

# MBA-RAG: a Bandit Approach for Adaptive Retrieval-Augmented Generation through Question Complexity

Xiaqiang Tang <sup>\*1,2</sup>, Qiang Gao <sup>\*2,3</sup>, Jian Li <sup>\*2</sup>, Nan Du<sup>2</sup>,  
Qi Li<sup>4</sup>, Sihong Xie <sup>†1</sup>,

<sup>1</sup>The Hong Kong University of Science and Technology (Guangzhou),

<sup>2</sup>Tencent Hunyuan, <sup>3</sup>Wuhan University, <sup>4</sup>Iowa State University,

Correspondence: [sihongxie@hkust-gz.edu.cn](mailto:sihongxie@hkust-gz.edu.cn)

## Abstract

Retrieval Augmented Generation (RAG) has proven to be highly effective in boosting the generative performance of language model in knowledge-intensive tasks. However, existing RAG frameworks either indiscriminately perform retrieval or rely on rigid single-class classifiers to select retrieval methods, leading to inefficiencies and suboptimal performance across queries of varying complexity. To address these challenges, we propose a reinforcement learning-based framework that dynamically selects the most suitable retrieval strategy based on query complexity. Our approach leverages a multi-armed bandit algorithm, which treats each retrieval method as a distinct “arm” and adapts the selection process by balancing exploration and exploitation. Additionally, we introduce a dynamic reward function that balances accuracy and efficiency, penalizing methods that require more retrieval steps, even if they lead to a correct result. Our method achieves new state-of-the-art results on multiple single-hop and multi-hop datasets while reducing retrieval costs. Our code is available at <https://github.com/FUTUREEEEEEE/MBA>

## 1 Introduction and Related Work

Retrieval Augmented Generation (Lewis et al., 2020) has shown significant promise in addressing knowledge-intensive natural language processing (NLP) tasks by integrating an updated knowledge base with a language model. This combination allows language model to access up-to-date knowledge, improving their faithfulness and reducing hallucinations.

Most existing RAG frameworks employ a retrieve-and-generate setup that indiscriminately performs retrieval based on the input. However, this approach may hinder the versatility of language models or introduce unnecessary or off-topic

passages. To address this issue, methods such as SEAKR (Yao et al., 2024) and FLARE (Jiang et al., 2023) have been designed to perform active retrieval only when necessary.

Furthermore, AdaptiveRAG (Jeong et al., 2024) argues that real-world queries often vary in difficulty, such as the number of reasoning steps required or the depth of information needed to answer a query. Thus, applying a single retrieval method across all queries can be ineffective. Simple queries may incur unnecessary computational overhead when complex retrieval strategies are used, while complex, multi-step queries may not be adequately addressed. To tackle this issue, AdaptiveRAG introduces an adaptive router that selects the retrieval method based on the complexity of the query. However, AdaptiveRAG simplifies the generation strategy into a single-choice task by using heuristic supervision that favors only one process with the least retrieval cost. This supervision is inaccurate for two main reasons. First, it assumes only one strategy is optimal for one query. For example, while multiple strategies (such as directly answering without retrieval, retrieving once, or performing iterative retrieval) might all provide correct answers depending on the query, only the direct answer method will be marked as correct. This strict supervision ignores scenarios where more complex strategies might offer better context or more comprehensive answers. Second, this heuristic approach is ambiguous because retrieval costs vary according to the query’s difficulty, what constitutes the least cost can differ significantly between queries of different complexities.

To overcome these limitations, we propose a **Multi-arm Bandit-based framework for Adaptive Retrieval-Augmented Generation (MBA-RAG)** Fig. 1, that introduces both flexibility and cost-awareness into the generation process.

