Proceedings of Machine Learning Research **318** (2026)
The 39th Canadian Conference on Artificial Intelligence

Canadian AI 2026
25–29 May 2026, Vancouver, BC

Query Refinement in Dense Retrieval Using LLM-Driven Relevance Feedback

Seyedehfatemeh Karimi[1], Maryam Khodabakhsh [2], Havva Alizadeh Noughabi[3], Fattane Zarrinkalam[4,*]
[1] Department of Computer Engineering, Ferdowsi University of Mashhad
[2] Faculty of Computer Engineering, Shahrood University of Technology
[3] Faculty of Engineering, University of Gonabad
[4] College of Engineering, University of Guelph

Abstract

Dense retrieval methods, which encode queries and documents into a shared semantic embedding space, have achieved strong performance in information retrieval tasks. However, their effectiveness diminishes in scenarios with limited or no domain-specific training data. To mitigate this limitation, recent approaches have leveraged large language models (LLMs) for query refinement in unsupervised dense retriever systems. A promising direction within this line of research involves using LLMs to assess the relevance of initially retrieved documents, and then incorporating the resulting relevance feedback to update the query embedding. Despite promising early results, a systematic investigation of how different prompting strategies and query update mechanisms influence retrieval performance remains absent. In this study, we explore four prompting strategies—Zero-Shot, Few-Shot, Role-Playing, and Chain-of-Thought—to guide LLMs in performing relevance judgments. Furthermore, we evaluate various query update formulas that utilize embeddings of LLM-identified relevant documents to refine query representations. Our experiments, conducted on two datasets and using two open-source LLMs, demonstrate that carefully crafted prompting combined with effective query updates can substantially enhance retrieval performance. These findings provide valuable insights for optimizing LLM-guided relevance feedback in unsupervised dense retrieval. All code and datasets are available at https://github.com/ftmkm97/ReFeed-IR.git.

Keywords: Dense Retrieval, Relevance Feedback, Large Language Models

1. Introduction

Recent progress in transformer-based language models has led to notable improvements in *dense retrieval methods* [1, 2], which represent queries and documents within a shared semantic embedding space [3]. While dense retrieval models have demonstrated significant advantages over traditional exact term matching methods such as BM25 [4], their deployment remains challenging in low-resource settings where substantial training data is lacking [5, 6].

To address the limitations posed by the lack of domain-specific training data, recent research has explored *unsupervised* dense retrieval methods that leverage Large Language Models (LLMs) to generate hypothetical documents. These generated documents serve as proxies to retrieve the most semantically similar real documents. For instance, HyDE [7] prompts an LLM to generate hypothetical passages based on the query, which are then used to retrieve relevant documents. However, such approaches heavily depend on the parametric knowledge stored within the LLM, which may limit their applicability in out-of-domain or proprietary corpora [8]. To overcome this, recent studies have proposed alternative approaches that leverage LLMs for relevance estimation rather than hypothetical document generation. For example, the ReDE-RF framework [8] begins by retrieving an initial set of candidate documents through a fully unsupervised dense retrieval. These documents are then presented to an LLM, which is prompted to classify them as either relevant or non-relevant to the query. Based on the documents identified as relevant, their embeddings are

* fzarrink@uoguelph.ca

extracted and used to construct an updated query representation. Notably, this approach does not require the LLM to generate any new textual content. As a result, the revised query vector benefits from LLM-guided semantic refinement while remaining grounded in the actual data, making it more robust for out-of-domain applications.

Despite promising early results from recent work on unsupervised dense retrieval methods that leverage LLMs for relevance feedback, a systematic investigation into the impact of different prompting strategies and query update mechanisms on retrieval performance remains lacking. In this paper, we explore the effectiveness of four prompting strategies—*Zero-Shot*, *Few-Shot*, *Role-Playing*, and *Chain-of-Thought*—to guide LLMs in making relevance judgments. Furthermore, we investigate the impact of various query vector update formulations, wherein the updated query is computed using the embeddings of documents identified as relevant through LLM-based feedback.

