Adaptive Retrieval with Cross-Session Feedback for Enterprise Search

Anonymous ACL submission

Abstract

Enterprise document retrieval differs fundamentally from open-domain search due to domain-specific terminology, expert-defined relevance, and limited labeled data. While recent neural retrieval models achieve strong performance on standard benchmarks, they are typically trained offline and remain static after deployment, causing discrepancy especially for the enterprise domain. To alleviate the issue, we propose a retrieval model that improves search performance through explicit relevance feedback in an online setting. Unlike prior work in information retrieval, in which user feedback is primarily leveraged within a single session, we introduce a new problem setting in which feedback is persistently stored and exploited across multiple sessions. In this scenario, similar queries are repeatedly issued and the system adapts over time by accumulating queries and their feedback. Our approach combines base retrieval scores with feedback-informed scores and dynamically adjusts their contribution without assuming an optimal weighting a priori. Experiments show that accumulating feedback consistently improves retrieval accuracy, even under query reformulation. These results illustrate the effectiveness of feedback-driven online adaptation as a lightweight alternative to fine-tuning for specialized retrieval tasks.

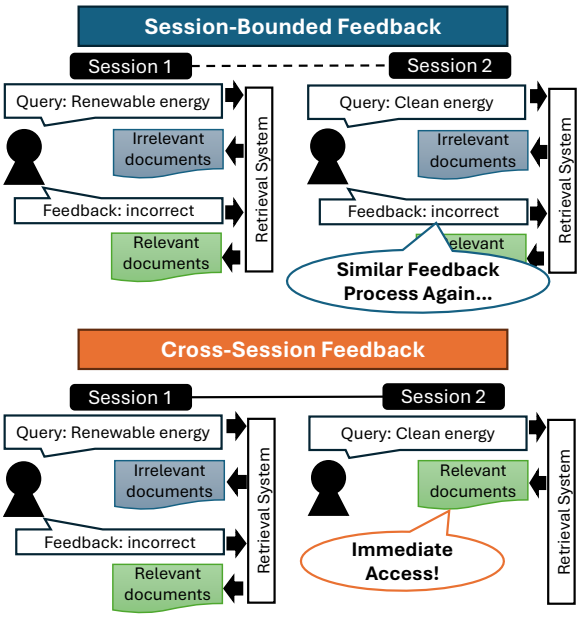

Figure 1: Top: Classical relevance feedback improves retrieval results only within a single search session, requiring users to provide similar feedback again when similar information needs arise in subsequent sessions. Bottom: Our setting retains feedback across sessions, enabling improved retrieval for recurring information needs without repeated user annotations.

1 Introduction

Information retrieval (IR) in enterprise environments presents unique challenges that differ fundamentally from open-domain search. Internal document collections are often heterogeneous and sparsely labeled, while relevance is defined by domain-specific criteria and expert judgment. In addition, such corpora frequently contain organization-specific terminology and abbreviations that are poorly represented in general-purpose pretrained models.

Although recent neural search models have achieved strong performance on standard benchmarks, most existing approaches treat retrieval as a static problem optimized offline (Mitra and Craswell, 2017; Karpukhin et al., 2020; Guu et al., 2020; Qu et al., 2021; Izacard et al., 2022). This assumption is particularly limiting in enterprise settings, where relevance notions are highly context-dependent and must be refined through continual user interaction usually performed within a small community. Moreover, similar information needs frequently arise across different users in such environments. In many real-world deployments, users explicitly indicate whether retrieved documents are relevant to their information needs, for example by confirming correct results or reporting mismatches. Such feedback reflects expert knowledge and task-

specific relevance, yet is typically incorporated only for ad-hoc retrieval (Shen and Zhai, 2005; Mao et al., 2024).

Figure 1 illustrates the problem setting addressed in this study. The session-bounded design limits the applicability of classical relevance feedback in enterprise search environments, where similar or recurring queries are issued over time and relevance criteria are improved by accumulated domain expertise. In such settings, treating each search interaction independently fails to exploit valuable historical feedback that could inform subsequent searches. In contrast, this paper addresses a different problem setting: we focus on persistent, cross-query utilization of explicit relevance feedback, enabling the retrieval system to adapt continuously across sessions. Rather than reformulating individual queries, we store feedback as a long-term signal and incorporate it as an auxiliary scoring component. This allows the system to incrementally adapt to user feedback while maintaining stable retrieval performance, a setting that has received limited attention in prior work. Notably, the queries and feedback need not originate from the same user, as relevance judgments can be accumulated from multiple users within an organization who exhibit similar information needs.

Under this setting, we propose a new retrieval model that leverages a history of user queries and feedback on the retrieval results for continual improvement. The feedback database consists of past search queries and (averaged) binary feedback for the documents shown to the user by those queries. This database is able to assist a conventional retrieval by adding feedback-based score with a specific weight. In addition, we propose to dynamically optimize the contribution of each score based on the given feedback.

We apply our method for text-to-text retrieval under question-answering based on documents. In order to demonstrate the effectiveness of our approach in an online fashion, we leverage synthetic questions generated from the original question-answering data using large language models. To account for more realistic settings, we also evaluate the proposed method under settings where feedback is frequently unavailable or occasionally incorrect. The results show that the proposed model consistently improves retrieval performance, including on previously unseen queries, and exhibits robustness to moderate levels of sparse or noisy feedback. The simplicity and fine-tuning–free nature of the method make it easy to deploy in practical settings where continuous adaptation and low operational overhead are required.

2 Related Work

Retrieval models The retrieval model mainly comprises a sparse model, such as TF-IDF and Best Matching 25 (BM25; Robertson and Zaragoza (2009)), and dense retrievals. BM25 is the most widely used sparse retrieval algorithm which excels at precise keyword matching, and it has been shown to perform competitively in specialized domains where document collections with organization-specific terminology.

For the dense retrieval, early work focused on representation learning with dual-encoder architectures, enabling efficient similarity-based retrieval in a shared embedding space (Huang et al., 2013). More recently, a deep passage retriever succeeded in achieving higher retrieval accuracy than that of BM25 in multiple datasets by training encoders (Karpukhin et al., 2020), and fine-tuning of the dense retrieval model is widely studied particularly for open domain question answering (Guu et al., 2020; Qu et al., 2021; Izacard et al., 2022). Generation-augmented retrieval is also one of the most famous methods to improve the retrieval accuracy with less training cost (Mao et al., 2021). Despite their success, neural retrieval models often struggle in enterprise settings characterized by domain-specific terminology and limited annotated data (Bruckhaus, 2024). Moreover, most models are trained offline and remain static after deployment, limiting their ability to adapt to evolving vocabularies and relevance criteria.

