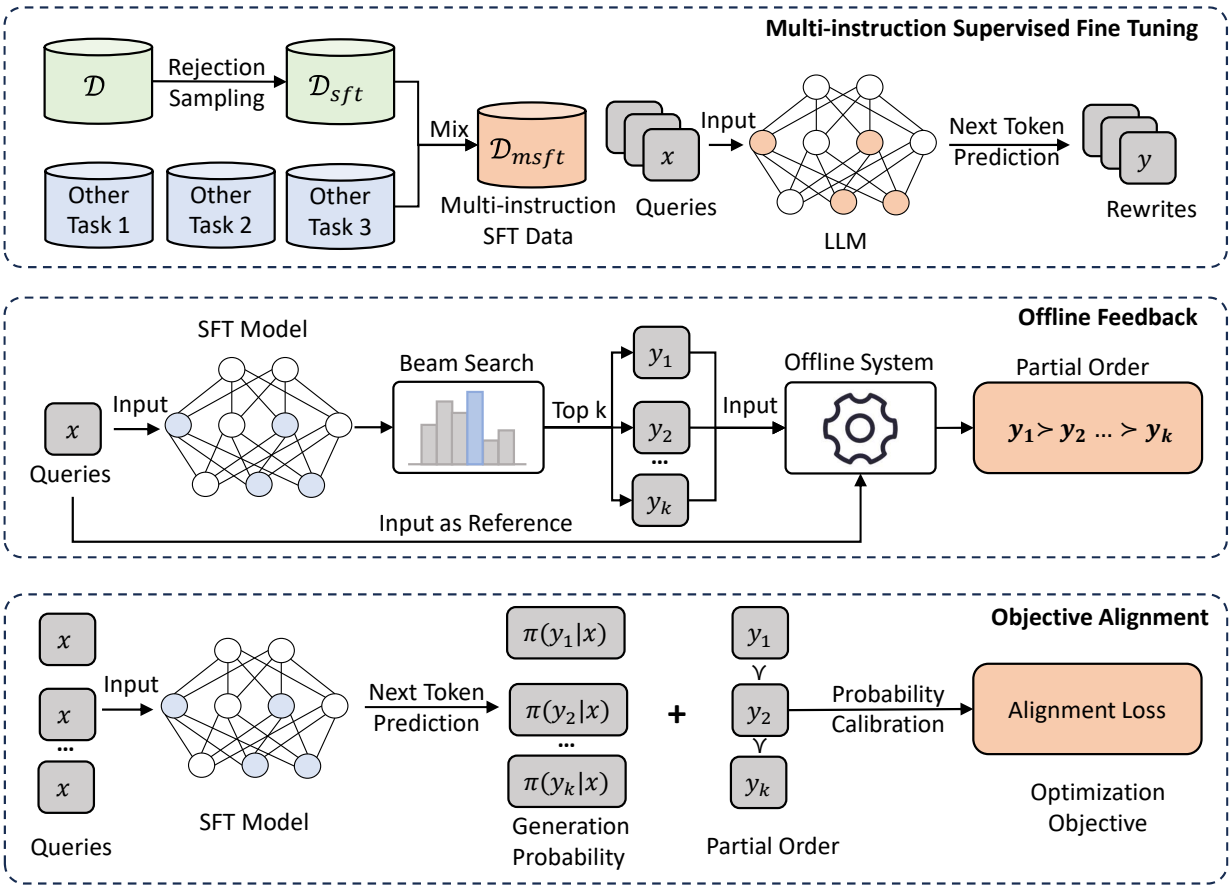

Figure 2: Framework of BEQUE.

1) For the quality classification task, our approach began with the extraction of query pairs from online logs. These query pairs were then subjected to human annotation to determine if they met the data requirements specified for SFT. 2) For the product title prediction task, we chose the most recent interacted product under the query as the reference, forming *<query, product title>* pairs. 3) As for the CoT task, we employed the original online queries to construct prompts for human evaluators. It's noteworthy that these evaluators were not only tasked with providing query rewrites aimed at improving the quality of query retrieval but were also expected to articulate their thought processes, explaining the rationale behind their specific revisions. The details prompt design for above auxiliary tasks are shown in Table 1. These datasets were subsequently incorporated into the rewriting task to construct the dataset for the SFT stage.

