Adapting to Non-Stationary Environments: Multi-Armed Bandit Enhanced Retrieval-Augmented Generation on Knowledge Graphs

Xiaqiang Tang[1,2], **Jian Li**[2*], **Nan Du**[2], **Sihong Xie**[1*]

[1]The Hong Kong University of Science and Technology (Guangzhou)
[2]Tencent Hunyuan

Abstract

Despite the superior performance of Large language models on many NLP tasks, they still face significant limitations in memorizing extensive world knowledge. Recent studies have demonstrated that leveraging the Retrieval-Augmented Generation (RAG) framework, combined with Knowledge Graphs that encapsulate extensive factual data in a structured format, robustly enhances the reasoning capabilities of LLMs. However, deploying such systems in real-world scenarios presents challenges: the continuous evolution of non-stationary environments may lead to performance degradation and user satisfaction requires a careful balance of performance and responsiveness. To address these challenges, we introduce a Multi-objective Multi-Armed Bandit enhanced RAG framework, supported by multiple retrieval methods with diverse capabilities under rich and evolving retrieval contexts in practice. Within this framework, each retrieval method is treated as a distinct "arm". The system utilizes real-time user feedback to adapt to dynamic environments, by selecting the appropriate retrieval method based on input queries and the historical multi-objective performance of each arm. Extensive experiments conducted on two benchmark KGQA datasets demonstrate that our method significantly outperforms baseline methods in non-stationary settings while achieving state-of-the-art performance in stationary environments.

Code —
https://github.com/FUTUREEEEEE/Dynamic-RAG

1 Introduction

Large language models (LLMs) (Chowdhery et al. 2023; Achiam et al. 2023; Touvron et al. 2023) excel in natural language processing tasks (Bang et al. 2023; Brown et al. 2020) but struggle with knowledge-intensive challenges, often producing unfaithful or hallucinated information (Petroni et al. 2020; Ji et al. 2023). Retrieval-Augmented Generation (RAG) (Lewis et al. 2020) has been developed to enhance LLM reasoning, effectively reducing hallucinations and providing reliable, up-to-date information. In this approach, when presented with a user query, a retriever first extracts relevant information from a knowledge base, which is then provided to the LLM to generate the final response.Recent advancements (He et al. 2024; Luo et al. 2023c; Sun et al. 2023; Xu et al. 2024) in RAG systems have increasingly incorporated Knowledge graphs (KGs) (Baek, Aji, and Saffari 2023; Luo et al. 2023b) as the underlying knowledge base. KGs store vast amounts of factual data in a structured format, which enables more dependable and systematic reasoning by LLMs.

Unlike unstructured text databases (e.g. Wikipedia), the organized nature of KGs provides diverse retrieval methods, with significantly different capabilities and costs. For example, dense retrieval methods (Zhang et al. 2023; Yu et al. 2022) are typically fast but offer limited reasoning capabilities. In contrast, using LLMs to generate KG query languages (e.g., SPARQL) as in ChatKBQA (Luo et al. 2023a) provides high coverage and is suitable for multi-entity retrieval. Methods like RoG (Luo et al. 2023c), where LLMs function as search agents excel in complex reasoning. However, both methods require interactions with LLMs like ChatGPT (OpenAI 2024), leading to longer execution times. However, current KG-based RAG systems often rely solely on a single retrieval method or use static neural network routers (Reis et al. 2019; lla 2024), which require complete labeled data for supervision and periodic fine-tuning. Moreover, while RAG systems are often deployed in scenarios where users can provide feedback on generated responses (Gamage et al. 2024; Alan, Aydın, and Karaarslan 2024), current systems generally neglect this feedback. Relying on a single retrieval method, or computationally intensive ensemble all retrieval results can not ensure responses that are both timely and informative. On the other hand, static neural network routers cannot effectively leverage real-time feedback to continually adapt to changing user needs and system variability.

Therefore, deploying RAG systems in real-world scenarios faces the following challenges as described in Fig. 1: **(C1)**: Non-stationary environments require RAG systems to adapt to two sides continuously: on the user side, the evolving nature of queries driven by trending topics, and on the server side, the backend retrieval model upgrading. **(C2)**: In practical applications RAG systems, such as personal home assistants and customer support chatbots, balancing multi-objective, such as efficiency, coverage, and reasoning power, is crucial to providing informative and satisfying user expe-

*Corresponding author

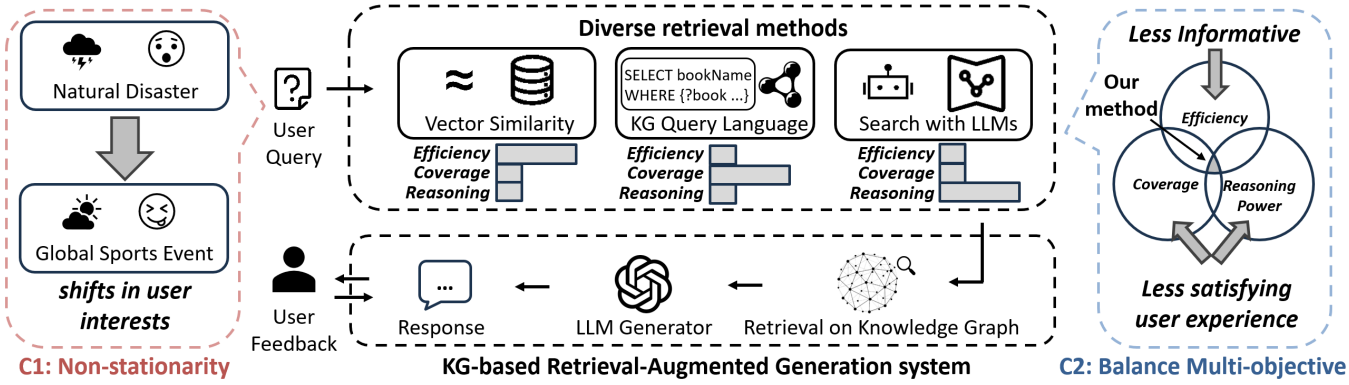

Figure 1: An online KG-based RAG system facing challenges from non-stationary environments and the need to balance multiple objectives for optimal user experience.

riences. Failing to address the diverse demands of queries and deliver timely, comprehensive responses can result in less informative interactions or unsatisfying user experience.

