AutoRaC: An Automatic Retrieval Data Construction Method Based on Multimodal Large Language Model Preferences

Anonymous ACL submission

Abstract

Multimodal retrievers play a pivotal role in multimodal retrieval-augmented generation, their performance directly determines the quality of acquired external knowledge. Since the retriever's effectiveness is highly dependent on the accuracy and coverage of its training data, the quality and diversity of retrieval training data become critically important. However, existing multimodal retrieval training data construction approaches primarily rely on imprecise pseudo-relevance and single-document paradigms within isolated knowledge base, resulting in inaccurate relevance annotations, limited expansion of external knowledge bases, and failure to simultaneously guarantee accuracy and diversity in data construction. To address these challenges, we propose An **Auto**matic **R**etrieval **Da**ta **C**onstruction Method Based on Multimodal Large Language Model Preferences (AutoRaC), which implements MLLM-preference-guided construction through a two-stage filtering pipeline, automatically generating high-fidelity retrieval data while enabling knowledge base expansion, thereby enhancing data diversity. Results on InfoSeek and EVQA demonstrate that our method achieves accurate relevance annotations while also enabling knowledge base expansion, with the constructed data matching the quality of existing high-quality datasets.

1 Introduction

Multimodal retrievers (Kong et al., 2025; Zhou et al., 2024) play a critical role in visual information-seeking tasks (Chen et al., 2023; Mensink et al., 2023), addressing the inability of multimodal large language models (MLLMs) to respond adequately due to insufficient parametric knowledge. The multimodal retriever takes an image and a question as query, retrieves top-k relevant documents from knowledge base, and concatenates these documents with original query for MLLMs to generate knowledge-grounded answers, thereby improving its performance in scenarios that require external knowledge. Researches show retrieval data (training data for retriever) plays a pivotal role in enhancing retriever performance (Neelakantan et al., 2022). When the retrieval data covers broader knowledge domains, the model's generalization capability improves significantly (Xiong et al., 2020). Consequently, constructing diverse high-quality multimodal retrieval data is essential.

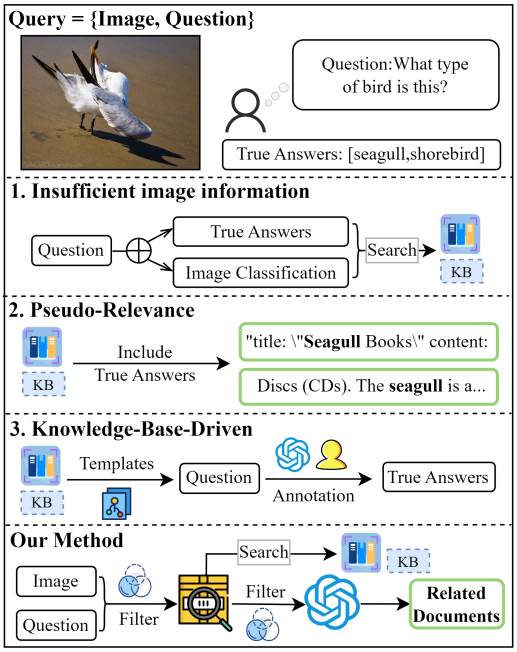

Figure 1: Mainstream methods of multimodal retrieval data construction. KB: Knowledge Base.

Recent years have seen significant progress in multimodal retrieval data construction research (Schwenk et al., 2022; Mekala et al., 2022). OKVQA (Marino et al., 2019) generates search queries by combining image classifications with textual questions, retrieving relevant articles via Wikipedia API as knowledge sources. (Hu et al., 2023) automatically matches Wikipedia entries as