First, to address the rigidity of single-label supervision, we employ a multi-armed bandit algo-

\*Equally contributed authors

†Corresponding author

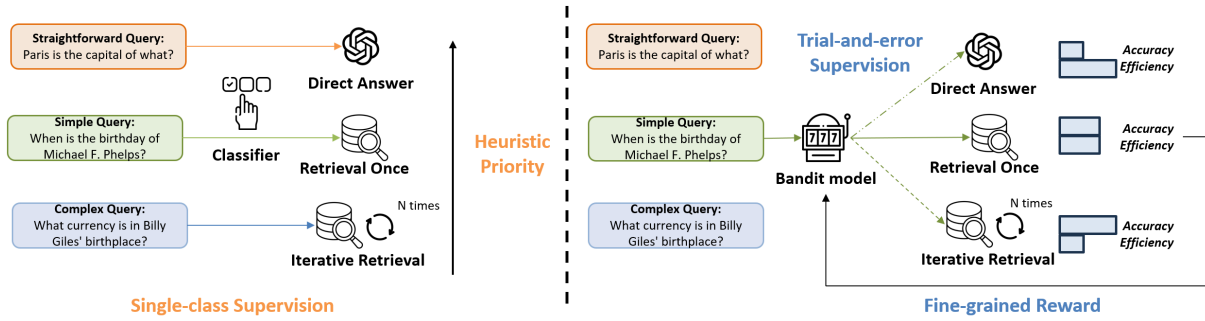


Figure 1: On the left, the AdaptiveRAG pipeline relies on an inaccurate heuristic approach to assign queries of different complexities to a single generation process. On the right, we propose an MBA-RAG framework that allows the model to explore the merits of different generation methods and rewards it based on fine-grained assessments of generation quality and cost.

rithm (Katehakis and Veinott Jr, 1987). The bandit framework requires only partial information for supervision; specifically, the model is supervised by the feedback of the chosen generation strategy, without indiscriminately penalizing the strategies that were not selected. This introduces greater flexibility by allowing the model to explore different generation strategies and obtain a more comprehensive selection.

Second, to tackle the issue of inaccurate heuristic supervision of retrieval costs and the varying costs associated with different generation strategies, we design a dynamic, fine-grained reward function. This function penalizes inefficient generation strategies, even if they lead to correct results, ensuring that the model not only prioritizes accuracy but also optimizes computational efficiency.

We evaluated our framework on both single-hop and multi-hop datasets. Results demonstrate that our method achieves SOTA generation accuracy while reducing retrieval costs by 20%.

## 2 Method

Using a single retrieval method for all queries may not always be effective. In this section, we introduce a RAG framework based on a multi-arm bandit approach that dynamically selects the optimal generation method to balance generation quality and computational efficiency.

### 2.1 Preliminaries

The RAG process (Lewis et al., 2020) includes a retrieval stage where the module  $R$  retrieves relevant documents  $D$  for a query  $x$ , and a generation stage where the LLM uses  $x$  and  $D$  to generate the response  $\bar{a} = LLM(y_t|x, D)$ .

### 2.2 Query Encoding and Arm Selection

To overcome the limitations of attributing each query to a single generation method, we reformulate the generation selection as a trial-and-error reinforcement learning problem, where each generation method is treated as an action (i.e. an arm) in a multi-armed bandit framework.

We employ DistilBERT, a lightweight pre-trained language model (PLM), to encode user queries, generating an action distribution  $\mathbf{z} = f_{\theta}(x)$ . Directly supervising this output could cause the system to overly favor a single generation method, leading to sub-optimal local convergence.

To prevent this, we incorporate an epsilon-greedy strategy (Langford and Zhang, 2007) to balance the trade-off between exploration and exploitation (Auer, 2002; Auer et al., 2002). This ensures that the system primarily utilizes the best-performing generation methods while still exploring other options to improve long-term performance. Specifically, the “arm” is chosen as  $a = \arg \max(\mathbf{z})$  with probability  $1 - \epsilon$ , or a random generation method with probability  $\epsilon$ .

### 2.3 Learning Algorithm

After selecting a generation method, the model updates its parameters based on both the quality and the cost of the language model’s response. Unlike traditional RAG systems that prioritize accuracy alone, our approach incorporates a fine-grained reward function that accounts for both accuracy and computational cost, thereby guiding the model to minimize unnecessary computational overhead.