In response to (C1), we proposed a RAG framework enhanced by deep contextual Multi-arm Bandit(Collier and Llorens 2018), utilizing a lightweight language model as the backbone to interpret user queries and predict the suitability of each retrieval method. The model is updated on a per-query basis using feedback, ensuring robust performance and adaptability in non-stationary environments. In response to (C2), we incorporated the Generalized Gini Index to aggregate multi-objective user demands effectively, ensuring that no single objective dominates the other objective. By balancing retrieval coverage, accuracy, and response time, our framework enhances user experience by providing informative answers under time constraints.

Our main technical contributions are as follows:

- We enhanced KG-based RAG systems by employing an MAB model for dynamic retrieval selection and continuous adaptation to non-stationarity using user feedback.
- We utilized the Generalized Gini Index to aggregate multi-objective rewards, ensuring both informative and timely responses.
- We evaluated our framework on two well-established KBQA datasets. Our results demonstrate that our methods significantly outperform baseline approaches in non-stationary environments and surpass state-of-the-art KG-based RAG systems in stationary settings.

2 Method

The diverse capabilities of different retrieval methods necessitate a strategic model for their selection. Simply running multiple retrievers and then aggregating their results often proves sub-optimal due to two main factors: the need for timely responses and the disparate performance characteristics of various retrieval methods, as highlighted in Table 1. For example, while dense retrievers provide rapid responses, KG agent-based retrievers slow down the system due to LLM inference.

Consequently, we developed a model that dynamically assigns queries to the most suitable retrievers. Unlike static neural network routers, which require collecting complete labeled data for supervision (involving the execution of all retrieval methods) and periodic fine-tuning, limiting their adaptability to non-stationary environments. Our approach leverages real-time user feedback as a reward signal to update the model. This adaptability is crucial in the dynamic nature of RAG applications, such as shifting user interests and backend retriever upgrades requiring continuous optimization.

2.1 Problem Setup

The optimization of KG-based RAG systems employing multiple retriever backends and real-time feedback is structured as follows:

- Initially, the system receives a user input query context x.
- The PLM model f_θ processes the query and selects an action a from the action space A, which includes K potential retrieval methods, each representing an arm in a multi-armed bandit.
- Upon selection, the system receives feedback on the performance of the chosen retrieval method a (e.g. 1 indicating a good response, 0 indicating a bad response), providing "partial-information" feedback. This limitation restricts the system's ability to assess unselected methods.
- Utilizing this feedback, the model iteratively refines its strategy to improve base retrieval method selection for future queries.

2.2 Deep Multi-objective Contextual Bandits

Query Encoding Model: In order to effectively select retrieval methods, it is crucial to discern patterns within user queries and associate these with the capabilities of suitable retrieval methods. Traditional linear models in contextual bandits (Li et al. 2010; Mehrotra, Xue, and Lalmas 2020), while effective in certain scenarios, often fall short due to the complex natural language patterns present in user queries.

To address limitations and ensure real-time service, we utilize the lightweight Pre-trained Language Model, DistilBERT (Sanh et al. 2019). As a streamlined version of BERT, DistilBERT retains approximately 97% of BERT's language understanding capabilities and increases processing speed

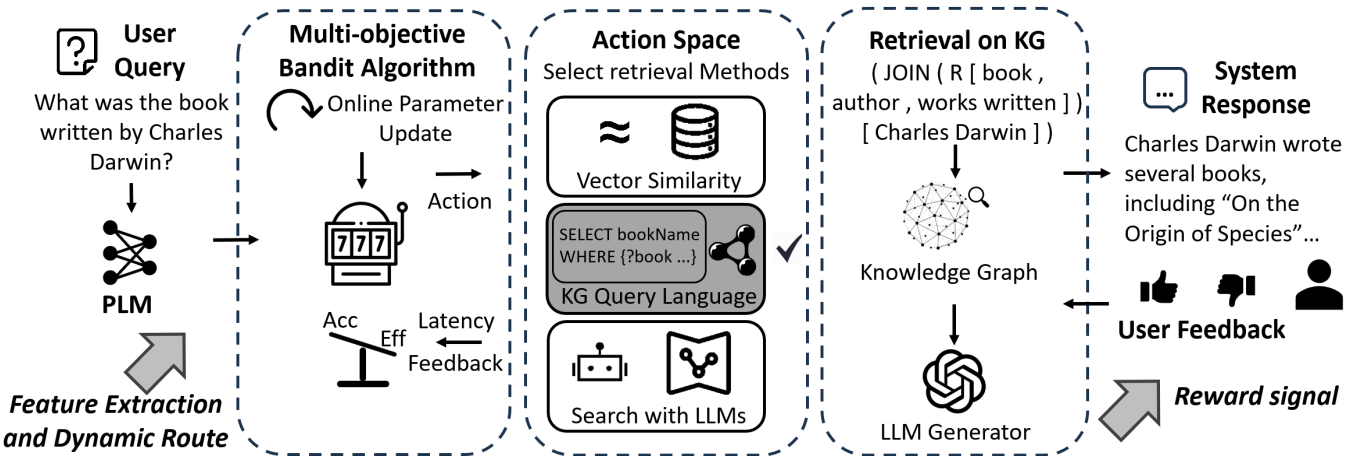

Figure 2: Proposed MAB-enhanced RAG framework. The input query undergoes feature extraction (e.g., multi-entity query), followed by the MAB algorithm, which selects the optimal retrieval method by predicting the most rewarding option (e.g., Query Language method). The selected method retrieves information from a Knowledge Graph (KG), and an LLM generates the final response. Feedback is collected as a reward, updating the MAB model parameters online, and enabling continuous adaptation to non-stationary environments.

by 60%. This model provides a robust and nuanced approach to modeling natural language queries, which facilitates the precise identification of appropriate retrieval methods for individual queries and supports continuous real-time refinement through user feedback.