Query and document expansion using Large Language Models (LLMs) can improve retrieval especially for under-specified queries (Li et al., 2022; Gao et al., 2023; Ma et al., 2023; Raina and Gales, 2024; Wang et al., 2025; Hayashi et al., 2025), but general-purpose LLMs are unreliable when applied to enterprise corpora because internal documents contain organization-specific terminology not present in public training data.

Feedback for retrieval models Interaction between human and machine learning models is actively studied in various fields such as reinforcement learning from human feedback (Christiano et al., 2017; Bai et al., 2022), alignment of generative models (Ouyang et al., 2022; Li et al., 2024), and development of datasets and benchmarks (Kirk

et al., 2023, 2024; Khirodkar et al., 2024). Some studies utilize human feedback for the retrieval method; however, they mostly focus on personal preferences specifically designed for recommendation tasks (Wang and Wu, 2023; Wu et al., 2024; Poddar et al., 2024). Similar studies such as image retrieval with human feedback have been widely conducted in various areas (Cox et al., 2000; Zhou, 2003; Ferecatu and Geman, 2007; Kovashka et al., 2012; Ak et al., 2018). However, they heavily rely on personal preferences, and are not suitable for the goal of objectively accurate information intended for professional/corporate use cases.

Various types of feedback exist such as binary (Poddar et al., 2024), relevance (Zhou, 2003), and natural language (Yuan and Lam, 2021). More recent work explores learning from user interactions at scale, including click logs and other implicit signals (Huang et al., 2012). While effective in web-scale settings, such signals are often noisy and ambiguous, making them less suitable for enterprise search, where relevance criteria are domain-specific and correctness is critical. In contrast, explicit relevance feedback provided by users offers high-quality supervision that directly reflects task-specific notions of relevance. However, existing approaches typically incorporate such feedback through session-level updates (*e.g.,* Mao et al. (2024)).

Classical relevance feedback methods, such as Rocchio-style relevance feedback and pseudo relevance feedback (PRF), have long been studied as effective techniques for improving retrieval quality (Salton and Buckley, 1990; Li et al., 2023). These approaches refine retrieval results by reformulating a query based on documents judged (or assumed) as relevant within a single search session. However, such methods are inherently session-bound: the feedback is used to update the query representation only for the current interaction and is discarded once the session ends. Consequently, feedback obtained from past queries is not retained as persistent knowledge, nor is it reused to benefit future retrievals.

3 Method

Figure 2 shows the overview of the proposed method, mainly comprising three phases: previous-query retrieval, document retrieval, and feedback. Given a query q, the top-k most similar past queries $\mathcal{Q}^{\text{top-k}}$ are extracted in the previous-query retrieval

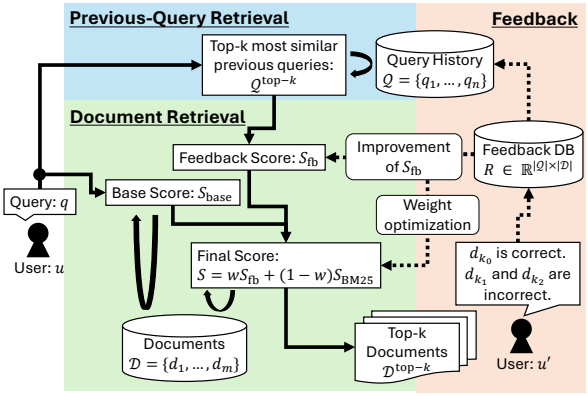

Figure 2: Overview of the proposed model. Our model can continuously improve search accuracy by utilizing past retrieval queries and the feedback results to score calculation.

phase. In document retrieval, feedback score S_{fb} is calculated using $\mathcal{Q}^{\text{top-k}}$ and their associated feedback result on each document. Then, the base relevant score S_{base} of each document for a given query q are calculated using a typical score such as that of BM25 and cosine similarity. The final score S is the linear interpolation of the above scores, i.e., S_{fb} and S_{base}, with weight w.

In the feedback phase, top-k documents retrieved based on S are presented to a user, e.g., $d_{k_0}, d_{k_1}, d_{k_2}$ in the figure. Then, the user can give binary (correct or incorrect) feedback on each document, which is used to improve the retrieval accuracy by two approaches. The results are stored in the feedback database and reflected in the score S_{fb} in future retrievals. The weight w is continuously optimized based on whether the retrieval is successful or not.

Intuitively, the score S_{fb} strengthens documents that users have previously marked as correct with a similar query. This complements conventional retrieval with human knowledge, allows the system to benefit from past interactions, and helps capture proprietary terms and abbreviations that standard lexical and dense models often overlook.

Feedback database Let $\mathcal{R} \in \mathbb{R}^{|\mathcal{D}|\times|\mathcal{Q}|}$ denote a sparse feedback reward matrix, where columns correspond to document indices i and rows correspond to query indices j given a document database $\mathcal{D}$ and a history of queries $\mathcal{Q}$. Each entry r_{ij} represents the reward value associated with the document-query pair (d_i, q_j), where $-1 \leq r_{ij} \leq 1$. The database is updated when a new user feedback is available.

Previous-Query Retrieval We first extract top-k most similar previous queries from the feedback record database. We use the score based on the cosine similarity between the dense embedded vectors of the given query q and the previous queries $q_j \in \mathcal{Q}$,

$$\mathcal{J}^{\text{top-}k} = \arg\operatorname*{top-k}_{j} \frac{E(q) \cdot E(q_j)}{\|E(q)\| \|E(q_j)\|}, \quad (1)$$

where $\mathcal{J}^{\text{top-k}}$ and $E(\cdot)$ indicate the set of indices of top-k most similar previous queries and encoder module, respectively.

Document Retrieval The score based on feedback S_{fb} for document i is calculated as an average of feedback reward r_{ij} for $j \in \mathcal{J}^{\text{top-k}}$ by following equation:

$$S_{\text{fb}}(i) = \frac{1}{k} \sum_{j \in \mathcal{J}^{\text{top-k}}} r_{ij} \frac{E(q) \cdot E(q_j)}{\|E(q)\| \|E(q_j)\|}. \quad (2)$$

To extract the top-k documents, the final score S for document d_i is calculated by linearly interpolating S_{fb} with a conventional retrieval score:

$$S(i) = wS_{\text{fb}}(i) + (1 - w)S_{\text{base}}(i), \quad (3)$$

where, S_{base} indicates a typical retrieval metrics such as BM25 score with sparse retrieval or cosine similarity with dense retrieval, which measures the relevance of q against d_i. S_{base} is normalized per query using its minimum and maximum values to match the scale of the feedback score. The weighted sum allows feedback to augment rather than replace conventional retrieval models, enabling the system to capture domain-specific knowledge while retaining the stability of established lexical or dense retrieval.