To summarize, our contributions are threefold: (1) we conduct a comprehensive study of prompting strategies for LLM-guided relevance estimation in unsupervised dense retrieval; (2) we propose and compare multiple formulations for query refinement based on LLM-generated feedbacks; and (3) we perform extensive experiments on two benchmark datasets—TREC DL19 [9] and TREC DL20 [10]—as well as on several low-resource retrieval datasets from BEIR [6]—evaluating performance across two LLMs, Mistral and Llama.

2. Methodology

In this section, we describe our approach to enhancing unsupervised dense retrieval with LLM-guided relevance feedback. Following [8], we first retrieve top-k candidate documents $\mathcal{D}$init for a query q using a standard dense retriever (e.g., *Contriever* [11]). An LLM then evaluates the relevance of each document, and the subset of relevant documents $\mathcal{D}\text{rel} \subseteq \mathcal{D}_{\text{init}}$ is used to refine the query representation. The prompting strategies and query update mechanisms are detailed in Subsections 2.1 and 2.2, respectively.

2.1. Prompting Strategies

Given a query q and its initially retrieved candidate documents $\mathcal{D}_{\text{init}}$, we aim to leverage a LLM to estimate the relevance of each document $d_i \in \mathcal{D}_{\text{init}}$ with respect to q. Unlike ReDE-RF [8], which uses binary relevance labels, we adopt a four-point scale: 0 (Irrelevant), 1 (Related), 2 (Highly relevant), and 3 (Perfectly relevant), enabling finer-grained feedback for query refinement. We evaluate four main prompting strategies: *Zero-Shot* [12], *Few-Shot* [13], *Role-Playing* [14], and *Chain-of-Thought* [12]. Additionally, we introduce a *hybrid prompting* strategy that combines Few-Shot, CoT, and Role-Playing elements to guide LLM relevance judgments more effectively. Prompt templates are provided in Appendix A.

2.2. Query Refinement.

After identifying relevant documents $\mathcal{D}_{\text{rel}}$ via LLM-based judgments, we refine the original query q to incorporate their semantic signal. Denoting the original query embedding as v_q, the updated embedding v'_q is computed using one of the strategies described below to enhance downstream retrieval performance.

(1) ReDE-RF Formula. Following [8], the updated query embedding is obtained by averaging the original query embedding with embeddings of the relevant documents, serving as our baseline for comparison with the proposed query refinement methods.

$$v'_q = \frac{1}{k^+ + 1}(v_q + \sum_{i=1}^{k^+} v_{d_i}) \tag{2.1}$$

In this equation, k^+ indicates the total number of documents marked as relevant by the LLM and v_{d_i} represents the embedding of the i-th relevant document.

(2) Contrastive Query Update (CQU). Inspired by the Rocchio algorithm [15], CQU refines the query by moving it toward relevant documents and away from non-relevant ones, using the difference between their average embeddings.

$$v'_q = \alpha \cdot v_q + (1 - \alpha) \cdot \left(\frac{1}{k^+} \sum_{i=1}^{k^+} v_{d_i} - \frac{1}{k^-} \sum_{i=1}^{k^-} v_{d_i} \right) \tag{2.2}$$

The variable k^- indicates the number of non-relevant documents (i.e., documents assigned a label of 0 by the LLM). Finally, α is scalar hyperparameters that determine the contribution of the original query vector and the contrastive feedback signal in the updated embedding.

(3) Weighted Relevance Query Update (WRQU). WRQU computes a weighted average of relevant document embeddings, where each weight w_i corresponds to the LLM-assigned relevance score (1–3), giving more influence to highly relevant documents in query refinement.

$$v'_q = \alpha \cdot v_q + (1 - \alpha) \cdot \left(\frac{\sum_{i=1}^{k^+} w_i \cdot v_{d_i}}{\sum_{i=1}^{k^+} w_i} \right) \tag{2.3}$$

The weight $w_i \in \{1, 2, 3\}$ reflects the relevance score assigned to document d_i by the LLM.