Specifically, our query encoding model, f_θ, uses DistilBERT to efficiently interpret natural language, taking a query as input context and producing an arm selection distribution $z = f_\theta(x)$.

Arm Selection Strategy: Upon receiving an action distribution estimation $\mathbf{z} = f_\theta(x)$ from the encoding model, we employ an epsilon-greedy strategy (Langford and Zhang 2007) to balance the trade-off between exploration and exploitation (Auer 2002; Auer, Cesa-Bianchi, and Fischer 2002). This balance is crucial in ensuring that the system not only leverages the information gathered so far (exploit known retrieval methods that have proven effective) but also explores new possibilities to enhance learning (explore other retrieval methods that could potentially offer better results). Specifically:

- With a probability of $1-\epsilon$, the system selects the arm with the highest predicted reward, $a = \max(\mathbf{z})$, based on the output from the encoding model.
- Conversely, with a probability of ϵ, the system explores by randomly selecting an arm, facilitating the discovery of potentially more effective retrieval methods.

This strategy enables the system to predominantly rely on the best-known actions to maximize immediate rewards while maintaining the flexibility to explore new possibilities. This approach is essential to mitigate the risk of converging to a locally optimal model due to partial information feedback, fostering the discovery of superior long-term solutions through randomized exploration.

Learning Algorithm: After selecting a retrieval method, our model updates based on the observation associated with the chosen method, but it does not have access to the information from methods not selected (i.e., partial information feedback). Inspired by "offline-to-online" learning(Lee et al. 2022; Guo et al. 2024), we first pre-train the model in an offline environment to learn a robust initial strategy. Subsequently, we fine-tune the model in an online setting using partial user feedback, allowing it to adapt continuously to real-world conditions.

Traditional RAG systems often focus on optimizing model accuracy. However, real-world applications of RAG systems demand not only accuracy but also real-time responsiveness, introducing the need for multi-objective optimization. We use the Generalized Gini Index (Weymark 1981) to balance system performance with retrieval time, ensuring both accuracy and efficiency are optimized simultaneously.

During training, we use detailed evaluation metrics, including informativeness measures like hit and recall, to optimize for accuracy and coverage. Retrieval latency is also used as feedback to enhance efficiency. In testing, we simulate an online environment with a hit value (0 or 1) to approximate binary user feedback. This offline-to-online learning approach ensures the model is well-prepared before deployment and can adapt effectively to dynamic user interactions.

The methodology underpinning our approach is detailed in Algorithm 1. We have designed our system with a focus on three critical objectives to evaluate the performance of the retrieval methods comprehensively, enhancing the overall functionality of our RAG system. Each query's final response, generated by the LLM in natural language, is assessed according to these objectives, which include accuracy and efficiency metrics.

Accuracy Metrics: For accuracy, we consider two key metrics: hit (h, whether the response contains the correct answer) and recall (rc, which assesses the system's ability to retrieve all relevant items). These metrics are crucial for assessing the precision and completeness of the system.

Algorithm 1: Deep GGI-MO bandit enhanced RAG learning algorithm

Input: The query context set X, pre-trained language model parameters θ.
Initialize: Set equal initial weights for the $\mathbf{w}$.
for $x \in X$ **do**
    Encode the query x and obtain the estimated action distribution $z = f_\theta(x)$.
    Select an retriever (arm) a based on the selection strategy described in Section 2.2.
    Observe the retrieval context to compute the loss components as per Eq. (2) and the execution time d for the retrieval process.
    Update the model weights θ by minimizing the loss $Loss_{GGI}(\theta)$ using gradient descent.
end for
Output: Updated model weights θ.

Efficiency Metrics: Efficiency is evaluated based on the mean delay time (d_i) experienced by each retrieval method within the system. We utilize a distribution $\sigma(d_i)$ to quantitatively represent each method's efficiency. This distribution helps the model understand the temporal performance across different retrieval strategies, ensuring the system delivers not only accurate but also timely responses.

$$\sigma(d_i) = \frac{e^{1/d_i}}{\sum_{j=1}^{K} e^{1/d_j}}, \tag{1}$$

where methods with longer delays are assigned lower values, thus incentivizing quicker retrieval methods.

Multi-Objective Optimization with GGI: To balance these objectives, we compute the multi-objective GGI value, which integrates accuracy and efficiency metrics, further detail of GGI property can be found in subsection 1 in the appendix (Tang et al. 2024). The GGI values for each objective are calculated as follows:

$$l_1 = MSE(\max(f_\theta(x)), h), \quad \text{(Loss- Accuracy - Hit)} \tag{2}$$

$$l_2 = MSE(\max(f_\theta(x)), rc), \quad \text{(Loss-Accuracy - Recall)} \tag{3}$$

$$l_3 = KLDiv(f_\theta(x), \sigma(d_i)), \quad \text{(Loss- Efficiency)} \tag{4}$$

Each loss component l_i corresponds to a specific objective:

- l_1 and l_2 measures the deviation in accuracy, encouraging the model to select a method with a high probability produce high recall and hit retrieval methods.
- l_3 quantifies the efficiency using the Kullback-Leibler Divergence (KLDiv) between the predicted arm selection distribution from the model and the efficiency distribution $\sigma(d_i)$ encourage the model to select an efficient retrieval method.