Feedback for Continuous Improvement After the retrieval, a user will provide positive feedback, i.e., +1, for correct documents and negative feedback, i.e., -1, for incorrect ones as shown in Figure 2. The feedback is used to improve future retrieval accuracy in two ways: (1) improving S_{fb} by expanding feedback record $\mathcal{R}$ and (2) continuous optimization of fusion weight w.

First, feedback database is continuously updated using the feedback results. Let $\mathcal{D}^{\text{top-k}} \subseteq \mathcal{D}$ denote the set of retrieved documents by Equation 3 and let d_q^* be the ground-truth document for the given query q identified by the user. If q is not in the query history $\mathcal{Q}$, we define the sparse row $r \in \{-1, 0, 1\}^{|\mathcal{D}|}$ by

$$r_i = \begin{cases} +1, & \text{if } d_i = d_q^* \text{ and } d_i \in \mathcal{D}^{\text{top-k}}, \\ -1, & \text{if } d_i \neq d_q^* \text{ and } d_i \in \mathcal{D}^{\text{top-k}}, \\ 0, & \text{otherwise.} \end{cases}$$

Then, we append r as a new row to matrix $\mathcal{R}$.

Second, feedback is used to estimate the optimal weight w dynamically. Importantly, the optimal weight is not known a priori and depends on corpus characteristics, query types, and the amount of available feedback. Dynamic weight allows the model to benefit from the stability of the base model in early stages and increasingly rely on the feedback signal as it becomes informative, effectively interpolating between conventional retrieval and user feedback.

To address this uncertainty, we model weight selection as an online decision-making problem and apply Thompson Sampling (Thompson, 1933) to dynamically optimize the weighting parameter based on observed relevance feedback as described in Algorithm 1. Initially, we discretize the interval into N uniformly spaced candidates $\mathcal{W}$ that are regarded as arms in line 1. In our experiments, the reward is binary, i.e., successful if $d_q^* \in \mathcal{D}^{\text{top-k}}$, such that we assume that the reward follows the Bernoulli distribution. Then, the success probability $(\theta_1, ..., \theta_N)$ of each arm n is estimated by

Algorithm 1 Thompson sampling for weight w

1: **Input:** number of arms $N \geq 2$, discretized weight $\mathcal{W} = (w_1, ..., w_N)$; $\forall w \in [0, 1]$, set of retrieved top-k documents $D^{\text{top-k}}$, correct document d_q^*
2: **Initialize:** $S_n = 0$, $F_n = 0$; $\forall n \in \{1, ..., N\}$
3: **for** $t = 1$ **to** T **do**
4: **for** $n = 1$ **to** N **do**
5: Sample $\theta_n(t) \sim \text{Beta}(S_n, F_n)$
6: **end for**
7: Select $i^* = \arg\max_i \theta_i(t)$
8: Play retrieval task with weight w_{n^*}
9: Observe

$$r_{n^*} = \begin{cases} 1, & \text{if } d_q^* \in D^{\text{top-k}} \\ -1, & \text{otherwise} \end{cases}$$

10: Update $S_{n^*} \leftarrow S_{n^*} + r_{n^*}$, $F_{n^*} \leftarrow F_{n^*} + (1 - r_{n^*})$
11: **end for**

the Beta distribution in lines 4-6. The arm with the highest value is selected by sampling from the distribution. This allows the system to balance exploration and exploitation without assuming a static optimal weight in advance. We note that this interaction-level reward for weight optimization differs from the document-level rewards used to update the feedback score S_{fb}, where each retrieved document receives its own relevance label.

4 Experiments

We validated the hit rate of the model with relevance feedback simulation by using a domain-specific dataset comprising pairs of a question and supporting contexts. Unlike standard supervised fine-tuning, our setting does not assume a clear separation between training and evaluation queries. Instead, we consider a retrieval scenario in which similar queries are repeatedly given, and the system improves over time based on provided relevance feedback. Our goal is not to optimize generalization to unseen queries, but to enable rapid adaptation to recurring information needs and domain-specific usage patterns in an active learning setting.

4.1 Synthetic Question Generation

Our model is designed to improve the retrieval accuracy when the feedback database contains similar questions and the feedback results. However, it demands intensive human labors in the human-in-the-loop setting, and not realistic to conduct experiments using actual human-made questions. Therefore, we propose to generate synthetic questions from original questions in datasets using LLMs. To preserve linguistic diversity among generated queries, we carefully designed the prompt to allow varied phrasing. Through this process, we obtain N_{sq} additional synthetic questions for each question, assuming that the generated questions should retrieve the same documents as the original questions, i.e., $d^*_{q'} = d^*_q$ for the synthetic question q' generated from q.

To ensure the consistency of query intent, we employ a filtering mechanism on a set of synthetic queries generated in large quantities. Specifically, we prompt the LLM to answer both the original query and its synthetic variant using the same supporting contexts, and measure the agreement between the two outputs using the F1 score. Only those synthetic queries whose answers achieve sufficiently high scores are retained for subsequent verifications. The prompts for query generation and filtering are described in Appendix A.

4.2 Datasets

We conduct our experiments on publicly available BEIR datasets (Thakur et al., 2021), a heterogeneous benchmark for diverse IR tasks, to ensure reproducibility and fair comparison. We used test splits of BEIR/NQ (Natural Questions; Kwiatkowski et al. (2019)) and BEIR/FiQA (Maia et al., 2018) in this study. NQ is a large-scale question answering dataset consisting of real user queries issued to the Google Search engine, which is used to evaluate our method under general domain. Each question is paired with a Wikipedia page and annotated with long and short answers derived from that page. As a benchmark to simulate enterprise retrieval, we employ FiQA, a finance-domain opinion-based retrieval benchmark, where queries are answered by retrieving relevant explanatory documents rather than extracting a single factual answer.