The objective function, given by

$$\min_{\theta} (r_a - f_{\theta}(x)_a)^2 \quad (1)$$

is to minimize the squared error between the actual reward  $r_a$  and the predicted reward  $f_{\theta}(x)_a$  for

action  $a$ :

$$\theta_{t+1} = \theta_t - \alpha \nabla_{\theta} \left( (r_a - f_{\theta}(x)_a)^2 \right)$$

Here,  $\theta_{t+1}$  represents the parameter set at iteration  $t + 1$ ,  $\theta_t$  at iteration  $t$ , and  $\alpha$  is the learning rate. The reward  $r_a$  for the selected retrieval method  $a$  is calculated as  $r_a = \mathcal{A}(y, \hat{y}_a) - \lambda C(a)$ , where  $\mathcal{A}$  is a generation quality metric (e.g., exact match defined as  $\mathcal{A}(y, \hat{y}_a) = \mathbb{I}\{y = \hat{y}_a\}$  if the generated answer  $\hat{y}_a$  matches the ground truth  $y$ ),  $C(a)$  represents the computational cost of method  $a$  (such as the number of retrieval steps), and  $\lambda$  is a scaling factor balancing accuracy and efficiency.

### 3 Experiments

#### 3.1 Datasets

To demonstrate the performance of our method for queries of different complexities, following the work of Adaptive-RAG (Jeong et al., 2024), we selected three single-hop QA and three multi-hop QA datasets as our experimental datasets.

For Single-hop QA, we choose **SquAD v1.1** (Rajpurkar et al., 2016), **Natural Questions** (Kwiatkowski et al., 2019) and **TriviaQA** (Joshi et al., 2017). These datasets consist of each query and its related articles that contain the answers.

For Multi-hop QA, we choose **MuSiQue** (Trivedi et al., 2022b), **HotpotQA** (Yang et al., 2018) and **2WikiMultiHopQA** (Ho et al., 2020). For these multi-hop datasets, the answers cannot be directly obtained and require multi-step reasoning based on the information in the articles to arrive at the final answer.

#### 3.2 Baseline Models

We selected a range of related models as baselines to compare with our model, including:

**1) No-Retrieval:** Directly generates answers without performing retrieval. **2) Adaptive-Retrieval** (Mallen et al., 2023): Dynamically determining whether retrieval is necessary. **3) Self-RAG** (Asai et al., 2023): Dynamically determines whether retrieval is needed. **4) DRAGIN** (Su et al., 2024): evaluating the uncertainty of each token to activate the retrieval model. **5) SEAKR** (Yao et al., 2024) introduce self-aware uncertainty to decide whether activating the retrieval model based on its value. **6) Adaptive-RAG** (Jeong et al., 2024): Using a classifier to dynamically choose the most suitable retrieval strategy based on the complexity of the query.

#### 3.3 Metrics

Following the approach of Jeong et al. (2024), in addition to reporting performance metrics of the generated results such as Exact Match (EM),  $F_1$ , and Accuracy (Acc), we also report efficiency metrics of the retrieval strategy, Step. **EM** measures whether the predicted result exactly matches the ground truth,  **$F_1$**  measures the overlap of words between the predicted answer and the ground truth, and **Acc** indicates whether the predicted answer contains the ground truth. **Step** denotes the number of retrieval steps required by the selected retrieval strategy.

#### 3.4 Experiment Settings

We use DistilBERT (Sanh et al., 2020) as the query encoding model. This model allows for efficient query encoding and ensures that the retrieval method selection is based on a robust representation of the query. The retrieval settings, dataset configurations, and generation model setups all follow the approach used by Adaptive-RAG (Jeong et al., 2024). We utilize BM25 as the retrieval model and FLAN-T5-XL (Chung et al., 2022) as the generation model. For the MBA-RAG retrieval, we employ the epsilon-greedy algorithm (Sutton and Barto, 1998) as the action strategy. The learning rate for the encoding model is set as 5e-5.

Methods	FLAN-T5-XL (3B)			
	EM	$F_1$	Acc	Step
No Retrieval*	14.87	21.12	15.97	0.00
Adaptive Retrieval*	23.87	32.24	26.73	0.50
Self-RAG*	9.90	20.79	31.57	0.72
Adaptive-RAG*	37.17	46.94	42.10	2.17
Adaptive-RAG (DistilBert)	34.37	43.80	38.50	1.69
MBA-RAG (Ours)(DistilBert)	<b>38.80</b>	<b>48.61</b>	<b>43.57</b>	<u>1.80</u>
MBA-RAG (Ours)(T5-Large)	38.40	48.38	43.30	1.95

Table 1: Averaged results on a collection of benchmark datasets. Items marked with \* have results from Jeong et al. (2024). Adaptive-RAG (DistilBert) means we use DistilBert as classifier in Adaptive-RAG settings.