The aggregate loss function to be minimized, representing the overall GGI, is then given by:

$$Loss = GGI_w(\mathbf{l}) = \sum_{i=1}^{D} w_i (l_i)_\tau = \mathbf{w}^T (\mathbf{l})_\tau \tag{5}$$

where $w_1 > w_2 > \cdots > w_d > 0$ and τ permutes the elements of $\mathbf{l}$ such that $(l_i)_\tau > (l_{i+1})_\tau$.

In Equation 5, the GGI function aggregates the individual loss components, weighted by w_i, to update the parameter θ, optimizing towards a better response quality with satisfying user experience.

3 Experiment

3.1 Dataset & Setup

Datasets: We evaluate our systems on two KGQA datasets WebQSP (Yih et al. 2016) and ComplexWebQuestions (CWQ) (Talmor and Berant 2018) which contain up to 4-hop questions. The statistics of the datasets are given in appendix (Tang et al. 2024).

Baselines: To valid the effectiveness of our MAB-enhanced KG-based RAG system under stationary environment, we compared it with state-of-the-art KG-based RAG systems, including the query language-based RAG: StructGPT (Jiang et al. 2023), ChatKBQA(Luo et al. 2023a), LLM agent-based RAG: Think-on-Graph (Sun et al. 2023) and Reason-on-Graph (Luo et al. 2023c), and dense retrieval based RAG (Zhang et al. 2023; Yu et al. 2022).

Evaluation Metrics: Following previous works, we evaluate the performance of our Retrieval-Augmented Generation system, all results are assessed based on the final generated response's hit rate and recall. We ran at least ten independent rounds with different seeds and reported the results as mean ± standard deviation to ensure the stability of our findings.

Implementations: We consistently use Llama-2-7b-chat-hf (Touvron et al. 2023) as the LLM generator, applying a standard RAG prompt (LlamaIndex 2024) across all methods to ensure a fair comparison. All experiments are conducted on the Nvidia Tesla V100 graphical card with the Intel Xeon Platinum 8255C CPU. See subsection 3 in the appendix (Tang et al. 2024) for detail set up.

3.2 Research Questions and Main Results

RQ1: Can our Multi-Armed Bandit enhanced Retrieval-Augmented Generation system effectively improve performance compared to RAG systems that rely on a single retrieval method?

The comparative analysis, summarized in Table 1, revealed that our MAB-enhanced RAG system demonstrated superior performance across both datasets. Notably, on the CWQ dataset, which poses more intricate multi-hop reasoning challenges, our method exceeds the next-best performance by nearly 2% in hit rate and over 2.5% in recall.

We present examples of our MAB-enhanced RAG systems superior case.

In the first case Fig. 3, from the WebQSP dataset where the user queries, "What are some books that Mark Twain wrote?" This question is challenging in terms of achieving high recall since all retrieval methods can provide related context, but not all can accurately list the books. Our MAB-enhanced RAG system effectively selects the appropriate methods (SPARQL generator) to achieve the highest recall, significantly outperforming individual retrieval approaches.

Retriever Type	Method	WebQSP		CWQ	
		Hit ↑	Recall ↑	Hit ↑	Recall ↑
Dense Retrieval	BGE (Zhang et al. 2023)	63.03	44.43	52.46	46.68
	DECAF (Yu et al. 2022)	71.37	50.93	47.46	41.47
KG Query Languge Retrieval	StructGPT (Jiang et al. 2023)	75.56	55.26	\	\
	ChatKBQA (Luo et al. 2023a)	80.77	64.31	77.37	69.46
LLM agent Retreival	Think-on-graph (Sun et al. 2023)	66.64	47.24	58.90	52.49
	Reason-on-graph (Luo et al. 2023c)	85.70	75.07	56.63	52.38
Ensemble (DECAF+ChatKBQA+Reason-on-graph)		83.74	67.52	67.93	68.01
Static Router	LLM Router (lla 2024)	82.48	68.9	65.75	59.33
	NN-Router (Reis et al. 2019)	86.20	75.03	78.53	71.52
Ours	GGI-MAB	**86.64**	**75.60**	**79.35**	**72.02**

Table 1: Results under Stationary environment

Non-stationarity	Method	Test Hit ↑	Test Recall ↑	Test Retrieval Delay ↓ (second per query)
Retriever update	Retrieval Ensemble	83.74 ± 0.58	67.52 ± 0.77	15.00 ± 0.00
	Offline MO-MAB	82.25 ± 2.18	66.05 ± 2.56	13.32 ± 1.90
	NN Router (Reis et al. 2019)	81.48 ± 0.28	64.90 ± 0.23	14.09 ± 0.42
	LLM Router (lla 2024)	82.19 ± 0.47	67.11 ± 0.52	10.36 ± 3.98
	Ours	**84.80 ± 0.39**	**72.24 ± 0.71**	**5.88 ± 0.99**
Domain shift	Retrieval Ensemble	67.93 ± 0.61	68.01± 0.34	15.00 ± 0.00
	Offline MO-MAB	64.76 ± 4.40	61.32 ± 2.90	7.78 ± 0.44
	NN Router (Reis et al. 2019)	69.57 ± 4.52	63.99 ± 3.94	**7.65 ± 1.81**
	LLM Router (lla 2024)	65.75 ± 0.16	59.33 ± 0.25	9.39 ± 0.06
	Ours	**76.35 ± 0.69**	**69.47 ± 0.76**	11.63 ± 0.28

Table 2: Results under Non-stationary environment (mean ± std)

In the second case in Figure 7 in the appendix (Tang et al. 2024) derived from the challenging CWQ dataset, the query pertains to the birthplace of the lyricist for "Stop Standing There." Dense retrieval fails to relate the query to relevant information, and while the SPARQL retriever approaches a correct formulation, it ultimately generates a wrong query language. Thanks to the reasoning ability of LLM, the LLM-based KG agent successfully retrieves the related triplets from the knowledge graph and enables our MAB Enhanced RAG system to give an accurate response.