4.3 Setups

In this study, we conducted experiments with $k = 1$ and 3 for top-k retrieval. As described in Section 3, the weight w is optimized by Thompson sampling denoted as *dynamic weight*. In order to quantify the gains of the weight optimization, we also conducted the experiments with a constant weight $w = 0.2$ and $w = 0.8$ denoted as *static weight*. We evaluate retrieval performance using *Hit Rate@k*, where a query is considered successful if at least one relevant document appears in the top-k retrieved results.

BM25 and cosine similarity measured by *e5-small-v2* query and context embeddings is used for S_{base}. *e5-small-v2* is also used to calculate similarities between a given query q and previous queries $\mathcal{Q}$ in Equation 1. Number of arms in Thompson sampling N is set to 6: $\mathcal{W} = \{0.0, 0.2, \ldots, 1.0\}$. To reduce redundant similarity computations in repeated experiments, we precompute the top-100 retrieved candidates per query using BM25s (Lù, 2024) and FAISS (Douze et al., 2025), respectively. S_{base} is normalized between 0 and 1 for each query because the changes in score scales make it difficult to find an optimal w.

We used GPT-4o-mini (Hurst et al., 2024) to generate and filter synthetic queries. Setting $N_{sq} = 10$, we generate synthetic questions using nucleus sampling with a temperature of 0.7 and top-p of 0.9

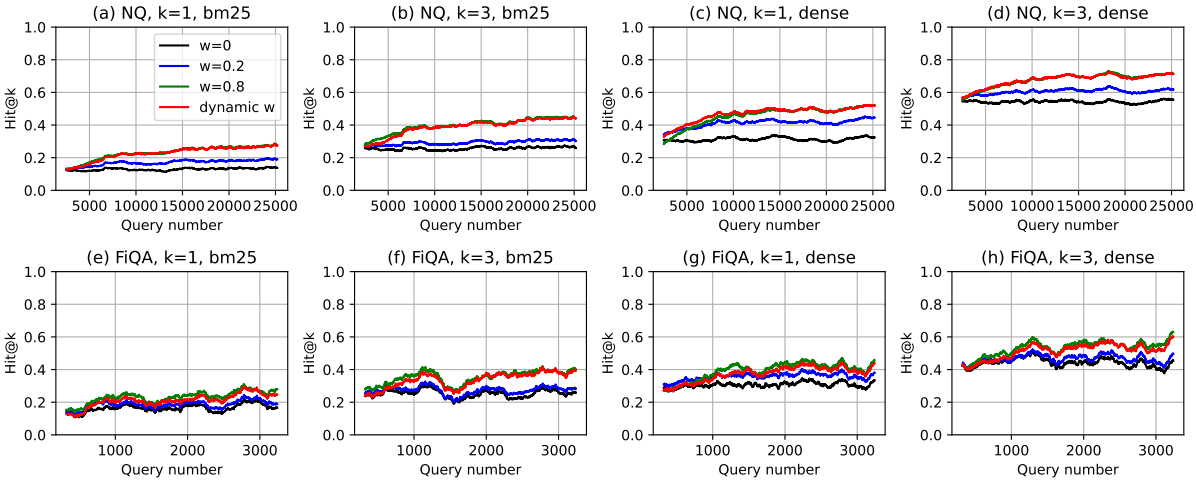

Figure 3: Moving average of hit rates of proposed model with dynamic and static weights ($w = 0.2$ and $w = 0.8$) and baselines along query input using synthetic questions.

to encourage diversity. In contrast, for the filtering step, we use greedy decoding (temperature = 0) to obtain deterministic and reproducible answers. Synthetic queries are filtered by requiring that the F1 score between the LLM-generated answers for the original and synthetic queries is at least 0.7. As a result, the experiments were conducted using 25,187 and 3,241 synthetic queries for NQ and FiQA, respectively. Appendix B shows examples of synthetic questions of the two datasets.

In real-world enterprise search scenarios, users are often reluctant to provide explicit relevance feedback for every retrieval result (Manning, 2008). To reflect this practical constraint, we simulate sparse feedback conditions by randomly dropping feedback signals. Specifically, we conduct experiments where explicit feedback is unavailable for 50% and 90% of queries, respectively.

4.4 Results

We report results with and without feedback to isolate its effect on retrieval performance.

Figure 3 shows the hit rates of proposed model with static ($w = 0.2$ and $w = 0.8$) and dynamic weights and no weighting baselines by setting $w = 0$ using base score alone. The hit rate is reported as a moving average computed over every 10% of total queries in each dataset, and the same averaging procedure is applied to all subsequent figures. The order of the synthetic questions is randomly shuffled considering practical applications. These results are obtained under an idealized assumption that all feedback is provided correctly. Across both NQ and FiQA datasets, sparse and dense retrieval methods, and for different values of k, the proposed method continues to outperform the baseline without feedback ($w = 0$). We additionally conducted experiments with multiple random seeds to assess robustness. As shown in Appendix C, the proposed method consistently outperforms the baseline across all settings.

Figure 4 shows hit rates with full and sparse feedback along each query input. As expected, retrieval performance degrades as feedback becomes sparser; however, once sufficient feedback has been accumulated, the gap between the proposed method and the baseline remains substantial. These results indicate that our approach is robust to highly sparse feedback and can effectively accumulate and exploit limited user feedback over time.

5 Analysis

Robustness to noisy feedback To evaluate the robustness of the proposed method to imperfect human feedback, we conducted additional experiments of injecting noises into the feedback signals. Specifically, during feedback simulation, the reward for each retrieved document was flipped with a static probability, such that a positive reward ($+1$) became negative (-1) and vice versa. We have experimented two noise levels, 0.1 and 0.3, corresponding to cases where 10% and 30% of feedback signals were corrupted, respectively.