3. Experiments and Results

3.1. Experimental Settings

Dataset. We conduct our evaluations on two datasets: TREC DL19 [9] and TREC DL20 [10]. In addition, we evaluate our methods on four low-resource retrieval datasets from BEIR [6], including TREC Covid, Touche 2020, nfCorpus, and SciFact.

LLMs. We use two open-source models—Mistral-7B-Instruct [16] and Llama3.2 [17]—as LLM judges for document relevance. All models are executed locally using Ollama[1]. Following the setup in [8], we set $k = 20$ to retrieve the top candidate documents, which are subsequently provided to the LLM for relevance judgment.

Baseline. We use ReDE-RF [8] as our primary baseline, which first proposed a zero-shot dense retrieval approach leveraging LLM-guided relevance feedback. ReDE-RF shows that estimating the relevance of initially retrieved documents—without domain-specific training—can substantially improve retrieval. We adopt their original setup to evaluate the effectiveness of our prompting strategies and query reformulation methods, using the unsupervised Contriever [11] for dense retrieval.

3.2. Results

This section presents the findings based on the below research questions:

RQ1. *What is the effect of different LLM prompting strategies for relevance feedback on dense retrieval performance?*

RQ2. *How do different query update strategies based on relevant documents affect dense retrieval performance?*

RQ3. *How does document length affect the effectiveness of different LLM prompting strategies for relevance feedback?*

[1] https://ollama.com

3.2.1. **Effect of Prompting Strategies on Retrieval Performance (RQ1).**

Table 1 presents retrieval performance across different prompting strategies on two datasets (DL19 and DL20) and two LLMs. To isolate the effect of prompting strategies (RQ1), all results are obtained using a fixed query refinement method, namely **ReDE-RF**, which integrates LLM-generated graded relevance feedback into the dense query representation. Values indicate average NDCG@10 scores, with percentage improvement (Δ) over the Zero-Shot baseline from [8]. The * symbol denotes statistical significance on a paired t-test with $p < 0.05$.

Results show that *Few-shot* prompting consistently improves performance relative to Zero-Shot, with statistically significant gains for Mistral across both datasets. *Role-Playing* and *CoT* strategies also generally outperform the baseline, though in two cases small, statistically *in*significant declines occur. Most notably, the *Hybrid* approach delivers the largest and most consistent improvements, reaching up to 20% NDCG@10 gains with strong statistical significance across LLMs and datasets. These findings highlight the importance of carefully selecting and combining prompting strategies to maximize retrieval effectiveness with LLM-based relevance feedback.

Dataset	LLM	Zero-Shot	Few-Shot	Δ	Role-Playing	Δ	CoT	Δ	Hybrid	Δ
DL19	Mistral	49.2	52.1*	+5.81	52.3*	+6.26	49.5	+0.60	54.4*	+10.52
	LLaMA	49.7	50.4	+1.48	49.1	-1.18	54.7*	+10.08	58.0*	+16.78
DL20	Mistral	44.6	48.0*	+7.60	46.1	+3.47	44.5	-0.20	51.4*	+15.19
	LLaMA	44.4	45.4	+2.28	45.6*	+2.65	50.6*	+13.89	53.4*	+20.26

Table 1. NDCG@10 results for different prompting strategies across datasets and LLMs.

3.2.2. **Effect of Query Update Strategies on Retrieval Performance (RQ2).**

This section investigates the impact of different query embedding update methods (see Section 2.2) on retrieval performance. Specifically, we examine how incorporating information from relevant documents—obtained through LLM-generated relevance feedback—affects query representations and, consequently, retrieval effectiveness. We systematically evaluate multiple query embedding update strategies, including CQU (Equation 2.2) and WRQU (Equation 2.3), relative to our baseline formulation, ReDE-RF (Equation 2.1).

