

Retrieving More Concrete Product Reviews by Query Rewriting with Retrieved Review Concretization

Tomoya Fukui¹[0009-0005-4612-285X], Takehiro Yamamoto¹[0000-0003-0601-3139],
and Takayuki Yumoto¹[0009-0005-1854-0448]

University of Hyogo, Kobe, Japan
ad24a050@guh.u-hyogo.ac.jp
t.yamamoto@sis.u-hyogo.ac.jp
yumoto@sis.u-hyogo.ac.jp

Abstract. In this study, we propose a method for retrieving concrete reviews for a product, using an abstract review as a query. However, it is difficult to retrieve concrete reviews from keywords or abstract reviews, as simple sparse or dense retrieval methods cannot account for concreteness. Therefore, we propose a method called Query Rewriting with Retrieved Review Concretization (QR-ReReC), which utilizes retrieved reviews to rewrite the original query. For the experiment, we constructed a dataset to evaluate the effectiveness of QR-ReReC. The results showed that QR-ReReC is more effective for retrieving more concrete reviews than the retrieval methods without query rewriting and pseudo relevance feedback.

Keywords: Information Retrieval · Query Rewriting · LLM

1 Introduction

In recent years, a vast number of user reviews have been posted on e-commerce sites such as eBay and Amazon. *Concrete* reviews are particularly beneficial for understanding a product. However, reviews on e-commerce sites also include *abstract* reviews that are insufficient for understanding a product. Moreover, formulating a query to find such concrete reviews is also challenging, as the user is not necessarily an expert on the product in question, and using an abstract review directly as a query is ineffective, as simple sparse or dense retrieval methods do not account for “concreteness”.

Therefore, we propose a method for retrieving more concrete reviews for a given abstract review; that is, a retrieval method that considers concreteness. To address this challenge, we adopt an approach that rewrites the abstract review to concrete reviews using a Large Language Model (LLM) [2]. However, the process may generate content containing vocabulary that does not exist in the actual documents, potentially affecting retrieval performance [2].

Therefore, we propose Query Rewriting with Retrieved Review Concretization (QR-ReReC), a concreteness-aware query rewriting method using an LLM.

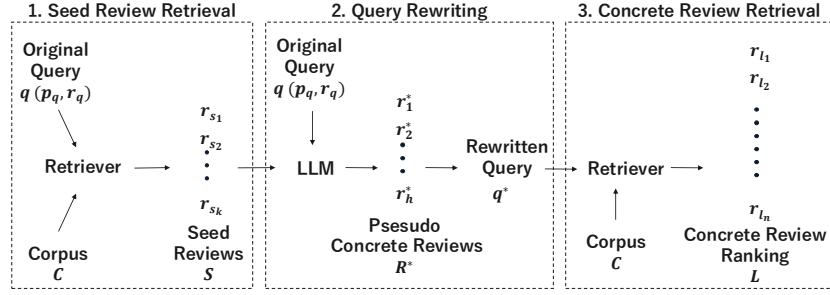


Fig. 1. Overview of our Query Rewriting with Retrieved Review Concretization (QR-ReReC).

We first retrieve actual reviews from the retrieval corpus based on the abstract review. We then provide the top- k retrieved reviews and the original abstract review to the LLM to generate pseudo-concrete reviews as the rewritten query. This process aims to limit the generation of vocabulary not present in the actual reviews.

To verify the effectiveness of our proposed method for concreteness-aware retrieval, we prepared an evaluation dataset comprising 50 abstract reviews, which serve as queries. In our experiments, using this dataset, we compared QR-ReReC against three baseline methods. The results of the experiment showed that although the difference was not statistically significant, QR-ReReC is more effective than LLM-based query rewriting without retrieved review.

2 Query Rewriting with Retrieved Review Concretization for Retrieving More Concrete Reviews

2.1 Problem Definition

This study defines the task of retrieving more concrete reviews for a given abstract review. Following Tanaka et al. [5], we define “concrete” to mean that the product’s usage context can be clearly and easily imagined.

Simply using an abstract review as a query to find more concrete information can be ineffective, as it is unclear which aspect of the abstract review the user wants to learn more about. Therefore, the original query is defined to consist not only of the abstract review but also of a *target phrase* within that abstract review. This target phrase explicitly indicates the specific part of the abstract review that the user wishes to see elaborated with more concrete details.

Let us define the problem to be tackled in this paper. Given a product review corpus $C = \{r_1, r_2, \dots, r_n\}$, and a original query $q = (p_q, r_q)$, which is a combination of an abstract review $r_q \in C$ and a target phrase p_q . The goal is to obtain a ranked list $L = [r_{l_1}, r_{l_2}, \dots, r_{l_n}]$. Here, r_{l_i} (for $1 \leq i \leq n$) is the review at rank i . The subscript l_i denotes the index of this review in the original corpus C .