Our MAB-enhanced RAG system, effectively optimizes the selection process of retrieval methods, thereby proving to be highly effective in improving overall system performance. Furthermore, as illustrated in Figure 5 in the appendix (Tang et al. 2024), we evaluate our method across different Large Language Model generators to prove the robustness of our system.

RQ2: Can the MAB enhanced RAG system adapts dynamically to the non-stationary nature of real-world environments, ensuring that they continuously meet evolving query demands and operational conditions?

To evaluate our methods under non-stationary environments, we use two non-stationary settings: (1) we employed the KG agent-based retrieval method (Sun et al. 2023) during the training phase. For online testing, we switched to (Luo et al. 2023c) a method with superior performance, to simulate the effect of upgrading backend retrievers independently to enhance system functionality. This approach tests the system's ability to adapt seamlessly to improvements in retrieval methods, reflecting real-world conditions where continuous updates are crucial for maintaining system efficacy. (2) To simulate the shift in query domains resulting from changes in trending topics, we initially train our methods using the WebQSP dataset. Subsequently, we evaluate the system's adaptability by testing it on the ComplexWebQuestions dataset. This approach allows us to assess how well the system can handle transitions between different types of query complexities and content, mirroring real-world scenarios where query characteristics can vary significantly due to external influences.

The results, as detailed in Table 2, in the first scenario, during the retriever upgrade tests, our method demonstrated the highest Test Hit and Test Recall rates f 84.80% and 72.24% respectively, with a significantly reduced Test Retrieval Delay of 5.88 seconds per query. This improvement stems from our system's capability to leverage partial information during testing to continuously refine the model. In contrast, retrieval ensemble methods, which require running

	Method	Test Hit ↑	Test Recall ↑	Test Retrieval Delay ↓ (second per query)
Baselines	UCB (Auer 2002)	78.44 ± 6.07	62.18 ± 10.08	5.80 ± 6.08
	Thompsom Sampling (Agrawal and Goyal 2013)	84.12 ± 2.20	71.82 ± 4.93	6.60 ± 5.50
	LinUCB (Li et al. 2010)	81.99 ± 2.91	68.55 ± 5.30	5.31 ± 2.80
	SO-Deep-MAB (Collier and Llorens 2018)	86.79 ± 0.33	75.18 ± 0.18	11.1 ± 0.39
	MOU-UCB(Wanigasekara et al. 2019)	85.55 ± 0.89	75.05 ± 0.15	5.52 ± 1.35
Ours	MO-MAB	85.31 ± 0.55	74.38 ± 0.48	5.20 ± 0.58
	GGI-MO-MAB	**86.64 ± 0.29**	**75.60 ± 0.38**	**4.84 ± 0.81**

Table 3: Results of proposed multi-objective MAB algorithm under station environments (mean ± std)

Question: What are some books that mark twain wrote?
Gold Answer: 'Adventures of Huckleberry Finn', 'Tom Sawyer Detective' , 'Life on the Mississippi'……

Dense Retrieval:
'Mark Twain Mark Twain (November 30, 1835 – April 21, 1910), real name Samuel Langhorne Clemens, was an American writer, humorist, entrepreneur, publisher, and lecturer. Among his novels are "The Adventures of Tom Sawyer" (1876) and its sequel, the "Adventures of Huckleberry Finn" (1885), the latter often called "The Great American Novel". …. **(FAST BUT LOW RECALL)**

KG-Agent-Retriever:
Mark Twain -> book.written_work.author -> Mark Twain's correspondence with Henry Huttleston Rogers, 1893-1909
Mark Twain -> book.written_work.author -> Is He Dead?
…. **(SLOW WITH LOW RECALL)**

SPARQL-Retriever:
['(JOIN (R [book , author , book editions published]) [Mark Twain]) **(EFFICIENT AND ACCUATE)** ✓

Ours MAB Enhanced RAG system response:
Mark Twain wrote many books throughout his career, including:
1. "The Adventures of Tom Sawyer" (1876)
2. "Sketches New and Old" (1875)… ✓

Figure 3: Comparison of retrieval methods for the query, "What are some books that Mark Twain wrote?" Dense Retrieval is fast but has low recall, while KG-Agent-Retriever provides broad coverage but is slow. Our system selects the SPARQL-Retriever (Luo et al. 2023a), which generates an accurate search language command for precise and efficient results.

all retrieval methods, struggle with denoising information from different structures of retrieval results leading to the longest Retrieval delay. Both the offline classifier and offline multi-objective MAB (MO-MAB) were unable to adapt to the upgrades, resulting in inferior performance.

In the second scenario, our approach effectively adapted to domain shifts by utilizing the slower KG agent retriever and SPARQL generator retrieval methods. Although it needs more retrieval time at 11.63 seconds per query compared to some offline methods, it significantly outperformed comparative methods in accuracy metrics, achieving a Test Hit rate of 76.35% and a Test Recall of 69.47%.

The results further highlight our system's capacity to dynamically adjust operational parameters in response to evolving query complexities, ensuring high-quality user interactions even under challenging conditions.

RQ3: How can the Generalized Gini Index be effectively utilized to balance multiple performance metrics in RAG systems

In Table 3 we evaluate our proposed method on the WebQSP dataset, results highlight the effectiveness of the Generalized Gini Index enhanced Multi-Objective Multi-Armed Bandit (GGI-MO-MAB), achieving the highest Test Hit rate and Test Recall, while maintaining the lowest retrieval delay compared to the baselines. Non-contextual baselines like UCB (Auer 2002) and Thompson Sampling (Agrawal and Goyal 2013) approximate only a single optimal retrieval method. LinUCB (Li et al. 2010) under-performs due to its inability to handle the high-dimensional, complex natural language embeddings. Single-objective deep contextual MAB models, while improving accuracy metrics such as Hit rate, often neglect retrieval time, adversely affecting user experience. Our GGI-MO-MAB can also outperform multi-objective baseline MOU-UCB (Wanigasekara et al. 2019). To underscore the efficacy of our approach, we include an ablation study comparing the GGI function to a learnable weight aggregation baseline (MO-MAB), confirming the robust performance improvement of our method.