Table 2 reports the results for the Mistral model across prompting strategies and datasets, where the hyperparameter α is fixed to 0.5. Overall, both CQU and WRQU consistently outperform ReDE-RF, with particularly strong gains under Zero-Shot, Chain-of-Thought, and Hybrid prompting. For example, WRQU achieves improvements of up to +13.53% on DL19 and +11.99% on DL20, while CQU also demonstrates notable gains under similar settings. Marginal declines observed in certain Few-Shot cases are not statistically significant. Results for LLaMA exhibit similar patterns and are provided in in our public repository.

In both update formulas, the hyperparameter α controls the balance between the original query vector and LLM-judged relevant document information. Its effect on NDCG@10 across DL19 and DL20 for the Mistral model is illustrated in Appendix B. For CQU, performance generally improves as α increases to approximately 0.5, then gradually declines. WRQU performs best at lower α (0.1–0.3), suggesting that it is most effective when the updated embedding relies primarily on relevant document signals.

3.2.3. **Effect of Document Length on Retrieval Performance (RQ3).**

Table 3 reports the NDCG@10 scores for different prompting strategies—Zero-Shot, Few-Shot, Role-Playing, Chain-of-Thought (CoT), and Hybrid—across four BEIR datasets using

Dataset	Method	Zero-Shot	Δ	Few-Shot	Δ	Role-Playing	Δ	CoT	Δ	Hybrid	Δ
DL19	ReDE-RF	49.28	-	52.14	-	52.37	-	49.58	-	54.47	-
	CQU	54.90*	+11.40	51.53	-1.17	54.95	+4.92	55.64*	+12.23	56.73	+4.14
	WRQU	55.95*	+13.53	55.78*	+6.96	57.64*	+9.73	56.21*	+13.37	58.48*	+7.36
DL20	ReDE-RF	44.64	-	48.04	-	46.19	-	44.55	-	51.42	-
	CQU	47.22	+5.77	49.90	+3.87	47.54	+2.92	46.19	+3.67	55.39*	+7.71
	WRQU	50.00*	+11.99	51.10*	+6.37	51.58*	+11.66	50.00*	+12.23	54.81	+6.57

Table 2. NDCG@10 results for Mistral across different prompting methods and datasets using CQU and WRQU query update strategies, with α fixed to 0.5.

Dataset	Method	Zero-Shot	Δ	Few-Shot	Δ	Role-Playing	Δ	CoT	Δ	Hybrid	Δ
Covid	ReDE-RF	26.88	-	30.87	-	30.23	-	30.85	-	32.06	-
	CQU	41.35***	+53.79	40.86**	+32.34	42.86***	+41.73	41.27***	+33.78	42.28***	+31.86
	WRQU	29.89**	+11.15	31.32	+1.45	33.43**	+10.57	33.91**	+9.92	32.06	-0.01
Touche	ReDE-RF	12.25	-	12.94	-	12.39	-	12.41	-	12.91	-
	CQU	17.21*	+40.48	16.98**	+31.24	16.77**	+35.29	17.03**	+37.25	15.71*	+21.62
	WRQU	16.65*	+35.88	16.33**	+26.15	15.80**	+27.50	15.71**	+26.58	16.29**	+26.14
nfCorpus	ReDE-RF	29.04	-	30.72	-	29.3	-	29.06	-	31.14	-
	CQU	35.45***	+22.05	33.12**	+7.82	34.59***	+17.72	34.97***	+20.31	34.85***	+11.92
	WRQU	34.60***	+19.14	33.87***	+10.26	34.98***	+19.04	34.86***	+19.93	35.17***	+12.96
SciFact	ReDE-RF	56.15	-	60.72	-	56.93	-	56.61	-	58.79	-
	CQU	64.07***	+14.10	60.60	-0.19	64.94***	+14.06	64.51***	+13.95	61.77*	+5.05
	WRQU	63.30***	+16.29	64.75***	+6.63	65.61***	+15.23	65.69***	+16.04	64.45***	+9.62