Supervised Fine Tuning: The process of generating text with a condition language model can be viewed as a constrained auto-regressive sampling strategy. Given a prompt x and its gold standard y, The training objective is to maximize the conditional probability $p(y|x)$. Considering our multi-instruction SFT dataset and assuming that $p(y|x) = \prod_{i=1} p(y_i|y_{0:i-1}, x)$, the training objective of rewriting model involves minimizing the negative log likelihood:

$$\mathcal{L}_{\text{SFT}}(\theta) = -\mathbb{E}_{(x,y)\sim\mathcal{D}_{msft}} \sum_{i=1} \log \pi\left(y_i \mid y_{0:i-1}, x; \theta\right), \quad (4)$$

where $\mathcal{D}_{msft}$ denotes multi-instruction SFT data, which consists of a mixture of query rewriting dataset $\mathcal{D}_{sft}$ and a variety of auxiliary task datasets, $\pi(\cdot)$ and θ denote our query rewriting model and its parameters. It is worth mentioning that LLMs typically have fixed prefixes in the prompt x. To avoid introducing noise, we disregarded the losses coressponding to x.

3.3 Offline Feedback

Currently, most alignment methods[1, 11, 21, 24] rely on manual annotation and training-based reward models. However, we argue that these approaches can be easily influenced by the quality of annotations and the effectiveness of the reward model training, which often leads to inaccurate reflection of response scores and compromises the learning of the generation model. To address this issue, we propose a feedback system based on the Taobao search engine, which provides more accurate rewrite scores.

The main structure of the Taobao online service process is illustrated in Figure 1. When a query is received by the Taobao search

Table 1: Prompt examples of different instructions.

Task	Prompt Example
Quality Classification	Is this a good e-commerce query rewrite? Query: {query} Rewrite: {rewrite} System: {Yes or No}
Title Prediction	Please generate product titles that match input query Query: {query} System: {Yes or No}
Chain of Thought	Your task is to rewrite the input query into a query that makes it easier to search for related products, and you are required to give the tho--ught process and then the query rewriting re--sult. The thought process and the query rewr--iting result are separated by a semicolon. Query: {query} System: {CoT}; {Rewrite}

engine, it undergoes preprocessing and is then passed to the query understanding. This module comprehends the semantic meaning of the query and send it to the retrieval module. The retrieval module retrieves a large number of candidate products from a massive Taobao product dataset. After deduplication and filtering, the candidate product sets from different retrieval paths are merged into an unordered and non-repetitive product set. These products are presented to the user through a complex ranking system.

Similarly, when our offline system receives a rewrite, it simulates the process of the Taobao online service to retrieve the corresponding products for the rewrite. Based on the product set, our system provides us with a quality score. It is important to note that we mainly address the semantic gap issue caused by long-tail queries in exact match. Therefore, our rewriting module only operates on the inverted index matching of the retrieval module, and the product set considered for rewrite retrieval is related only to the inverted index path. We propose three scores to measure the quality of the rewrite, namely relevance, increment, and hitrate.

As mentioned in Section 3.2 , even if the query and rewrite are semantically similar, it does not guarantee that the retrieved product sets will also be related. To prevent the model from retrieving products that are completely different from the user's original intent, we introduce the relevance score. It is calculated as follows:

$$rele(x,y) = \frac{\sum_i \mathbb{1}_{f(x,z_y^i)>\tau'}}{|\mathcal{Z}_y|}, z_y^i \in \mathcal{Z}_y, \tag{5}$$

where x and y denote the query and rewrite, respectively, $f(\cdot)$ is Taobao relevance function which is design for evaluate the relevance between item title and query text, τ' denotes the semantic relevance threshold of query-item pair, $\mathcal{Z}_y$ denote the offline retrieval product list of text y.

Furthermore, we need to ensure that the rewrite can expand the semantic meaning of the original query to some extent, avoiding the issue of *"few-recall."* Therefore, we introduce the increment score, calculated as follows:

$$incr(x,y) = \frac{\sum_i \mathbb{1}_{f(x,z_{xy}^i)>\tau'}}{\sum_i \mathbb{1}_{f(x,z_{x'}^i)>\tau'}}, z_{xy}^i \in \mathcal{Z}_e \cap (\mathcal{Z}_x \cup \mathcal{Z}_y), z_{x'}^i \in \mathcal{Z}_e \cap \mathcal{Z}_x, \tag{6}$$

where, similarly, x and y denote the query and rewrite, $f(\cdot)$, τ' is relevance function and threshold of query-item pair, $\mathcal{Z}_x$ and $\mathcal{Z}_y$ denote the offline retrieval product set of text x and y, respectively, $\mathcal{Z}_e$ is the excellent product set maintained by Taobao search group.