The results in Figure 5 show that the proposed method remains relatively robust under moderate

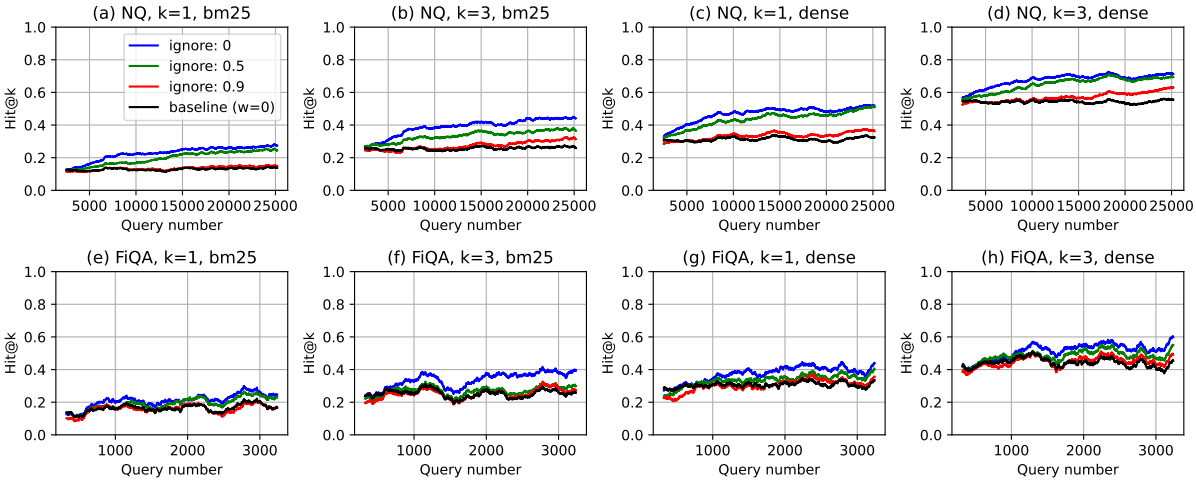

Figure 4: Hit rates of proposed model along query inputs using LLM-reformulated questions and dynamic weights under conditions where feedback is not provided with a certain probability (50% and 90%).

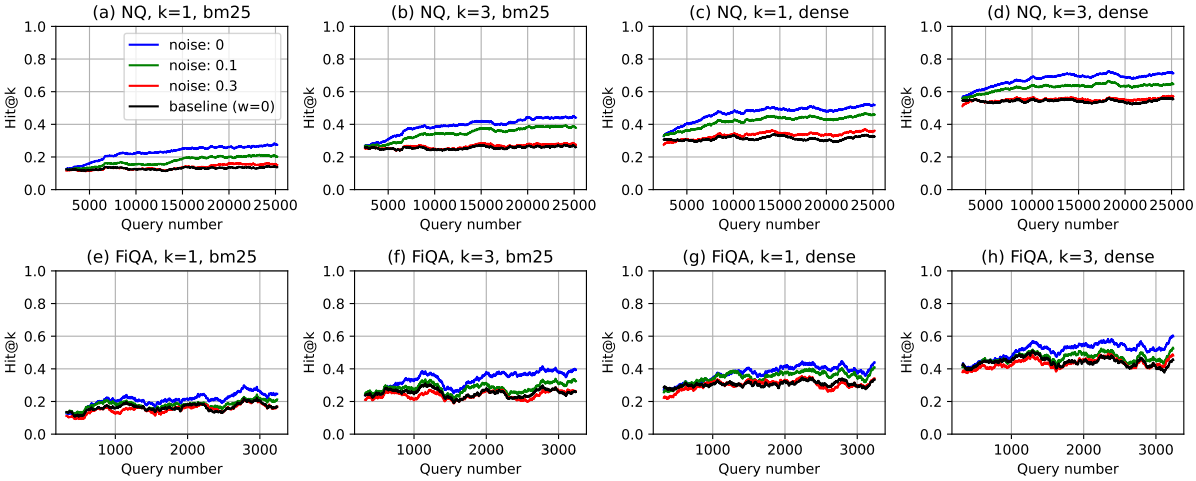

Figure 5: Hit rates of proposed model with a certain noise level (10% and 30%) along query inputs using LLM-reformulated questions and dynamic weights.

noise. When the noise level was set to 0.1, retrieval accuracy decreased slightly compared to the noiseless setting; however, the system still exhibited clear performance gains over the no-feedback baseline, indicating sufficient adaptability to moderately noisy feedback. In contrast, at a higher noise level of 0.3, retrieval accuracy degraded more substantially, and the performance became comparable to or worse than that of the baseline model without feedback. This suggests that while the proposed approach can tolerate a limited amount of feedback noise, excessively noisy signals undermine the benefit of feedback-driven adaptation.

Sensitivity to weight We further analyze how the selected weight changes throughout the feedback process. As shown in Figure 3, when feedback accumulation begins, smaller weights yield higher accuracy, since the feedback score is based on small amount of interactions and thus remains unreliable. As more feedback is collected, larger weights gradually become more effective, reflecting the improved quality of the feedback scores. This trend indicates that no single static weight is optimal across feedback rounds and highlights the importance of dynamically adjusting the contribution of the feedback score.

Figure 6 indicates the plots of the selected

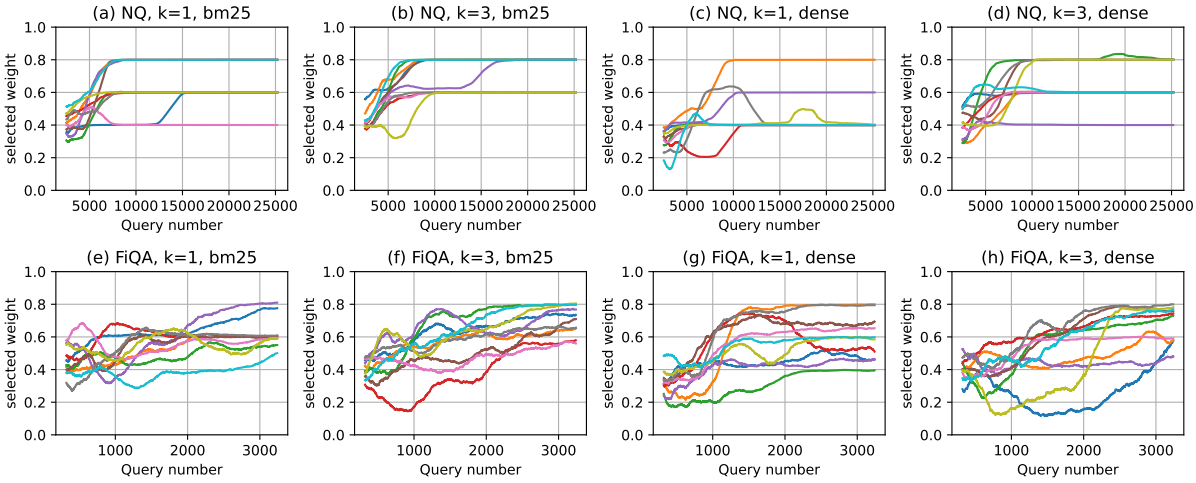

Figure 6: Plots of the selected weight w through feedback process. Different line colors correspond to different random seeds.

weights throughout the feedback process in the idealized setting with full and correct feedback. We visualize multiple runs with different random seeds, where each line color corresponds to a distinct seed. According to the result, in many cases, the selected weights tend to drop from a non-converged initial state in some feedback counts and then rise as feedback accumulates more. For instance, the orange line in panel (g), it starts at $w \simeq 0.3$, drops to $w \simeq 0.2$ in around 500 queries, and finally rises to $w \simeq 0.8$. This trend indicates that the original motivation behind dynamic weighting has been achieved: the model benefits from the stability of the base model in the early stages, and as the feedback signal becomes more informative, it increasingly relies on that signal.