Reviews are ranked higher if they are both relevant to the target phrase p_q and allow the product’s usage to be more clearly imagined (i.e., are more concrete).

2.2 Method Overview

As shown in Figure 1, QR-ReReC consists of three main steps: Seed Review Retrieval, Query Rewriting, and Concrete Review Retrieval. In Seed Review Retrieval, a retrieval is performed using the original query q to obtain actual reviews (seed reviews) that will be used for the query rewriting. In Query Rewriting, pseudo-concrete reviews are generated by an LLM based on the seed reviews S , along with the abstract review r_q , the target phrase p_q , and the product name. Then, a rewritten query q^* is derived from the set of pseudo-concrete reviews R^* . In Concrete Review Retrieval, the retrieval is performed based on the rewritten query q^* to obtain concrete reviews from the corpus C .

2.3 Query Rewriting

In the Query Rewriting step, we first prompt an LLM to generate a set of h pseudo-concrete reviews, denoted as $R^* = \{r_1^*, r_2^*, \dots, r_h^*\}$. This set is then used to rewrite the original query q into a rewritten query q^* .

The prompt consists of the abstract review r_q , the target phrase p_q , the product name, and the seed reviews S obtained in the Seed Review Retrieval step. The prompt used is shown below.

Please generate $<h>$ more concrete reviews for the following product based on the provided information. Note: Please write more concrete reviews, focusing only on the aspect mentioned in the “<Target Phrase p_q >”. Here, “concrete” means that the usage context can be clearly imagined. If necessary, please refer to the following reviews.

###Reference Reviews

sample_review 1: $<r_{s_1} \in S>$

$\vdots$

sample_review k : $<r_{s_k} \in S>$

###Product

$<\text{Product Name}>$

###Review

$<\text{Abstract Review } r_q>$

###Output

We believe that providing such reference reviews in the prompt makes it easier for the model to generate vocabulary that exists in the actual review corpus.

Then, we derive the rewritten query q^* from the set of pseudo-concrete reviews R^* to be used in the Concrete Review Retrieval step. The method for deriving the rewritten query q^* differs depending on whether the retrieval model is a sparse or a dense retrieval model. For sparse retrieval, to expand the retrieval vocabulary, the concatenation of the pseudo-concrete reviews R^* is used as the

rewritten query q^* . For dense retrieval, similar to HyDE [2], the average vector of the pseudo-concrete reviews R^* is used as the rewritten query q^* .

3 Experiments

3.1 Evaluation Data Creation

For our evaluation data, we used the Rakuten Ichiba product review data from the Rakuten Dataset [4]. Rakuten Ichiba is one of Japan’s e-commerce sites¹. All our experiments were conducted in Japanese. We used reviews for 25 home appliance products from the dataset as our evaluation data. For each product, we prepared two queries, for a total of 50 queries. Each original query q consists of a pair: an abstract review r_q and a target phrase p_q .

We used nDCG to evaluate the performance of our methods. The gain for each review $r_i \in C$, required for the nDCG calculation, was determined by a binary evaluation of whether the review’s content was easier to imagine compared to the abstract review r_q with respect to the target phrase p_q . The evaluation data was created by three annotators, and the score for each review was determined by a majority vote of their judgments. As a result of the annotation, Fleiss’ kappa was 0.460, indicating moderate agreement [3].

3.2 Comparison Methods

To demonstrate the effectiveness of our proposed method, we conducted experiments using the following methods.

Review Retrieval without Query Rewriting (Original Query): A retrieval is performed using the original query q , which is a concatenation of the abstract review r_q and the target phrase p_q . Note that this method is the same as the one in the Seed Review Retrieval.

Review Retrieval with Pseudo Relevance Feedback (PRF) [1]: First, a retrieval is performed in the same way as the Original Query method. Next, the top 5 retrieval results are considered a set of relevant reviews R_+ . A new query q^* is derived from the union of the original query q and the relevant reviews R_+ using the same procedure as the proposed method. Finally, a retrieval is performed using the resulting query q^* .

Query Rewriting without Retrieved Review Concretization

(QR-w/oReReC): This method performs query rewriting using an LLM without using the seed reviews S . Specifically, it generates 5 pseudo-concrete reviews based only on the abstract review r_q , the target phrase p_q , and the product name, then performs query rewriting and retrieval. The rewritten query was derived from the set of pseudo-concrete reviews in the same way as the proposed method.