Lastly, we define the hitrate, which measures to what extent the rewrite can compensate the semantic gap of the original query. The calculation process is as follows:

$$hitrate(x,y) = \frac{|\mathcal{E} \cap (\mathcal{Z}_x \cup \mathcal{Z}_y)|}{\sum_i \mathbb{1}_{f(x,e^i)>\tau'}}, e^i \in \mathcal{E}, \tag{7}$$

where $\mathcal{E}$ is the collection of products that users have transacted outside of the search scenario, $\mathcal{Z}_x$ and $\mathcal{Z}_y$ denote the offline retrieval product set of text x and y. Assuming that a product, which is semantically related to the current user's query, is not transacted in the search. This indicates that the original query fails to retrieve the product. However, if the rewrite is able to return the product, then it implies that the product exists in the $\mathcal{Z}_x \cup \mathcal{Z}_y$. This proves that the rewrite can indeed compensate for the semantic gap of the original query. As a result, the hitrate can effectively reflect the model's ability to compensate for this semantic gap.

Overall, our proposed feedback system based on the Taobao search engine provides more accurate rewrite scores by considering relevance, increment, and hitrate. This helps improve the alignment process and ensures better learning of the generation model.

3.4 Objective Alignment

To avoid introducing bias through the reward model, we introduce the Preference Rank Optimization (PRO) [25], based on the Bradley Terry Model. This method aims to enforce the model to learn the rewrite partial order provided by offline feedback. According to the Bradley Terry Model, the probability of selecting an policy should be proportional to its corresponding reward. Given the partial order: $y_1 \succ y_2$, the preference probability can be expressed as:

$$P_{BT} = \frac{\exp(r(y_1,x))}{\exp(r(y_1,x)) + \exp(r(y_2,x))}, \tag{8}$$

where $r(\cdot)$ is the reward function, which is defined as the normalized log probability of the rewrite generated by the LLM in PRO.

PRO expands this pairwise partial order to a more general listwise partial order. Additionally, a temperature is introduced in order to capture the significance of the ranking based on the reward. PRO loss is represented by the following equation:

$$\mathcal{L}_{PRO}(\theta) = -\mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D}_{PRO}} \sum_{k=1}^{n-1} \log \frac{\exp(\frac{\pi_{PRO}(y_k|x;\theta)}{\mathcal{T}_k^k})}{\sum_{i=k}^{n} \exp(\frac{\pi_{PRO}(y_i|x;\theta)}{\mathcal{T}_k^i})}, \tag{9}$$

where $\mathcal{T}_k^i = 1/(r(y_k) - r(y_i))$ and $\mathcal{T}_k^k = \min_{i>k}(\mathcal{T}_k^i)$for ranking difference, $\mathcal{D}_{PRO}$ denotes the dataset for alignment, $\pi_{PRO}(\cdot)$ and θ denote the policy model and its parameters, n denotes the number

of candidate rewrites. In addition, we added SFT loss to maintain the model's ability to generate normal outputs:

$$\mathcal{L}_{ALIGN} = \mathcal{L}_{SFT} + \lambda \mathcal{L}_{PRO}. \tag{10}$$

3.5 Online Serving

Due to the constraints imposed by numerous parameters and the autoregressive prediction mode of large language models, it is almost impractical to deploy BEQUE online and meet the latency requirements of the Taobao search system. To address this issue, we utilize BEQUE offline to perform rewriting inference on torso and tail queries, which are defined as queries with retrieval results containing less than 70% related products or less than ten thousand products. The rewrites generated from these queries are stored in an online key-value graph, enabling quick online response. When a long-tail query matches the key-value graph, the search engine retrieves the corresponding rewrite. Subsequently, both the query and rewrite are tokenized into terms and used as keywords for inverted index matching to obtain a set of related products. The union of the query and rewrite retrieval sets forms the final candidate set of products for the ranking system. The offline inference of BEQUE covers 27% of the page views (PV) in Taobao's main search and has a minimal impact on the latency of the online retrieval system.

4 EXPERIMENTS

4.1 Datasets

Training Dataset: For the SFT training process, we extracted approximately 20 million records from the online rewriting logs on Taobao prior to September 2023 as the initial training data. To enhance the quality of the training data, we conducted two rounds of rejection sampling to obtain the query rewriting dataset, which consists of 419,806 pairs of *<query, rewrite>*. Additionally, we included 155,662 manually rewriting data in the dataset to ensure that the SFT model's rewriting adheres to human preferences. Finally, we combined 50,000 samples each from quality classification, product title prediction and CoT task with the query rewriting dataset to construct the Multi-instruction SFT dataset.