Under retrospective analysis, a static weight such as 0.8 can achieve better performance in several cases (*e.g.*, Panel (e) in Figure 3). However, identifying such an optimal weight is only possible after observing the results. In realistic deployment scenarios, the optimal weight is unknown in advance and may vary depending on the amount and quality of feedback, corpus and baseline. Our dynamic weighting approach is designed to address this practical constraint by adaptively adjusting the weight, rather than relying on a fixed parameter selected post hoc.

Sensitivity to the number of retrieval We observe that top-3 retrieval leads to faster improvement in hit rate than top-1 retrieval. This is because the larger the value of k, the more information provided by each feedback, and the faster the accuracy of the model can be improved.

In contrast, in practice, the maximum value of k would be constrained by the cost and availability of manual human feedback. Therefore, we only conducted the cases of $k = 1, 3$ while some studies evaluate the performance with much higher k (*e.g.*, top-20 in Karpukhin et al. (2020)).

6 Conclusion

We introduce a new problem setting where feedback is accumulated and exploited across sessions. In addition, we proposed a retrieval model that improves performance through explicit relevance feedback in an online setting. Unlike conventional problem setting, our evaluation focuses on retrieval scenarios where similar queries are repeatedly issued to a system. Experiments on the dataset demonstrate that accumulating feedback consistently improves retrieval accuracy, even under query reformulation by LLMs. Additional analyses further show that the proposed method remains robust to moderate levels of feedback noise, while its effectiveness naturally degrades when feedback becomes highly unreliable or sparse. We also observe that the optimal weighting between the base model and the feedback score evolves over time: smaller weights are preferable early on, while larger weights become effective as more feedback accumulates. These results highlight the effectiveness of online feedback-driven adaptation for specialized retrieval tasks.

Limitations

Corpora A key limitation of this study is the lack of evaluation on proprietary enterprise corpora. Since the internal corpus used in our real-world system cannot be publicly released, we conduct our experiments only on publicly available datasets. Unlike open-domain datasets, internal document collections typically exhibit highly domain-specific terminology and implicit contextual knowledge, which can significantly degrade the effectiveness of dense retrieval methods trained on general corpora. However, these datasets are inherently non-public due to confidentiality and privacy constraints, preventing standardized experimental validation.

Baselines Our experiments use BM25 and a standard dense retriever based on a bi-encoder architecture as baselines. While these approaches are widely adopted and provide strong and reliable points of comparison, they represent relatively classical retrieval methods. More advanced combinations—such as hybrid dense–sparse retrieval or cross-encoder re-ranking—are not explored in this work and may further improve performance.

Synthetic queries Creating thousands of diverse and well-controlled queries through manual annotation is not practical at scale. Our use of synthetic query variations is motivated by this constraint and is intended to systematically simulate realistic linguistic variations, such as paraphrasing and lexical substitutions, that naturally occur in user queries. The goal of this evaluation is not to replicate any specific real-world query distribution, but to assess whether the model can robustly leverage feedback under controlled query perturbations. We believe this setup provides a reasonable and scalable alternative for studying feedback-driven retrieval models. We consider a human-in-the-loop study an important direction for future work.

Credit assignment for dynamic weight In weight optimization, the current top-k feedback formulation can introduce ambiguity in credit assignment when $k > 1$. Incorporating rank-sensitive feedback is a promising direction.

Acknowledgments

References

Kenan E. Ak, Ashraf A. Kassim, Joo Hwee Lim, and Jo Yew Tham. 2018. Learning attribute representations with localization for flexible fashion search. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7708–7717.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova Dassarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, John Kernion, Tom Conerly, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, and 12 others. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *ArXiv*, abs/2204.05862.

Tilmann Bruckhaus. 2024. Rag does not work for enterprises. *arXiv preprint arXiv:2406.04369*.

Paul F. Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 4302–4310, Red Hook, NY, USA. Curran Associates Inc.

I.J. Cox, M.L. Miller, T.P. Minka, T.V. Papathomas, and P.N. Yianilos. 2000. The bayesian image retrieval system, pichunter: theory, implementation, and psychophysical experiments. *IEEE Transactions on Image Processing*, 9(1):20–37.

Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. 2025. The faiss library. *IEEE Transactions on Big Data*.

Marin Ferecatu and Donald Geman. 2007. Interactive search for image categories by mental matching. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8.

Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. 2023. 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)*, pages 1762–1777, Toronto, Canada. Association for Computational Linguistics.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Realm: retrieval-augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org.

Kazuki Hayashi, Hidetaka Kamigaito, Shinya Kouda, and Taro Watanabe. 2025. Iterkey: Iterative keyword generation with llms for enhanced retrieval augmented generation. *arXiv preprint arXiv:2505.08450*.

Jeff Huang, Ryen W White, Georg Buscher, and Kuansan Wang. 2012. Improving searcher models using mouse cursor activity. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, pages 195–204.

Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. 2013. Learning deep structured semantic models for web search using clickthrough data. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pages 2333–2338.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, and 1 others. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2022. Unsupervised dense information retrieval with contrastive learning. *Transactions on Machine Learning Research*.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Rawal Khirodkar, Jyun-Ting Song, Jinkun Cao, Zhengyi Luo, and Kris Kitani. 2024. Harmony4d: A video dataset for in-the-wild close human interactions.

Hannah Rose Kirk, Andrew M. Bean, Bertie Vidgen, Paul Röttger, and Scott A. Hale. 2023. The past, present and better future of feedback learning in large language models for subjective human preferences and values. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2409–2430, Singapore. Association for Computational Linguistics.

Hannah Rose Kirk, Alexander Whitefield, Paul Röttger, Andrew Bean, Katerina Margatina, Juan Ciro, Rafael Mosquera, Max Bartolo, Adina Williams, He He, Bertie Vidgen, and Scott A. Hale. 2024. The prism alignment dataset: What participatory, representative and individualised human feedback reveals about the subjective and multicultural alignment of large language models. *Preprint*, arXiv:2404.16019.