#### 3.5 Main Results

We report the average results across six datasets as shown in Table 1, and the results for each individual dataset as shown in Table 2. Note that while we focus on performance metrics such as EM,  $F_1$ , and Acc, we also consider efficiency metrics like Step. The experimental results demonstrate that our method achieves a balance between the two, surpassing the baseline in performance metrics while requiring relatively fewer steps.

As shown in Table 2, our method has achieved performance improvements on all single-step datasets, with significant gains observed on the

Data	Methods	SQuAD				Natural Questions				TriviaQA			
		EM	$F_1$	Acc	Step	EM	$F_1$	Acc	Step	EM	$F_1$	Acc	Step
Single-step	No Retrieval*	3.60	10.50	5.00	0.00	14.20	19.00	15.60	0.00	25.00	31.80	27.00	0.00
	Adaptive Retrieval*	13.40	23.10	17.60	0.50	28.20	36.00	33.00	0.50	38.40	46.90	42.60	0.50
	Self-RAG*	2.20	11.20	18.40	0.63	31.40	39.00	33.60	0.63	12.80	29.30	57.00	0.68
	DRAGIN <sup>†</sup>	18.70	28.70	–	–	23.20	33.20	–	–	54.00	62.30	–	–
	SEAKR <sup>†</sup>	27.10	36.50	–	–	25.60	35.50	–	–	<b>54.40</b>	<b>63.10</b>	–	–
	Adaptive-RAG*	26.80	38.30	33.00	<u>1.37</u>	37.80	47.30	44.60	<u>1.00</u>	52.20	60.70	58.20	<u>1.23</u>
	Adaptive-RAG (DistilBert)	22.60	33.90	28.20	1.57	34.60	43.00	39.80	1.58	49.60	57.50	54.80	1.32
	MBA-RAG (Ours)(DistilBert)	<b>27.60</b>	<b>39.10</b>	<b>33.80</b>	<u>1.11</u>	<b>37.80</b>	<b>47.50</b>	<b>44.60</b>	<u>1.23</u>	53.60	62.40	<b>60.20</b>	<u>1.06</u>
	MBA-RAG (Ours)(T5-Large)	27.20	19.00	33.40	1.32	<b>37.80</b>	47.30	44.55	1.10	53.20	62.30	60.00	1.00
Data	Methods	MuSiQue				HotpotQA				2WikiMultiHopQA			
		EM	$F_1$	Acc	Step	EM	$F_1$	Acc	Step	EM	$F_1$	Acc	Step
Multi-step	No Retrieval*	2.40	10.70	3.20	0.00	16.60	22.71	17.20	0.00	27.40	32.04	27.80	0.00
	Adaptive Retrieval*	6.40	15.80	8.00	0.50	23.60	32.22	25.00	0.50	33.20	39.44	34.20	0.50
	Self-RAG*	1.60	8.10	12.00	0.73	6.80	17.53	29.60	0.73	4.60	19.59	38.80	0.93
	DRAGIN <sup>†</sup>	–	–	–	–	23.70	34.20	–	–	22.40	30.0	–	–
	SEAKR <sup>†</sup>	–	–	–	–	27.90	39.70	–	–	30.20	36.0	–	–
	Adaptive-RAG*	23.60	31.80	<b>26.00</b>	<u>3.22</u>	<b>42.00</b>	<b>53.82</b>	<b>44.40</b>	<u>3.55</u>	40.60	49.75	46.40	<u>2.63</u>
	Adaptive-RAG (DistilBert)	22.20	30.90	23.80	2.22	35.80	47.68	38.00	1.69	41.40	49.70	46.40	1.78
	MBA-RAG (Ours)(DistilBert)	<b>23.80</b>	<b>31.90</b>	25.40	<u>2.56</u>	40.60	52.44	42.60	<u>2.25</u>	<b>49.40</b>	<b>58.33</b>	<b>54.60</b>	<u>2.57</u>
	MBA-RAG (Ours)(T5-Large)	21.20	30.90	23.80	2.31	<b>42.00</b>	53.30	44.30	2.70	49.20	58.30	<b>54.60</b>	2.93

Table 2: Results on each of a collection of datasets with FLAN-T5-XL (3B) as the LLM. Items marked with \* have results from Jeong et al. (2024), while items marked with <sup>†</sup> have results from Yao et al. (2024), which uses LLaMA-2-7B-Chat as the backbone LLM.