For the objective alignment training process, we randomly selected 10,000 queries from the query rewriting dataset to generate candidates. For each query, we used the SFT Model to generate 5 candidate queries. These 50,000 rewrites were then scored using the Taobao offline system. After removing any outliers, the alignment training dataset consisted of a total of 45,350 candidate rewrites.

Test Dataset: For the offline test set, we selected 14,981 queries from Taobao search logs to evaluate the model's performance. Among them, 50% of the queries were randomly sampled proportionally to the actual queries. Furthermore, to assess the model's capability to rewrite long-tail queries, we sampled the remaining 50% of the data from the long-tail queries.

4.2 Evaluation

Offline Metrics: We employ relevance (rele), increment (incr) and hitrate as offline metrics to valid the effectiveness of our method. These metrics are defined in Section 3.3.

Online Metrics: We introduce three key online metrics: GMV, #Trans, and UV, to evaluate the model's online performance. These metrics are defined as follows:

$$UV = \#\text{unique IP of visitors}, \tag{11}$$

$$GMV = \#\text{pay amount}, \tag{12}$$

$$\#Trans = \#\text{transaction}. \tag{13}$$

Table 2: Comparison of different LLMs trained on multi-instruction SFT dataset.

Method	#params	*rele* (%)	*incr* (%)	*hitrate* (%)
ChatGLM	6b	63.5	143.4	15.88
ChatGLM2.0	6b	63.2	105.0	14.39
Baichuan	7b	66.0	114.3	15.70
Qwen	7b	62.6	133.2	15.63

4.3 Implementation Details

Unless stated otherwise, the optimizer used for model training is AdamW. During the multi-instruction SFT stage, the model is trained for one epoch with the learning rate set to 1e-5. In the objective alignment stage, the model is trained for four epochs with the learning rate set to 1e-6. Additionally, the maximum length of the prompt for the rewriting task is set to 64, while for the rest of the tasks it is set to 256.

4.4 Offline Experiments

4.4.1 Results of Different LLMs. In this section, we present the comparison results of different LLMs as base models trained on the multi-instruction SFT dataset, as shown in Table 2. The base models include ChatGLM [10, 38], ChatGLM2.0 [10, 38], Baichuan [13], and Qwen [29]. The variability in the advantages of these base models can be attributed to disparities in their pre-training data and model architectures. Notably, ChatGLM exhibits superior performance in both increment and hitrate compared to the other three models. However, due to copyright constraints, we have to utilize Qwen as the base model to fulfill our business objectives. Furthermore, Qwen's pre-training incorporates e-commerce data, augmenting its comprehension of the e-commerce domain. This indicates a higher potential for enhancing Qwen's ability to rewrite long-tail queries.

4.4.2 Ablation Study of Multi-instruction SFT data. To investigate the impact of the multi-instruction SFT dataset on the rewriting ability of LLMs, an ablation study was conducted in this section. First, we randomly extracted query pairs from online logs, with a size similar to that of the multi-instruction SFT dataset, which were not subjected to rejection sampling and were not mixed with other task data. Then, we used the data to train the model and obtained Qwen w/o MI. As depicted in Table 3, it is evident that the multi-instruction SFT dataset significantly enhances the model's performance in terms of relevance, increment, and hitrate. This improvement can be attributed to the substantial enhancement in the quality of query pairs after rejection sampling. Furthermore, auxiliary tasks closely associated with query rewriting, such as

Table 3: Comparison of Qwen with/without auxiliary tasks. MI means multi-instruction

Method	*rele* (%)	*incr* (%)	*hitrate* (%)
Qwen w/o MI	61.4	109.6	14.58
Qwen	62.6	133.2	15.63

Table 4: Comparison of BEQUE with different number of contrast candidates.

Number	*rele* (%)	*incr* (%)	*hitrate* (%)
2	53.3	215.6	17.66
3	56.6	205.4	17.36
4	57.7	198.7	17.27
5	58.8	190.3	17.21

quality classification, product title prediction, CoT, etc., further augment the model's comprehension of query semantics, subsequently elevating the quality of the produced rewriting results.