Table 3. NDCG@10 results for different prompting strategies across BEIR datasets using Mistral model ($\alpha = 0.5$). All values are reported as percentages.

the Mistral model, with α fixed to 0.5. The Δ column represents the percentage improvement over the Zero-Shot baseline with ReDE-RF, and statistical significance is indicated as * $p<0.05$, ** $p<0.01$, and *** $p<0.001$. The results demonstrate that both CQU and WRQU query update strategies substantially improve retrieval performance across all datasets. On the Covid dataset, CQU yields large gains, with +53.79% for Zero-Shot, +32.34% for Few-Shot, and +41.73% for Role-Playing prompting, whereas WRQU also provides notable improvements, such as +11.15% for Zero-Shot and +10.57% for Role-Playing. On Touche, CQU consistently outperforms ReDE-RF, achieving up to +40.48% under Zero-Shot and +37.25% with CoT prompting, while WRQU delivers gains between +26.14% and +35.88%. Similar trends are observed for nfCorpus and SciFact, where CQU frequently achieves improvements above +20% with strong statistical significance, and WRQU provides consistent, significant gains as well. Across all datasets, the Hybrid prompting strategy tends to reach the highest NDCG@10 scores, highlighting that combining multiple prompting approaches remains beneficial irrespective of document length. These findings indicate that document length does not constrain the effectiveness of LLM-based relevance feedback, and that integrating relevant document information through CQU or WRQU reliably enhances dense retrieval performance, with statistically significant improvements in the majority of cases.

4. **Conclusion**

In this study, we systematically examined the impact of different prompting strategies and query update methods in LLM-assisted dense retrieval systems. The results demonstrate that the proposed query update methods consistently enhance information retrieval performance. Additionally, we show that alternative prompting techniques, beyond simple Zero-shot prompting, can significantly influence retrieval outcomes. These insights emphasize the importance of combining effective prompting and embedding updates to boost dense retrieval accuracy.

GenAI Usage Disclosure

OpenAI's ChatGPT was used to refine sentence clarity and grammatical correctness during manuscript preparation.

References

[1] V. Karpukhin, B. Oguz, S. Min, P. S. Lewis, L. Wu, S. Edunov, D. Chen, and W.-t. Yih. "Dense Passage Retrieval for Open-Domain Question Answering." In: *EMNLP (1)*. 2020, pp. 6769–6781.

[2] W. X. Zhao, J. Liu, R. Ren, and J.-R. Wen. "Dense text retrieval based on pretrained language models: A survey". In: *ACM Transactions on Information Systems* 42.4 (2024), pp. 1–60.

[3] N. Arabzadeh, F. Zarrinkalam, J. Jovanovic, and E. Bagheri. "Geometric estimation of specificity within embedding spaces". In: *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. 2019, pp. 2109–2112.

[4] S. Robertson, H. Zaragoza, et al. "The probabilistic relevance framework: BM25 and beyond". In: *Foundations and Trends® in Information Retrieval* 3.4 (2009), pp. 333–389.

[5] Z. Peng, X. Wu, Q. Wang, and Y. Fang. "Soft prompt tuning for augmenting dense retrieval with large language models". In: *Knowledge-Based Systems* 309 (2025), p. 112758.

[6] N. Thakur, N. Reimers, A. Rücklé, A. Srivastava, and I. Gurevych. "Beir: A heterogenous benchmark for zero-shot evaluation of information retrieval models". In: *arXiv preprint arXiv:2104.08663* (2021).

[7] L. Gao, X. Ma, J. Lin, and J. Callan. "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)*. 2023, pp. 1762–1777.

[8] N. Jedidi, Y.-S. Chuang, L. Shing, and J. Glass. "Zero-Shot Dense Retrieval with Embeddings from Relevance Feedback". In: *arXiv preprint arXiv:2410.21242* (2024).

[9] N. Craswell, B. Mitra, E. Yilmaz, D. Campos, and E. M. Voorhees. "Overview of the TREC 2019 deep learning track". In: (2020). arXiv: 2003.07820 [cs.IR]. URL: https://arxiv.org/abs/2003.07820.