SQuAD and TriviaQA datasets. In particular, our approach achieves a substantial reduction in step costs, suggesting that it can dynamically select retrieval strategies with fewer retrieval steps. On the NQ dataset, our results are roughly on par with Adaptive-RAG, but with an increase in steps. This is because Adaptive-RAG, when constructing the dataset, set the condition that when zero, one, or multiple retrieval paths could yield the correct answer, the path requiring the fewest steps would be chosen as the correct label, with others marked as incorrect. However, this does not reflect reality as other paths can also yield correct answers. Therefore, our reward model does not solely choose the retrieval path with the minimum steps but may also select other feasible paths.

For the multi-step dataset, we have achieved a reduction in step costs of over 20%, clearly demonstrating a significant enhancement in the retrieval efficiency of our method. Our method achieved significant improvements on the 2wikiMultiHopQA. On the MusiQue dataset, we also achieved results comparable to Adaptive-RAG, but with significantly lower step costs. This indicates that when two retrieval strategies can achieve comparable performance metrics, our strategy incurs lower retrieval costs. However, on the HotpotQA dataset, our method slightly underperformed Adaptive-RAG in terms of EM, Accuracy, and  $F_1$ , while having lower step costs. This may be because, compared to Adaptive-RAG which uses T5-large (770M) as the classifier, our method uses DistilBERT (66M), which is 10 times smaller. Un-

like the other two datasets, HotpotQA is a very challenging dataset with complex question types like actoid comparison questions that demand high general knowledge performance from the model. DistilBERT’s knowledge capabilities are not on par with T5. Nevertheless, our MBA-RAG still achieved more optimal strategy selection, with step costs lower than Adaptive-RAG, indicating that our MBA-RAG-based method has balanced performance metrics with efficiency metrics.

Furthermore, as shown in Table 4, we compare our bandit approach with a multi-label classifier. Due to the classifier’s inability to consider retrieval costs, it incurs excessive computational expenses, with the majority of queries opting for a multiple retrieval strategy. For more details, please refer to Appendix A.1.

**Retrieval Strategy Performance** To measure the performance of different retrieval strategies, we compared the accuracy of label selection between our MBA-RAG-based classification strategy and other adaptive retrieval strategies. The results, as shown in Figure 2, indicate that our classification accuracy is significantly higher than that of Adaptive-RAG. This suggests that our multi-label supervising strategy is better suited to adapt to queries of varying complexities, leading to an overall improvement in results.

## 4 Conclusion

In this study, we introduced an adaptive retrieval-augmented generation framework using a multi-

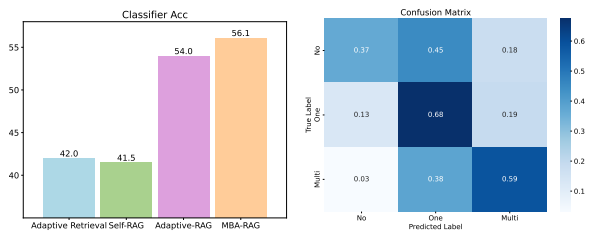


Figure 2: On the left: classification accuracy of different adaptive approaches and the confusion matrix across all six datasets. On the right: we analyze MBA-RAG performance across different complexity labels, presenting the overall experimental results across six datasets.

armed bandit approach that dynamically selects the most efficient retrieval strategy based on query complexity. Our experimental results demonstrate the effectiveness of this method in reducing computational costs while maintaining high accuracy. Future work will aim to further refine this approach and explore its applicability to a broader range of NLP tasks.

## Limitations

Our reinforcement learning-based approach for adaptive retrieval in language models demonstrates promising results; however, it is not without limitations. The framework’s dependence on the specific structure of the multi-armed bandit algorithm can introduce challenges in scalability and adaptability to new, unseen query types. Future work could explore more efficient algorithms that maintain performance while reducing computational demands.

## References

Asari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2023. [Self-rag: Learning to retrieve, generate, and critique through self-reflection](#). *Preprint*, arXiv:2310.11511.