Adriana Kovashka, Devi Parikh, and Kristen Grauman. 2012. Whittlesearch: Image search with relative attribute feedback. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2973–2980.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, and 1 others. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

Hang Li, Ahmed Mourad, Shengyao Zhuang, Bevan Koopman, and Guido Zuccon. 2023. Pseudo relevance feedback with deep language models and dense retrievers: Successes and pitfalls. *ACM Transactions on Information Systems*, 41(3):1–40.

Hang Li, Shengyao Zhuang, Ahmed Mourad, Xueguang Ma, Jimmy Lin, and Guido Zuccon. 2022. Improving query representations for dense retrieval with pseudo relevance feedback: A reproducibility study. In *Advances in Information Retrieval*, pages 599–612, Cham. Springer International Publishing.

Shufan Li, Konstantinos Kallidromitis, Akash Gokul, Yusuke Kato, and Kazuki Kozuka. 2024. Aligning diffusion models by optimizing human utility. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Xing Han Lù. 2024. Bm25s: Orders of magnitude faster lexical search via eager sparse scoring. *Preprint*, arXiv:2407.03618.

Xinbei Ma, Yeyun Gong, Pengcheng He, Hai Zhao, and Nan Duan. 2023. Query rewriting in retrieval-augmented large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5303–5315.

Macedo Maia, Siegfried Handschuh, André Freitas, Brian Davis, Ross McDermott, Manel Zarrouk, and Alexandra Balahur. 2018. Www'18 open challenge: financial opinion mining and question answering. In *Companion proceedings of the the web conference 2018*, pages 1941–1942.

Christopher D Manning. 2008. *Introduction to information retrieval*. Syngress Publishing,.

Xinyu Mao, Shengyao Zhuang, Bevan Koopman, and Guido Zuccon. 2024. Dense retrieval with continuous explicit feedback for systematic review screening prioritisation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '24, page 2357–2362, New York, NY, USA. Association for Computing Machinery.

Yuning Mao, Pengcheng He, Xiaodong Liu, Yelong Shen, Jianfeng Gao, Jiawei Han, and Weizhu Chen. 2021. Generation-augmented retrieval for open-domain question answering. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4089–4100, Online. Association for Computational Linguistics.

Bhaskar Mitra and Nick Craswell. 2017. Neural models for information retrieval. *arXiv preprint arXiv:1705.01509*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with

human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744. Curran Associates, Inc.

Sriyash Poddar, Yanming Wan, Hamish Ivison, Abhishek Gupta, and Natasha Jaques. 2024. Personalizing reinforcement learning from human feedback with variational preference learning. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Yingqi Qu, Yuchen Ding, Jing Liu, Kai Liu, Ruiyang Ren, Wayne Xin Zhao, Daxiang Dong, Hua Wu, and Haifeng Wang. 2021. RocketQA: An optimized training approach to dense passage retrieval for open-domain question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5835–5847, Online. Association for Computational Linguistics.

Vatsal Raina and Mark Gales. 2024. Question-based retrieval using atomic units for enterprise RAG. In *Proceedings of the Seventh Fact Extraction and VERification Workshop (FEVER)*, pages 219–233, Miami, Florida, USA. Association for Computational Linguistics.

Stephen Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: Bm25 and beyond. *Found. Trends Inf. Retr.*, 3(4):333–389.

Gerard Salton and Chris Buckley. 1990. Improving retrieval performance by relevance feedback. *Journal of the American society for information science*, 41(4):288–297.

Xuehua Shen and ChengXiang Zhai. 2005. Active feedback in ad hoc information retrieval. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 59–66.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. BEIR: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

William R Thompson. 1933. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294.

Yujing Wang, Hainan Zhang, Liang Pang, Binghui Guo, Hongwei Zheng, and Zhiming Zheng. 2025. Maferw: Query rewriting with multi-aspect feedbacks for retrieval-augmented large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 25434–25442.

Zeyu Wang and Yu Wu. 2023. Click-Feedback Retrieval. *arXiv e-prints*, arXiv:2305.00052.

Haolun Wu, Ofer Meshi, Masrour Zoghi, Fernando Diaz, Xue Liu, Craig Boutilier, and MARYAM KARIMZADEHGAN. 2024. Density-based user representation using gaussian process regression for multi-interest personalized retrieval. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Yifei Yuan and Wai Lam. 2021. Conversational fashion image retrieval via multiturn natural language feedback. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, page 839–848, New York, NY, USA. Association for Computing Machinery.

Xiang Zhou. 2003. Relevance feedback in image retrieval: A comprehensive review. *Multimedia Syst.*, 8:536–544.

A Prompts for Synthetic Question Generation and Filtering

Figure 7 describes the prompt for query generation, which was designed to generate multiple synthetic queries from an original question while preserving semantic equivalence. N_{sq} represents number of synthetic queries to be generated per single original query. ($N_{sq} = 10$ in our experiments.) Figure 8 shows the prompt for query filtering, which was designed to enforce context-grounded, minimal answers for reliable automatic comparison. Only synthetic queries whose LLM-generated answers achieve a sufficiently high F1 score when compared with the answer to the original query are used for evaluation.

B Examples of Synthetic Questions

Table 1 and Table 2 represent examples of LLM-rephrased questions for NQ and FiQA, respectively. Due to space limitations, we will show one example of each dataset. We used GPT-4o-mini for reformation of of both datasets with using the prompt described in Appendix A.

C Results with various random seeds

Figure 9 shows the hit rates of proposed model with dynamic weights under different random seeds (color solid lines) and no weighting baselines by setting $w = 0$ using base score alone (black dotted line). As same as Figure 3, the hit rate is reported as a moving average computed over every 10% of total queries in each dataset. The results are obtained under an idealized assumption that all feedback is provided correctly. Among all the random seeds,

Table 1: Example of original and synthetic questions for BEIR/NQ and synthetic answers generated by GPT-4o-mini.