4.4.3 Results of Different Contrast Number. In Table 4, we present the impact of varying contrastive numbers of candidates on model performance during the objective alignment stage. It can be observed that, as the number of candidates increases, the relevance shows a consistent improvement, while the increment and hitrate decrease. This outcome can be attributed to our modified PRO, where all candidates are treated as gold standard and SFT Loss is calculated for each of them. Consequently, a larger candidate pool implies an increase in the SFT loss weights and a decrease in the partial order learning weights, improving relevance of generated rewrites. Furthermore, there exists a trade-off between relevance and increment, where an increase in one metric necessitates sacrificing the other, leading to a negative impact on the increment metric. In addition, it is important to mention that hitrate can be regarded as the increment with weak relevance constraint, which explains its decrease in this context. Considering the balance of the three metrics, we select the model with contrast number of 4 to be best checkpoint for overall comparison.

4.4.4 Main Results. We compare BEQUE with multiple baselines, including CLE-QR, query2doc (Q2D), BART, Qwen, and RL-based LLM. CLE-QR [17] is the previous-generation query rewriter of Taobao search that generates semantic representations and retrieve related rewrites for each query based on contrastive learning. BART [16] is a powerful pre-trained generation model based on the encoder-decoder structure. We fine-tune it with query pairs from online logs to enhance its ability to rewrite e-commerce queries. Qwen [29] is a large-scale language model based on the decoder-only structure that contains 7 Billion parameters. Similarly, we fine-tune it with query pairs from online logs to enhance its ability to rewrite e-commerce queries. Furthermore, following the settings of [1], we introduce an RL-based LLM and utilize relevance, increment, and hitrate as rewards to encourage the RL model to align with the Taobao offline metrics, respectively. From analyzing the data presented in Table 5, the following conclusions can be drawn:

Generative models outperform discriminative models when rewriting "Torso" and "Tail" queries. For instance, when considering CLE-QR and BART, both models exhibit similar performance on "Top Queries" across three metrics. However, BART significantly outperforms CLE-QR in terms of hitrate and increment on "Torso Queries" and "Tail Queries" while maintaining relevance. This discrepancy arises because discriminative models like CLE-QR rely on existing queries in the search system as rewrite candidates, which are often biased towards top queries. As a result, torso and tail queries, which lack semantically similar top rewrites, do not receive related search candidates from CLE-QR. In contrast, BART's rewriting process is not restricted by the semantic scope of online queries, enabling it to generate rewrites that are not present in the search system's history. This allows BART to overcome the limitations of discriminative models and optimize torso and tail query rewriting problem.

LLMs exhibit superior long-tail semantic understanding capabilities compared to small models. Qwen and BART serve as examples, where Qwen, with its extensive parameter size, demonstrates stronger semantic expansion than BART in terms of hitrate and increment of "All Queries". Analyzing individual query slices, Qwen's improvement in hitrate and increment primarily occurs in the "Tail Queries", further validating the suitability of LLMs for long-tail query rewriting tasks.

Retrieval augmentation methods demonstrate limited semantic expansion capabilities. Comparing Q2D (ChatGPT) and BEQUE, Q2D (ChatGPT) maintains good retrieval relevance across all query slices but lacks sufficient semantic expansion capabilities, resulting in subpar increment and hitrate performance. Conversely, our BEQUE, which is specifically optimized for semantic expansion in rewriting, significantly enhances these two metrics.

The reinforcement learning (RL) may introduce bias and impact the effectiveness of the rewriting LLMs. Examining RL and BEQUE, RL process introduces a reward model to guide the base model's training. However, calculating the reward requires offline search system simulation, and the reward model may not accurately capture the search system's features, leading to reduced performance of RL models. In contrast, our BEQUE employs contrastive learning to explicitly learn the partial order of candidates, circumventing potential bias caused by the reward model. Ultimately, while minimizing the adverse impact on retrieval relevance, BEQUE substantially improves the model's increment and hitrate.

Different offline metrics work differently as rewards. For instance, considering the BEQUE framework, when it prioritizes relevance as its training objective, it exhibits a more conservative approach to bridge semantic gap. The improvements in both increment and hitrate tend to be challenging to achieve in this context. However, when the primary objective shifts to maximizing increment, the model demonstrates a significant capacity to enhance both the increment and hitrate of retrieval, effectively addressing the issue of *"few-recall"*. In such cases, a marginal decrease in relevance becomes an acceptable trade-off. When hitrate becomes the target, the model can effectively enhance both the increment and hitrate. Nevertheless, owing to the intricacies of the hitrate computation process, the model encounters difficulties in capturing the partial order among candidates. Consequently, the model's ability

Table 5: Overall performance of BEQUE with multiple baselines. The best results are in bold, and the second-best results are underlined. "Top Queries" are defined as queries with retrieval results containing more than 70% related products; "Torso Queries" are defined as queries with retrieval results containing 10%-70% related products; "Tail Queries" are defined as queries with retrieval results containing less than 10% related products.