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

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

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams

Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. [Scaling instruction-finetuned language models](#). *Preprint*, arXiv:2210.11416.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020. [Constructing a multi-hop QA dataset for comprehensive evaluation of reasoning steps](#). In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6609–6625, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Soyeong Jeong, Jinheon Baek, Sukmin Cho, Sung Ju Hwang, and Jong C Park. 2024. [Adaptive-rag: Learning to adapt retrieval-augmented large language models through question complexity](#). *arXiv preprint arXiv:2403.14403*.

Zhengbao Jiang, Frank F Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. 2023. [Active retrieval augmented generation](#). *arXiv preprint arXiv:2305.06983*.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. [TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension](#). In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada. Association for Computational Linguistics.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick SH Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *EMNLP (1)*, pages 6769–6781.

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

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. [Natural questions: A benchmark for question answering research](#). *Transactions of the Association for Computational Linguistics*, 7:452–466.

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

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

- Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. [When not to trust language models: Investigating effectiveness of parametric and non-parametric memories](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9802–9822, Toronto, Canada. Association for Computational Linguistics.
- Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. [SQuAD: 100,000+ questions for machine comprehension of text](#). In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.
- Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2020. [Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter](#). *Preprint*, arXiv:1910.01108.
- Weihang Su, Yichen Tang, Qingyao Ai, Zhijing Wu, and Yiqun Liu. 2024. [DRAGIN: Dynamic retrieval augmented generation based on the real-time information needs of large language models](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12991–13013, Bangkok, Thailand. Association for Computational Linguistics.
- Weigao Sun, Zhen Qin, Dong Li, Xuyang Shen, Yu Qiao, and Yiran Zhong. 2024a. [Linear attention sequence parallelism](#). *arXiv preprint arXiv:2404.02882*.
- Weigao Sun, Zhen Qin, Weixuan Sun, Shidi Li, Dong Li, Xuyang Shen, Yu Qiao, and Yiran Zhong. 2024b. [Co2: Efficient distributed training with full communication-computation overlap](#). *arXiv preprint arXiv:2401.16265*.
- R.S. Sutton and A.G. Barto. 1998. [Reinforcement learning: An introduction](#). *IEEE Transactions on Neural Networks*, 9(5):1054–1054.
- Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2022a. [Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions](#). *arXiv preprint arXiv:2212.10509*.
- Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2022b. [MuSiQue: Multi-hop questions via single-hop question composition](#). *Transactions of the Association for Computational Linguistics*, 10:539–554.
- Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. [HotpotQA: A dataset for diverse, explainable multi-hop question answering](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, Brussels, Belgium. Association for Computational Linguistics.
- Zijun Yao, Weijian Qi, Liangming Pan, Shulin Cao, Linmei Hu, Weichuan Liu, Lei Hou, and Juanzi Li. 2024. [Seakr: Self-aware knowledge retrieval for adaptive retrieval augmented generation](#). *Preprint*, arXiv:2406.19215.
- Zhaofan Zhang, Yanan Xiao, Lu Jiang, Dingqi Yang, Minghao Yin, and Pengyang Wang. 2024. [Spatial-temporal interplay in human mobility: A hierarchical reinforcement learning approach with hypergraph representation](#). In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 9396–9404.

## A Experiments Appendix

### A.1 Implementation Details

For the external document corpus, we use different sources depending on the dataset type: the Wikipedia corpus preprocessed by (Karpukhin et al., 2020) for single-hop datasets, and the preprocessed corpus by (Trivedi et al., 2022a) for multihop datasets. Approaches such as those proposed by (Sun et al., 2024b,a) can be utilized to accelerate the training process.

### A.2 Reward Setting

For single-hop and multi-hop datasets, due to the differences in their tendencies for retrieval strategy selection, the single-hop dataset tends to choose the “zero” and “one” retrieval strategies, while the multi-hop dataset favors the “one” and “multiple” strategies. Therefore, for the MAB algorithm, the same set of rewards cannot achieve optimal results on both types of datasets simultaneously. Motivated by (Zhang et al., 2024), during our experiments, we set two different groups of reward settings.

For the single-hop dataset, we adopted a dynamic reward approach, combining strategy selection with step costs to balance performance and efficiency, the rewards are set at 1, 0.9, and 1-step/10 respectively for the three retrieval strategies “zero,” “one,” and “multiple.”. When none of the three options can answer the question, the reward is set to -1.

For the multi-hop dataset, rewards are set at 4.3, 2.3, and 1.15.