RQ4: What are the effects of implementing multiple retrieval methods, such as dense retrieval and KG agent retrieval methods, on the response times and accuracy under different real-world scenarios?

The comparison of different types of retrieval methods is shown in Fig. 4, we also employed the Jaccard Similarity Coefficient to assess the Hit metric across results. Our findings reveal an average coefficient of 0.738, with the lowest observed at 0.496, indicating the distinctiveness of the results obtained by different retrieval strategies.

Moreover, while methods such as ChatKBQA and Reason-on-graph showed strong results on WebQSP, they were less effective on the more challenging CWQ dataset, highlighting the importance of retrieval method selection based on the complexity and nature of the dataset. Our system's consistent performance across different datasets underscores its robustness and adaptability, making it particularly suitable for diverse real-world applications where query demands and operational conditions can vary significantly.

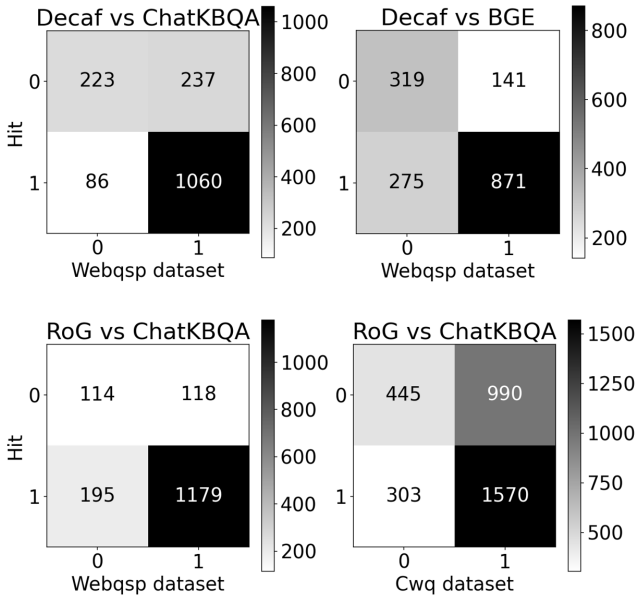

Figure 4: Confusion matrices comparing retrieval methods (Decaf, ChatKBQA, BGE, RoG) on WebQSP and CWQ datasets, indicating distinctiveness among methods. (0: Miss, 1: Hit)

In terms of response time, we observed significant differences in processing time; for instance, dense-vector retrieval methods average around 1 second, whereas more complex methods like ChatKBQA (Luo et al. 2023a), due to multiple interactions with ChatGPT (OpenAI 2024) to get an executable query code, can take 15-30 seconds. These findings highlight the trade-off between complexity and efficiency in retrieval operations.

Our findings confirm that the choice of retrieval method significantly impacts the accuracy and efficiency of RAG systems.

4 Related Work

KG-based RAG Systems: Retrieval-Augmented Generation (RAG)(Lewis et al. 2020) mitigates the hallucination issue of LLMs by retrieving external knowledge to enhance the accuracy and reliability of generation content. Recent RAG advancements have increasingly incorporated Knowledge Graphs (KGs) (Luo et al. 2023c; Sun et al. 2023; Xu et al. 2024; He et al. 2024), which store structured factual information, enabling more systematic reasoning by LLMs (Pan et al. 2024). KGs support diverse retrieval methods, each with different capabilities and costs, as detailed in appendix (Tang et al. 2024).

Our analysis of retrieval methods, discussed in Section 3.2, shows that current KG-based RAG systems(Luo et al. 2023c; Sun et al. 2023; Xu et al. 2024; He et al. 2024) predominantly rely on a single retrieval method, which often fails to meet the varied demands of real-world applications. These systems typically assume a stationary environment and remain static without subsequent fine-tuning, making them unable to adapt to potential shifts in the query domain and upgrades of the backend retriever. To address these issues,our work aims to develop an MAB-enhanced RAG system that strategically combines multiple retrievers. By leveraging real-time feedback, our system can dynamically adjust retrieval strategies to meet the evolving demands of diverse application scenarios of the RAG system effectively.

To our knowledge, the concurrent research by (Sawarkar, Mangal, and Solanki 2024) is one of the few studies attempting to integrate multiple retrieval methods, but it focuses on textual data sources and lacks the continuous optimization crucial for RAG systems in non-stationary environments.

Multi-Armed Bandit Algorithms: The Multi-Armed Bandit (MAB) (Katehakis and Veinott Jr 1987) framework optimizes the balance between exploiting historical data and exploring new information. It includes two main types: context-free (Bubeck, Cesa-Bianchi et al. 2012), which operates without external information, and contextual bandits (Mahajan and Teneketzis 2008), which incorporate contextual data such as user features. Traditional contextual bandits assume a linear relationship between context and expected rewards (Slivkins 2011), but recent developments have introduced non-linear models through deep learning (Collier and Llorens 2018; Zhou, Li, and Gu 2020; Shi et al. 2023). The multi-objective contextual MAB (MOCMAB) algorithm (Tekin and Turğay 2018) maximizes rewards across multiple objectives, managing both dominant and non-dominant goals. Some approaches (Busa-Fekete et al. 2017; Mehrotra, Xue, and Lalmas 2020) use the Generalized Gini Index (GGI) to convert multi-objective challenges into single-objective optimizations, simplifying decision-making in dynamic environments.