[10] N. Craswell, B. Mitra, E. Yilmaz, and D. Campos. "Overview of the TREC 2020 deep learning track". In: (2021). arXiv: 2102.07662 [cs.IR]. URL: https://arxiv.org/abs/2102.07662.

[11] G. Izacard, M. Caron, L. Hosseini, S. Riedel, P. Bojanowski, A. Joulin, and E. Grave. "Unsupervised dense information retrieval with contrastive learning". In: *arXiv preprint arXiv:2112.09118* (2021).

[12] T. Kojima, S. S. Gu, M. Reid, Y. Matsuo, and Y. Iwasawa. "Large language models are zero-shot reasoners". In: *Advances in neural information processing systems* 35 (2022), pp. 22199–22213.

[13] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. "Language models are few-shot learners". In: *Advances in neural information processing systems* 33 (2020), pp. 1877–1901.

[14] A Kong, S Zhao, H Chen, Q Li, Y Qin, R Sun, and X Zhou. "Better zero-shot reasoning with role-play prompting. arXiv". In: *URL: http://arxiv. org/abs/2308.07702 [Stand: 9.3. 2024]* (2023).

[15] J. J. Rocchio Jr. "Relevance feedback in information retrieval". In: *The SMART retrieval system: experiments in automatic document processing* (1971).

[16] D. S. Chaplot. "Albert q. jiang, alexandre sablayrolles, arthur mensch, chris bamford, devendra singh chaplot, diego de las casas, florian bressand, gianna lengyel, guillaume lample, lucile saulnier, lélio renard lavaud, marie-anne lachaux, pierre stock, teven le scao, thibaut lavril, thomas wang, timothée lacroix, william el sayed". In: *arXiv preprint arXiv:2310.06825* 3 (2023).

[17] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, et al. "Llama: Open and efficient foundation language models". In: *arXiv preprint arXiv:2302.13971* (2023).

Appendix A. Prompt Templates for LLM-based Relevance Estimation

This appendix provides the exact prompt templates used for all prompting strategies evaluated in this work. Prompts are shown verbatim to ensure full reproducibility.

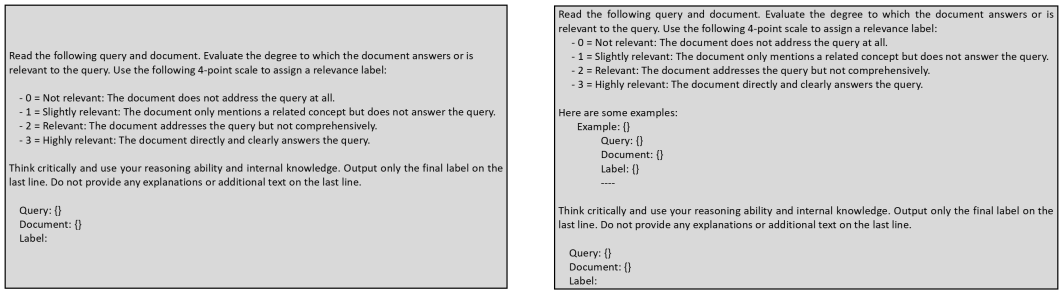


Figure 1. Prompt templates for relevance estimation: (left) Zero-shot prompt; (right) Few-shot prompt.