Method	Top Queries			Torso Queries			Tail Queries			All Queries		
	rele	*incr*	*hitrate*	*rele*	*incr*	*hitrate*	*rele*	*incr*	*hitrate*	*rele*	*incr*	*hitrate*
CLE-QR	<u>73.4</u>	90.0	13.16	24.7	36.4	15.77	10.0	17.2	12.36	<u>69.6</u>	90.0	12.95
BART	67.5	106.0	13.26	27.9	130.0	17.70	8.4	27.5	13.59	62.2	100.0	13.56
Q2D(ChatGPT)	71.7	47.3	12.96	<u>35.3</u>	66.7	16.86	<u>12.8</u>	15.9	17.21	66.7	45.5	14.73
Qwen(SFT)	67.1	117.4	14.18	25.5	56.4	17.49	7.2	34.9	14.91	61.4	109.6	14.58
RL (rele)	**74.7**	48.6	12.66	**39.4**	2.1	15.60	**14.5**	8.1	11.36	**70.0**	45.1	12.42
RL (incr)	46.9	76.1	13.33	17.7	134.1	18.00	4.4	60.9	15.20	42.5	75.9	14.22
RL (hitrate)	56.0	176.8	15.10	6.2	117.3	<u>20.16</u>	2.2	51.0	18.22	49.6	162.8	15.75
BEQUE (rele)	69.3	174.5	15.04	27.3	160.9	19.29	4.1	<u>66.4</u>	<u>18.27</u>	62.3	164.0	16.43
BEQUE (incr)	64.6	**212.3**	**15.45**	20.5	**207.7**	**21.45**	2.4	**71.7**	**19.62**	57.7	**198.7**	**17.27**
BEQUE (hitrate)	69.3	<u>177.0</u>	<u>15.39</u>	18.4	<u>204.0</u>	19.78	5.0	62.5	18.22	62.1	<u>167.0</u>	<u>16.64</u>

Table 6: Online A/B test of BEQUE on Mobile Taobao Search. "all queries": every query in the test bucket counts, regardless of whether it has been rewritten or not. "covered queries": only rewritten queries count. "long-tail queries": rewritten and long-tail queries count. "few-recall queries": rewritten and few-recall queries count.

Online Traffic	GMV	#Trans	UV
all queries	+0.40%	+0.34%	+0.33%
covered queries	+2.96%	+1.36%	+1.22%
long-tail queries	+1.57%	+2.52%	+2.32%
"few-recall" queries	+18.66%	+5.90%	+6.25%

to expand semantic is diminished in comparison to the BEQUE that focuses on increment.

4.5 Online Experiments

To assess the actual online performance of BEQUE, we deployed it on Taobao search for a 14-day online test, during which we recorded the three key metrics in the Taobao search scene: GMV, #Trans, and UV. Table 6 reveals that BEQUE surpassed the previous-generation rewriting model CLE-QR by 0.4%, 0.34%, and 0.33% in terms of GMV, #Trans, and UV, respectively. This implies that BEQUE contributes millions of GMV to Taobao search. It's important to note that the overall performance mentioned here refers to all queries in the testing buckets. Since we inference offline, there are about 70% of online queries that do not hit our rewriting table. Even in these cases, our model still delivers remarkable enhancements. Additionally, for the queries covered (rewritten) by BEQUE (approximately 27% of total PV), there were noteworthy increases of 2.96%, 1.36%, and 1.22% in GMV, #Trans, and UV, respectively. These findings indicate that BEQUE effectively rewrites queries and addresses potential semantic gaps in the semantic matching process. Moreover, BEQUE significantly improved online #Trans and UV for long-tail queries and *"few-recall"* queries, although we disregarded the GMV fluctuation for this subset due to its low proportion. This improvement can be attributed to our specialized optimization for long-tail queries. During the first-stage SFT of BEQUE, rejection sampling and auxiliary task data enhanced the model's performance in terms of retrieval increment and relevance, and also deepened its understanding of long-tail queries. The alignment process in the second and third stages effectively compelled the model to align with online objectives of Taobao search.