However, existing contextual bandit algorithms often assume reward is linear with respect to the context feature (Tekin and Turğay 2018; Mehrotra, Xue, and Lalmas 2020; Li et al. 2010), limiting their representational capacity to match user query patterns with retrieval strategies effectively, or they focus solely on single-objective optimization (Collier and Llorens 2018; Zhou, Li, and Gu 2020; Shi et al. 2023), which does not suffice for complex RAG systems with requirements of performance and real-time limitation. Therefore, in this work, we adopt a non-linear multi-objective contextual MAB model.

5 Conclusion

In this work, we introduced a novel KG-based RAG framework enhanced by a Multi-Armed Bandit (MAB) model. By leveraging real-time user feedback, our system dynamically adapts to shifting query demands and backend upgrades. We further incorporated the Generalized Gini Index to balance multiple objectives, ensuring that the system delivers both informative and timely responses.

Our comprehensive evaluations on two well-established KBQA datasets, WebQuestionSP and ComplexWebQuestions, demonstrate that our approach not only significantly outperforms baseline methods in non-stationary environments but also surpasses state-of-the-art KG-based RAG systems in stationary settings. These results underscore the robustness, adaptability, and practical applicability of our framework in real-world scenarios where query demands and operational conditions are constantly evolving.

Acknowledgments

Sihong Xie was supported in part by the National Key R&D Program of China (Grant No. 2023YFF0725001), the Guangzhou-HKUST(GZ) Joint Funding Program (Grant No. 2023A03J0008), and Education Bureau of Guangzhou Municipality.

References

2024. Define Selector Module for Routing — Llama Index Documentation. https://docs.llamaindex.ai/en/stable/examples/retrievers/router_retriever/#define-selector-module-for-routing. Accessed: 2024-08-07.

Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Agrawal, S.; and Goyal, N. 2013. Thompson sampling for contextual bandits with linear payoffs. In *International conference on machine learning*, 127–135. PMLR.

Alan, A. Y.; Aydın, Ö.; and Karaarslan, E. 2024. A RAG-based Question Answering System Proposal for Understanding Islam: MufassirQAS LLM. *Available at SSRN 4707470*.

Auer, P. 2002. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, 3(Nov): 397–422.

Auer, P.; Cesa-Bianchi, N.; and Fischer, P. 2002. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47: 235–256.

Baek, J.; Aji, A. F.; and Saffari, A. 2023. Knowledge-augmented language model prompting for zero-shot knowledge graph question answering. *arXiv preprint arXiv:2306.04136*.

Bang, Y.; Cahyawijaya, S.; Lee, N.; Dai, W.; Su, D.; Wilie, B.; Lovenia, H.; Ji, Z.; Yu, T.; Chung, W.; et al. 2023. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity. *arXiv preprint arXiv:2302.04023*.

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

Bubeck, S.; Cesa-Bianchi, N.; et al. 2012. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends® in Machine Learning*, 5(1): 1–122.

Busa-Fekete, R.; Szörényi, B.; Weng, P.; and Mannor, S. 2017. Multi-objective bandits: Optimizing the generalized Gini index. In *International Conference on Machine Learning*, 625–634. PMLR.

Chowdhery, A.; Narang, S.; Devlin, J.; Bosma, M.; Mishra, G.; Roberts, A.; Barham, P.; Chung, H. W.; Sutton, C.; Gehrmann, S.; et al. 2023. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240): 1–113.

Collier, M.; and Llorens, H. U. 2018. Deep contextual multi-armed bandits. *arXiv preprint arXiv:1807.09809*.

Gamage, G.; Mills, N.; De Silva, D.; Manic, M.; Moraliyage, H.; Jennings, A.; and Alahakoon, D. 2024. Multi-Agent RAG Chatbot Architecture for Decision Support in Net-Zero Emission Energy Systems. In *2024 IEEE International Conference on Industrial Technology (ICIT)*, 1–6. IEEE.

Guo, S.; Zou, L.; Chen, H.; Qu, B.; Chi, H.; Yu, P. S.; and Chang, Y. 2024. Sample Efficient Offline-to-Online Reinforcement Learning. *IEEE Transactions on Knowledge and Data Engineering*, 36(3): 1299–1310.

He, X.; Tian, Y.; Sun, Y.; Chawla, N. V.; Laurent, T.; LeCun, Y.; Bresson, X.; and Hooi, B. 2024. G-Retriever: Retrieval-Augmented Generation for Textual Graph Understanding and Question Answering. *arXiv preprint arXiv:2402.07630*.

Ji, Z.; Lee, N.; Frieske, R.; Yu, T.; Su, D.; Xu, Y.; Ishii, E.; Bang, Y. J.; Madotto, A.; and Fung, P. 2023. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12): 1–38.

Jiang, J.; Zhou, K.; Dong, Z.; Ye, K.; Zhao, W. X.; and Wen, J.-R. 2023. Structgpt: A general framework for large language model to reason over structured data. *arXiv preprint arXiv:2305.09645*.

Katehakis, M. N.; and Veinott Jr, A. F. 1987. The multi-armed bandit problem: decomposition and computation. *Mathematics of Operations Research*, 12(2): 262–268.

Langford, J.; and Zhang, T. 2007. The epoch-greedy algorithm for multi-armed bandits with side information. *Advances in neural information processing systems*, 20.

Lee, S.; Seo, Y.; Lee, K.; Abbeel, P.; and Shin, J. 2022. Offline-to-online reinforcement learning via balanced replay and pessimistic q-ensemble. In *Conference on Robot Learning*, 1702–1712. PMLR.

Lewis, P.; Perez, E.; Piktus, A.; Petroni, F.; Karpukhin, V.; Goyal, N.; Küttler, H.; Lewis, M.; Yih, W.-t.; Rocktäschel, T.; et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33: 9459–9474.