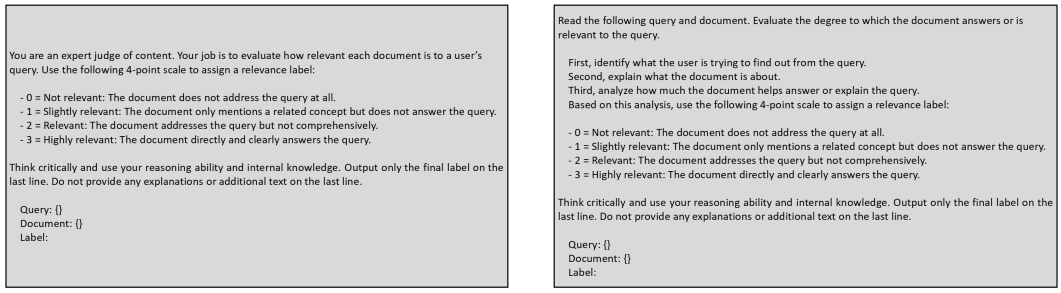


Figure 2. Prompt templates for relevance estimation: (left) Role-Playing prompt; (right) Chain-of-Thought prompt.

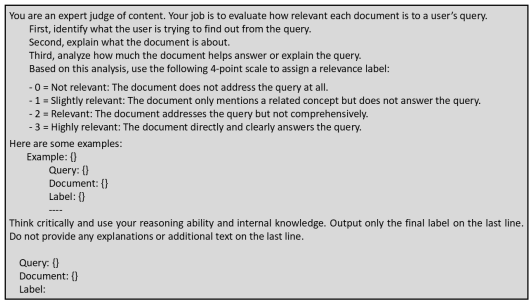
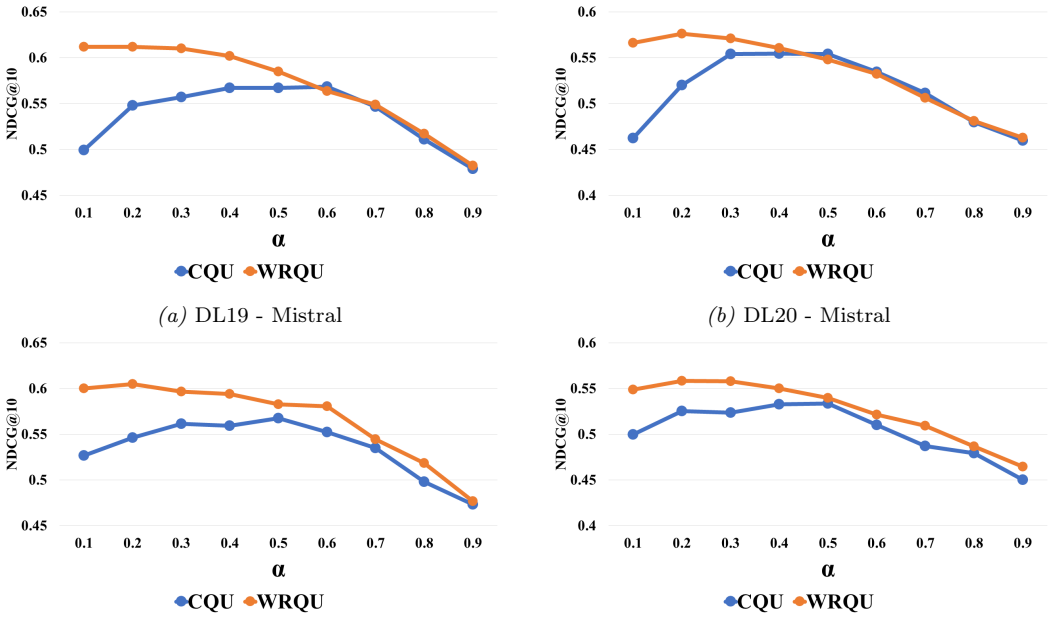


Figure 3. Hybrid prompt combining role-playing, few-shot examples, and chain-of-thought reasoning for fine-grained relevance estimation.

Appendix B. Effect of the Update Coefficient α

This appendix analyzes the sensitivity of retrieval performance to the update coefficient α used in the query embedding update formulas.



(a) DL19 - Mistral

(b) DL20 - Mistral

(c) DL19 - LLaMA

(d) DL20 - LLaMA

Figure 4. Effect of the hyperparameter α in CQU and WRQU query update formulas on retrieval performance across DL19 and DL20 datasets using Mistral and LLaMA.