5 CONCLUSION

In this paper, we introduce BEQUE, a framework specifically designed for e-commerce query rewriting. The main objective of BEQUE is to address the semantic gap that occurs during the semantic matching process, particularly for long-tail queries. Initially, we improve the quality of the rewriting dataset by employing rejection sampling and auxiliary task mixing. We then train a LLM using this refined dataset, which enhances the model's query understanding and enables effective rewriting of e-commerce queries. Utilizing the well-trained LLM, we generate multiple candidate rewrites for each sampled query. To establish a partial order among these candidates, we create an offline feedback system based on online Taobao search. This feedback system accurately evaluates the retrieval quality of the candidate rewrites from various perspectives, such as relevance, increment, and hitrate. Finally, by incorporating the partial order of rewriting retrieval quality, we introduce PRO, which aligns the model's objectives with those of Taobao. This ensures that our approach generates rewriting results that yield high-quality retrieval outcomes. Through multiple experiments, we have demonstrated the effectiveness of our approach in improving offline metrics. Additionally, online A/B experiments have substantiated a significant increase in Taobao Search's GMV, #Trans, and UV, particularly for long-tail queries.

REFERENCES

[1] Sanjay Agrawal, Srujana Merugu, and Vivek Sembium. 2023. Enhancing e-commerce product search through reinforcement learning-powered query reformulation. In *CIKM*.

[2] Avishek Anand, Abhijit Anand, Vinay Setty, et al. 2023. Query Understanding in the Age of Large Language Models. *arXiv preprint arXiv:2306.16004* (2023).

[3] Ioannis Antonellis, Hector Garcia-Molina, and Chi-Chao Chang. 2008. Simrank++ query rewriting through link analysis of the clickgraph (poster). In *TheWebConf*. 1177–1178.

[4] Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. 2022. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073* (2022).

[5] Jagdev Bhogal, Andrew MacFarlane, and Peter Smith. 2007. A review of ontology based query expansion. *Information processing & management* 43, 4 (2007), 866–886.

[6] Guihong Cao, Jian-Yun Nie, Jianfeng Gao, and Stephen Robertson. 2008. Selecting good expansion terms for pseudo-relevance feedback. In *SIGIR*. 243–250.

[7] Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. *NIPS* 30 (2017).

[8] Hang Cui, Ji-Rong Wen, Jian-Yun Nie, and Wei-Ying Ma. 2002. Probabilistic query expansion using query logs. In *TheWebConf*. 325–332.

[9] Yuntian Deng, Anton Bakhtin, Myle Ott, Arthur Szlam, and Marc'Aurelio Ranzato. 2020. Residual energy-based models for text generation. *arXiv preprint arXiv:2004.11714* (2020).

[10] Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. 2022. GLM: General Language Model Pretraining with Autoregressive Blank Infilling. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 320–335.

[11] Caglar Gulcehre, Tom Le Paine, Srivatsan Srinivasan, Ksenia Konyushkova, Lotte Weerts, Abhishek Sharma, Aditya Siddhant, Alex Ahern, Miaosen Wang, Chenjie Gu, et al. 2023. Reinforced self-training (rest) for language modeling. *arXiv preprint arXiv:2308.08998* (2023).

[12] Jui-Ting Huang, Ashish Sharma, Shuying Sun, Li Xia, David Zhang, Philip Pronin, Janani Padmanabhan, Giuseppe Ottaviano, and Linjun Yang. 2020. Embedding-based retrieval in facebook search. In *SIGKDD*. 2553–2561.

[13] Baichuan Inc. 2023. A large-scale 7b pretraining language model developed by baichuan-inc. https://github.com/baichuan-inc/Baichuan-7B

[14] Rolf Jagerman, Honglei Zhuang, Zhen Qin, Xuanhui Wang, and Michael Bendersky. 2023. Query Expansion by Prompting Large Language Models. *arXiv preprint arXiv:2305.03653* (2023).

[15] Mu-Chu Lee, Bin Gao, and Ruofei Zhang. 2018. Rare query expansion through generative adversarial networks in search advertising. In *SIGKDD*. 500–508.

[16] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In *ACL*. 7871–7880.

[17] Sen Li, Fuyu Lv, Taiwei Jin, Guiyang Li, Yukun Zheng, Tao Zhuang, Qingwen Liu, Xiaoyi Zeng, James Kwok, and Qianli Ma. 2022. Query Rewriting in TaoBao Search. In *CIKM*. 3262–3271.

[18] Sen Li, Fuyu Lv, Taiwei Jin, Guli Lin, Keping Yang, Xiaoyi Zeng, Xiao-Ming Wu, and Qianli Ma. 2021. Embedding-based product retrieval in taobao search. In *SIGKDD*. 3181–3189.