Li, L.; Chu, W.; Langford, J.; and Schapire, R. E. 2010. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*, 661–670.

LlamaIndex. 2024. Prompt Engineering for RAG. Accessed: 2024-05-16. URL: https://docs.llamaindex.ai/en/stable/examples/prompts/prompts_rag/.

Luo, H.; Tang, Z.; Peng, S.; Guo, Y.; Zhang, W.; Ma, C.; Dong, G.; Song, M.; Lin, W.; et al. 2023a. Chatkbqa: A generate-then-retrieve framework for knowledge base question answering with fine-tuned large language models. *arXiv preprint arXiv:2310.08975*.

Luo, L.; Ju, J.; Xiong, B.; Li, Y.-F.; Haffari, G.; and Pan, S. 2023b. Chatrule: Mining logical rules with large language models for knowledge graph reasoning. *arXiv preprint arXiv:2309.01538*.

Luo, L.; Li, Y.-F.; Haffari, G.; and Pan, S. 2023c. Reasoning on graphs: Faithful and interpretable large language model reasoning. *arXiv preprint arXiv:2310.01061*.

Mahajan, A.; and Teneketzis, D. 2008. Multi-armed bandit problems. In *Foundations and applications of sensor management*, 121–151. Springer.

Mehrotra, R.; Xue, N.; and Lalmas, M. 2020. Bandit based optimization of multiple objectives on a music streaming platform. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, 3224–3233.

OpenAI. 2024. ChatGPT. https://openai.com/chatgpt. Accessed: 2024-05-20.

Pan, S.; Luo, L.; Wang, Y.; Chen, C.; Wang, J.; and Wu, X. 2024. Unifying large language models and knowledge graphs: A roadmap. *IEEE Transactions on Knowledge and Data Engineering*.

Petroni, F.; Piktus, A.; Fan, A.; Lewis, P.; Yazdani, M.; De Cao, N.; Thorne, J.; Jernite, Y.; Karpukhin, V.; Maillard, J.; et al. 2020. KILT: a benchmark for knowledge intensive language tasks. *arXiv preprint arXiv:2009.02252*.

Reis, J.; Rocha, M.; Phan, T. K.; Griffin, D.; Le, F.; and Rio, M. 2019. Deep Neural Networks for Network Routing. In *2019 International Joint Conference on Neural Networks (IJCNN)*, 1–8.

Sanh, V.; Debut, L.; Chaumond, J.; and Wolf, T. 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Sawarkar, K.; Mangal, A.; and Solanki, S. R. 2024. Blended RAG: Improving RAG (Retriever-Augmented Generation) Accuracy with Semantic Search and Hybrid Query-Based Retrievers. *arXiv preprint arXiv:2404.07220*.

Shi, Q.; Xiao, F.; Pickard, D.; Chen, I.; and Chen, L. 2023. Deep neural network with linucb: A contextual bandit approach for personalized recommendation. In *Companion Proceedings of the ACM Web Conference 2023*, 778–782.

Slivkins, A. 2011. Contextual Bandits with Similarity Information. In Kakade, S. M.; and von Luxburg, U., eds., *Proceedings of the 24th Annual Conference on Learning Theory*, volume 19 of *Proceedings of Machine Learning Research*, 679–702. Budapest, Hungary: PMLR.

Sun, J.; Xu, C.; Tang, L.; Wang, S.; Lin, C.; Gong, Y.; Shum, H.-Y.; and Guo, J. 2023. Think-on-Graph: Deep and Responsible Reasoning of Large Language Model with Knowledge Graph. arXiv:2307.07697.

Talmor, A.; and Berant, J. 2018. The web as a knowledge-base for answering complex questions. *arXiv preprint arXiv:1803.06643*.

Tang, X.; Li, J.; Du, N.; and Xie, S. 2024. Adapting to Non-Stationary Environments: Multi-Armed Bandit Enhanced Retrieval-Augmented Generation on Knowledge Graphs. *arXiv preprint arXiv:2412.07618*.

Tekin, C.; and Turğay, E. 2018. Multi-objective contextual multi-armed bandit with a dominant objective. *IEEE Transactions on Signal Processing*, 66(14): 3799–3813.

Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Wanigasekara, N.; Liang, Y.; Goh, S. T.; Liu, Y.; Williams, J. J.; and Rosenblum, D. S. 2019. Learning Multi-Objective Rewards and User Utility Function in Contextual Bandits for Personalized Ranking. In *IJCAI*, volume 19, 3835–3841.

Weymark, J. A. 1981. Generalized Gini inequality indices. *Mathematical Social Sciences*, 1(4): 409–430.

Xu, Z.; Cruz, M. J.; Guevara, M.; Wang, T.; Deshpande, M.; Wang, X.; and Li, Z. 2024. Retrieval-Augmented Generation with Knowledge Graphs for Customer Service Question Answering. *arXiv preprint arXiv:2404.17723*.

Yih, W.-t.; Richardson, M.; Meek, C.; Chang, M.-W.; and Suh, J. 2016. The value of semantic parse labeling for knowledge base question answering. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 201–206.

Yu, D.; Zhang, S.; Ng, P.; Zhu, H.; Li, A. H.; Wang, J.; Hu, Y.; Wang, W.; Wang, Z.; and Xiang, B. 2022. Decaf: Joint decoding of answers and logical forms for question answering over knowledge bases. *arXiv preprint arXiv:2210.00063*.

Zhang, P.; Xiao, S.; Liu, Z.; Dou, Z.; and Nie, J.-Y. 2023. Retrieve anything to augment large language models. *arXiv preprint arXiv:2310.07554*.

Zhou, D.; Li, L.; and Gu, Q. 2020. Neural contextual bandits with ucb-based exploration. In *International Conference on Machine Learning*, 11492–11502. PMLR.