Query Rewriting with Retrieved Review Concretization (QR-ReReC):

This is our proposed method. Based on preliminary experiments, the number of

¹ <https://www.rakuten.co.jp/>

Table 1. Mean nDCG@k for each method in sparse retrieval (BM25). Bold values indicate the best performance for each nDCG@k metric. Superscripts indicate a statistically significant improvement ($p < 0.05$) over the methods corresponding to the letter. The superscripts o, p, w, and c correspond to Original, PRF, QR-w/oReReC, and QR-ReReC, respectively.

	nDCG@1	nDCG@2	nDCG@3	nDCG@5	nDCG@10
Original	0.220	0.189	0.184	0.179	0.231
PRF	0.240	0.240	0.210	0.193	0.240
QR-w/oReReC	0.280	0.311	0.298	0.310^{op}	0.338^{op}
QR-ReReC	0.320	0.328^o	0.323^{op}	0.304 ^{op}	0.337 ^{op}

Table 2. Mean nDCG@k for each method in dense retrieval (ruri-large). Bold and superscripts follow the same notation as in Table 1.

	nDCG@1	nDCG@2	nDCG@3	nDCG@5	nDCG@10
Original	0.180	0.134	0.144	0.161	0.188
PRF	0.120	0.143	0.145	0.157	0.185
QR-w/oReReC	0.220	0.205	0.219	0.228 ^{op}	0.254 ^{op}
QR-ReReC	0.280	0.280^{op}	0.294^{op}	0.273^{op}	0.299^{op}

pseudo-concrete reviews to generate, h , was set to 5, and the number of seed reviews used for Query Rewriting, k , was set to 2.

3.3 Experimental Setup

For each method, we test sparse and dense retrieval models. For the sparse retrieval model, we use BM25 from the Python library `rank_bm25`. For the dense retrieval model, we use ruri-large² [6], a BERT-based text embedding model specialized for Japanese. For the LLM-based query rewriting methods (QR-w/oReReC, QR-ReReC), we use `elyza/Llama-3-ELYZA-JP-8B`³. To prioritize experimental reproducibility, we configured the model for deterministic generation by setting its parameter `do_sample=False`, which ensures the same output for the same input.

For each method, we perform a retrieval based on the abstract review r_q and the target phrase p_q . The performance of each method is evaluated with nDCG@k, where k is set to 1, 2, 3, 5, and 10. For statistical testing, we used a paired t-test to check for statistically significant differences between pairs of the four methods. We used a significance level of 5% and applied Holm’s correction for multiple comparisons.

² <https://huggingface.co/cl-nagoya/ruri-large>

³ <https://huggingface.co/elyza/Llama-3-ELYZA-JP-8B>

3.4 Results

Tables 1 and 2 show the nDCG@k results for sparse and dense retrieval. In both settings, LLM-based query rewriting (QR-w/oReReC and QR-ReReC) consistently outperformed non-LLM baselines (Original and PRF).

For sparse retrieval, QR-ReReC showed statistically significant improvements over Original and PRF at nDCG@3, 5, and 10. For dense retrieval, the trend was even clearer. QR-ReReC significantly outperformed both Original and PRF across a broad range of metrics (nDCG@2, 3, 5, and 10).

4 Conclusion and Future Work

To retrieve concrete product reviews from an abstract review, we proposed Query Rewriting with Retrieved Review Concretization (QR-ReReC), a method that uses seed reviews to guide an LLM in rewriting a query for concreteness. Our experiments demonstrated that QR-ReReC is more effective than retrieval without query rewriting and pseudo relevance feedback. Furthermore, while not statistically significant, our retrieval-guided approach also proved more effective than the LLM-based rewriting method without such guidance. Future work includes validating the hyperparameters in the proposed method, such as the number of reviews to generate and the number of actual reviews to provide; and developing filtering mechanisms for seed reviews.

Acknowledgments

This work was supported by JSPS KAKENHI Grant Numbers JP24K03228, and JP25K03229. We also used “Rakuten Dataset” (<https://rit.rakuten.com/data.release/>) provided by Rakuten Group, Inc. via IDR Dataset Service of National Institute of Informatics.

References

1. Buckley, C., Singhal, A., Mitra, M., Salton, G.: New retrieval approaches using SMART: TREC 4. In: TREC-4. pp. 25–48 (1995)
2. Gao, L., Ma, X., Lin, J., Callan, J.: Precise Zero-Shot Dense Retrieval without Relevance Labels. In: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 1762–1777 (2023)
3. Landis, J.R., Koch, G.G.: The Measurement of Observer Agreement for Categorical Data. *Biometrics* **33**(1), 159–174 (1977)
4. Rakuten Group, Inc: Rakuten Ichiba data. Informatics Research Data Repository, National Institute of Informatics. (dataset). (2020), <https://doi.org/10.32130/idr.2.1>
5. Tanaka, S., Jatowt, A., Kato, M.P., Tanaka, K.: Estimating content concreteness for finding comprehensible documents. In: Proceedings of the Sixth ACM International Conference on Web Search and Data Mining. pp. 475–484 (2013)
6. Tsukagoshi, H., Sasano, R.: Ruri: Japanese General Text Embeddings. arXiv preprint arXiv:2409.07737 (2024)