[19] Saurav Manchanda, Mohit Sharma, and George Karypis. 2019. Intent term weighting in e-commerce queries. In *CIKM*. 2345–2348.

[20] Aritra Mandal, Ishita K Khan, and Prathyusha Senthil Kumar. 2019. Query Rewriting using Automatic Synonym Extraction for E-commerce Search.. In *eCOM@ SIGIR*.

[21] Akash Kumar Mohankumar, Nikit Begwani, and Amit Singh. 2021. Diversity driven query rewriting in search advertising. In *SIGKDD*. 3423–3431.

[22] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback, 2022. *arXiv preprint arXiv:2203.02155* 13 (2022).

[23] Yiming Qiu, Kang Zhang, Han Zhang, Songlin Wang, Sulong Xu, Yun Xiao, Bo Long, and Wen-Yun Yang. 2021. Query Rewriting via Cycle-Consistent Translation for E-Commerce Search. In *ICDE*. IEEE, 2435–2446.

[24] Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D Manning, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *arXiv preprint arXiv:2305.18290* (2023).

[25] Feifan Song, Bowen Yu, Minghao Li, Haiyang Yu, Fei Huang, Yongbin Li, and Houfeng Wang. 2023. Preference ranking optimization for human alignment. *arXiv preprint arXiv:2306.17492* (2023).

[26] Zhenqiao Song, Jiaze Chen, Hao Zhou, and Lei Li. 2021. Triangular Bidword Generation for Sponsored Search Auction. In *WSDM*. 707–715.

[27] Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. 2020. Learning to summarize with human feedback. *NIPS* 33 (2020), 3008–3021.

[28] Tao Tao and ChengXiang Zhai. 2006. Regularized estimation of mixture models for robust pseudo-relevance feedback. In *SIGIR*. 162–169.

[29] Qwen Team. 2023. *QWEN TECHNICAL REPORT*. Technical Report. Alibaba Group. https://qianwen-res.oss-cn-beijing.aliyuncs.com/QWEN_TECHNICAL_REPORT.pdf

[30] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288* (2023).

[31] Binbin Wang, Mingming Li, Zhixiong Zeng, Jingwei Zhuo, Songlin Wang, Sulong Xu, Bo Long, and Weipeng Yan. 2023. Learning Multi-Stage Multi-Grained Semantic Embeddings for E-Commerce Search. In *TheWebConf*. 411–415.

[32] Liang Wang, Nan Yang, and Furu Wei. 2023. Query2doc: Query Expansion with Large Language Models. *arXiv preprint arXiv:2303.07678* (2023).

[33] Shuai Wang, Harrisen Scells, Bevan Koopman, and Guido Zuccon. 2023. Can ChatGPT write a good boolean query for systematic review literature search? *arXiv preprint arXiv:2302.03495* (2023).

[34] Yaxuan Wang, Hanqing Lu, Yunwen Xu, Rahul Goutam, Yiwei Song, and Bing Yin. 2021. QUEEN: Neural query rewriting in e-commerce. (2021).

[35] Jinxi Xu and W Bruce Croft. 2017. Quary expansion using local and global document analysis. In *Acm sigir forum*, Vol. 51. ACM New York, NY, USA, 168–175.

[36] Aiyuan Yang, Bin Xiao, Bingning Wang, Borong Zhang, Chao Yin, Chenxu Lv, Da Pan, Dian Wang, Dong Yan, Fan Yang, et al. 2023. Baichuan 2: Open Large-scale Language Models. *arXiv preprint arXiv:2309.10305* (2023).

[37] Zheng Yuan, Hongyi Yuan, Chuanqi Tan, Wei Wang, Songfang Huang, and Fei Huang. 2023. Rrhf: Rank responses to align language models with human feedback without tears. *arXiv preprint arXiv:2304.05302* (2023).

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

[39] Han Zhang, Songlin Wang, Kang Zhang, Zhiling Tang, Yunjiang Jiang, Yun Xiao, Weipeng Yan, and Wen-Yun Yang. 2020. Towards personalized and semantic retrieval: An end-to-end solution for e-commerce search via embedding learning. In *SIGIR*. 2407–2416.

[40] Zhi Zheng, Kai Hui, Ben He, Xianpei Han, Le Sun, and Andrew Yates. 2020. BERT-QE: Contextualized Query Expansion for Document Re-ranking. In *EMNLP*. 4718–4728.