Original Query	GENERATED ANSWER	
WHEN DID AMERICAN TWO PARTY SYSTEM BEGAN TO EMERGE	1828	-
Synthetic Query	GENERATED ANSWER	F1 SCORE
WHAT YEAR DID THE AMERICAN TWO-PARTY SYSTEM START TO TAKE SHAPE?	1828	1.00
WHEN DID THE EMERGENCE OF THE TWO-PARTY SYSTEM IN AMERICA BEGIN?	THE EMERGENCE OF THE TWO-PARTY SYSTEM IN AMERICA BEGAN IN 1828.	0.20
AT WHAT POINT DID THE TWO-PARTY SYSTEM START DEVELOPING IN THE UNITED STATES?	AT THE END OF THE FIRST PARTY SYSTEM	0.00
CAN YOU TELL ME WHEN THE AMERICAN TWO-PARTY SYSTEM FIRST STARTED TO EMERGE?	THE FIRST PARTY SYSTEM	0.00
WHEN DID THE FORMATION OF A TWO-PARTY SYSTEM OCCUR IN AMERICA?	1828	1.00
IN WHICH ERA DID THE TWO-PARTY SYSTEM IN THE UNITED STATES BEGIN TO EVOLVE?	FIRST PARTY SYSTEM	0.00
WHEN DID THE AMERICAN POLITICAL LANDSCAPE START TO REFLECT A TWO-PARTY SYSTEM?	1828	1.00
WHAT WAS THE STARTING POINT FOR THE EMERGENCE OF THE TWO-PARTY SYSTEM IN THE U.S.?	THE FIRST PARTY SYSTEM	0.00
WHEN DID THE RISE OF THE TWO-PARTY SYSTEM IN AMERICAN POLITICS BEGIN?	BEGINNING IN 1828	0.50
WHAT TIME PERIOD MARKS THE BEGINNING OF THE AMERICAN TWO-PARTY POLITICAL SYSTEM?	1828	1.00

the proposed method continues to outperform the baseline without feedback.

System: You are an expert of rephrasing.
User: You are given a retrieval query: {original query}. Generate $\{N_{sq}\}$ alternative versions of this query.

Requirements:
- Preserve the original intent and meaning exactly.
- Rephrase using different words, synonyms, or sentence structures.
- Vary tone and style slightly (e.g., formal, casual, conversational).
- Do NOT answer the question itself.

Output format:
- List all $\{N_{sq}\}$ versions as bullet points only.
- Do not include explanations or extra commentary.

Figure 7: Prompt used for query generation.

Table 2: Example of original and synthetic questions for BEIR/fiqa and synthetic answers generated by GPT-4o-mini.

Original Query	GENERATED ANSWER	
INTENTIONS OF DEDUCTIBLE AMOUNT FOR SMALL BUSINESS	NET OPERATING LOSS OPTIONS	-
Synthetic Query	GENERATED ANSWER	F1 SCORE
PURPOSE OF DEDUCTIBLE AMOUNTS FOR SMALL ENTERPRISES	TO DECREASE FUTURE TAXES	0.00
GOALS OF DEDUCTIBLE FIGURES FOR SMALL BUSINESSES	NET OPERATING LOSS (NOL)	0.75
UNDERSTANDING THE DEDUCTIBLE AMOUNT INTENTIONS FOR SMALL BUSINESS OWNERS	NET OPERATING LOSS	0.86
SMALL BUSINESS DEDUCTIBLE AMOUNTS: WHAT ARE THEIR PURPOSES?	TO OFFSET TAXABLE INCOME IN CURRENT OR FUTURE YEARS	0.00
A LOOK AT DEDUCTIBLE AMOUNT INTENTIONS FOR SMALL BUSINESSES	NET OPERATING LOSS	0.86
THE RATIONALE BEHIND DEDUCTIBLE AMOUNTS FOR SMALL ENTERPRISES	NET OPERATING LOSS	0.86
EXPLORING THE INTENTIONS OF DEDUCTIBLE AMOUNTS IN SMALL BUSINESS CONTEXTS	NET OPERATING LOSS	0.86
WHAT SMALL BUSINESSES NEED TO KNOW ABOUT DEDUCTIBLE AMOUNT GOALS	NET OPERATING LOSS OPTIONS	1.00
THE INTENT BEHIND DEDUCTIBLE AMOUNTS FOR SMALL BUSINESS OPERATIONS	NET OPERATING LOSS	0.86
CLARIFYING THE INTENTIONS SURROUNDING DEDUCTIBLE AMOUNTS FOR SMALL BUSINESSES	NET OPERATING LOSS	0.86

User: You are given a question and a context. Answer the question using ONLY the information in the context.

Output rules:
- Output only the final answer as a short span.
- Do not include any explanation, punctuation, or extra words.

Question:
{question}

Context:
{supporting contexts}

Figure 8: Prompt used for query filtering.

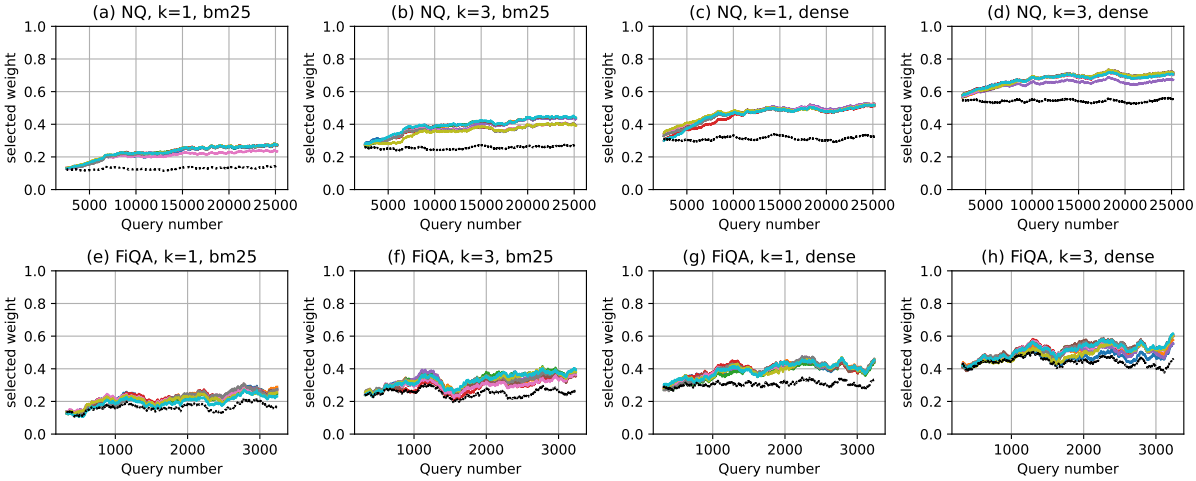

Figure 9: Hit rates of proposed model along query inputs using LLM-reformulated questions and dynamic weights. Different line colors correspond to different random seeds in dynamic weights. Black dotted lines indicate baselines ($w = 0$).