### A.3 Multi-label Classification

Due to Adaptive-RAG defining the selection of retrieval strategies as a single-label classification problem, where multiple strategies can yield correct results, it selects the strategy with the fewest

Data	Methods	SQuAD				Natural Questions				TriviaQA			
		EM	$F_1$	Acc	Step	EM	$F_1$	Acc	Step	EM	$F_1$	Acc	Step
Single-step	Adaptive-RAG	26.80	38.30	33.00	<u>1.37</u>	37.80	47.30	44.60	<u>1.00</u>	52.20	60.70	58.20	<u>1.23</u>
	Multi-label-classifier (Ours)	24.20	35.40	29.40	<u>4.392</u>	<b>38.6</b>	<b>47.70</b>	44.60	<u>4.514</u>	53.60	62.20	60.00	<u>5.152</u>
	MBA-RAG (Ours)	<b>27.60</b>	<b>39.10</b>	<b>33.80</b>	<u>1.11</u>	37.80	47.50	<b>44.60</b>	<u>1.23</u>	<b>53.60</b>	<b>62.40</b>	<b>60.20</b>	<u>1.06</u>

Data	Methods	MuSiQue				HotpotQA				2WikiMultiHopQA			
		EM	$F_1$	Acc	Step	EM	$F_1$	Acc	Step	EM	$F_1$	Acc	Step
Multi-step	Adaptive-RAG	23.60	31.80	<b>26.00</b>	<u>3.22</u>	42.00	53.82	44.40	<u>3.55</u>	40.60	49.75	46.40	<u>2.63</u>
	Multi-label-classifier (Ours)	23.00	31.90	25.80	<u>3.562</u>	<b>44.40</b>	<b>56.52</b>	<b>46.80</b>	<u>5.32</u>	<b>49.80</b>	<b>59.05</b>	<b>55.60</b>	<u>4.144</u>
	MBA-RAG (Ours)	<b>23.80</b>	<b>31.90</b>	25.40	<u>2.56</u>	40.60	52.44	42.60	<u>2.25</u>	49.40	58.33	54.60	<u>2.57</u>

Table 3: Results of multi-label classification on each of a collection of datasets with FLAN-T5-XL (3B) as the LLM.

steps as the correct label, treating other choices as incorrect. This approach is somewhat unreasonable because the other choices are also correct, albeit with a higher step cost. Therefore, to make the classification choice more rational while considering step costs, we implemented a multi-label classification setup, treating all retrieval strategies that could yield the correct answer as correct labels. We trained the classification model to predict multiple potential labels and selected the most likely label during the final inference.

### A.3.1 Setting

In the experimental setup, for the Multi-label classifier, we maintained consistency with the Adaptive-RAG’s model setting, using T5-large as the classifier and modifying the model to a multi-label classification head. The learning rate was set to  $1e-4$ . We employ multi-label training and use single-label form for inference, ultimately selecting retrieval generation according to the corresponding retrieval strategy. The experimental results are shown in Tables 4 and 3.

Methods	FLAN-T5-XL (3B)			
	EM	$F_1$	Acc	Step
Adaptive-RAG	37.17	46.94	42.10	<u>2.17</u>
Multi-label-classifier (Ours)	<b>38.93</b>	<b>48.79</b>	<b>43.70</b>	<u>4.514</u>
MBA-RAG (Ours)	38.80	48.61	43.57	<u>1.80</u>

Table 4: Averaged results of multi-label classification on a collection of benchmark datasets including single-hop and multi-hop queries.

## A.4 Results

The experimental results show that classifiers trained with multi-labels are highly prone to overfitting. Due to the high proportion of the “multiple” labels, the final model tends to exclusively choose “multiple.” While this achieves good performance in metrics, the step cost is excessively high, even for single-step datasets. Such choices are overly inefficient. In contrast, our MBA-RAG is essentially a form of multi-label classification, and the

dynamic reward settings enable the model to make the most rational choices based on the combination of step costs, making it more practical and feasible.

Although the Adaptive-RAG-MultiLabel yields generally high results, this is because, under this training mode, the model ultimately opts for “multiple” labels. During the construction of the training set, a large number of samples are tagged with multiple labels, leading the model to completely overfit on the “multiple” label after training. However, this approach results in an excessively high number of steps, and most problems do not require multi-step retrieval.