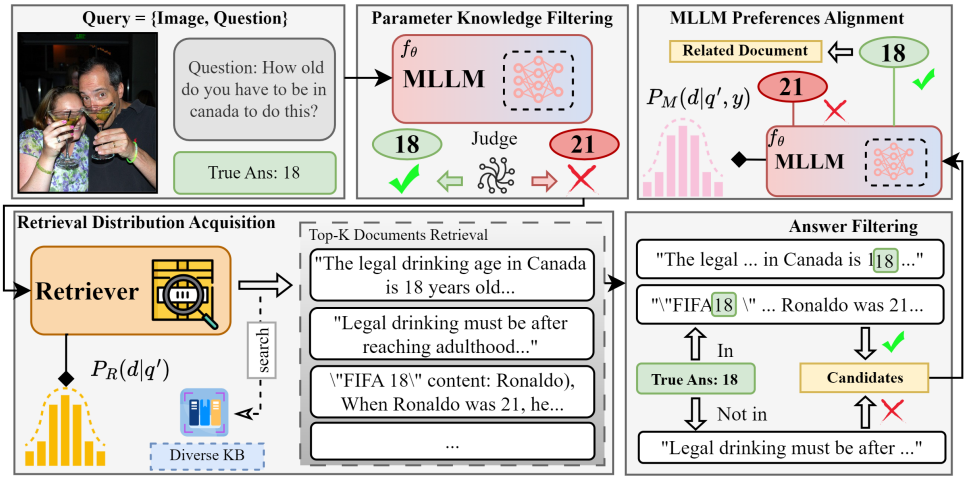

Figure 2: Overview of AutoRaC. Our objective is to minimize two distributions generated by retriever and MLLM.

relevant documents through the GENRE entity linking system (Cao et al., 2021). (Luo et al., 2021) constructs queries by combining textual questions with corresponding answers and then retrieves relevant document snippets through Google Search API. However, these methods rely on unimodal information or simplistic multimodal fusion strategies, failing to fully capture complex vision-text interactions. REPLUG (Shi et al., 2024) optimizes retrievers using language model but cannot distinguish between retrieved knowledge and parametric knowledge, lacking interpretability. (Lin et al., 2023) constructs a Wikipedia-based knowledge base through pseudo-relevance, determining document relevance by answer inclusion, which risks erroneous judgments. While these methods enable diverse data construction, the accuracy of their relevance annotations still has room for improvement. InfoSeek (Chen et al., 2023) manually annotates Wikipedia answers for visual entity queries while converting Wikidata triples into QA pairs via templates. EVQA (Mensink et al., 2023) automatically converts Wikidata triples into questions using templates and generates answers via pretrained language models. While these methods achieve accurate relevance annotations, their knowledge-base-driven nature inherently limits the full exploitation of knowledge base potential, hindering knowledge scalability and impeding diverse data construction. Although each approach exhibits distinct strengths, none can simultaneously satisfy both accuracy and diversity requirements in multimodal retrieval data construction.

In this work, to address the limitations of existing methods, we propose **AutoRaC**, an MLLM-preference-based, QA-pair-driven automated data construction method that ensures both annotation accuracy and knowledge base extensibility, thereby generating high-quality and diverse retrieval data, as shown in Figure 1. The relevance of external knowledge documents depends on whether their integration enables correct question answering (Sun et al., 2024). Our approach is inspired by this viewpoint. We conduct extensive evaluations of our method across different datasets. Experimental results demonstrate that our method not only enhances the quality of retrieval data but also improves MLLM performance on vision tasks. To summarize, our contributions are threefold: (1) We propose AutoRaC, an automated methodology for constructing retrieval data based on MLLM preferences, while supporting accurate relevance annotations. (2) Our method can fully explore the potential of the current knowledge base while supporting expansion to external knowledge sources, thereby increasing the diversity of data. (3) Experimental results verify that our constructed data enhances both the retriever's retrieval performance and the MLLM's capability on VQA tasks, achieving comparable quality to existing high-quality datasets.

2 Related Work

2.1 Multimodal Retrieval Augmented Generation

Multimodal retrieval augmented generation (Mei et al., 2025; Jeong et al., 2025) aims to enrich the non-parametric knowledge of MLLMs. For multimodal retrievers, existing approaches employ BM25 (Robertson and Zaragoza, 2009), DPR (Karpukhin et al., 2020), and fine-grained retrieval

(Lin et al., 2024) techniques for retrieval. REPLUG (Shi et al., 2024) enhances performance by treating the language model as a black box with a tunable retrieval module. PreFLMR (Lin et al., 2024) is a fine-grained multimodal retriever that employs token-level late interaction. Our method primarily focuses on improving multimodal retriever performance based on MLLM preferences.

2.2 Multimodal Retrieval Data Construction

Current multimodal retrieval datasets (Zhang et al., 2024; Deng et al., 2025) exhibit hierarchical and scenario-specific characteristics. Approaches like (Marino et al., 2019; Luo et al., 2021) generate search queries by combining image classifications or answers with textual questions, retrieving relevant articles via knowledge-search API as knowledge sources. SURF (Sun et al., 2024) reconstructs fine-tuning data through a liberalized framework to improve selective information utilization in multimodal retrieval-augmented generation. Muka (Deng et al., 2025) enhances visual information retrieval tasks by automatically pairing text with entity images to build multimodal knowledge bases, effectively addressing the limitations of text-only knowledge bases in cross-modal retrieval. There are also some knowledge-base-driven methods. InfoSeek (Chen et al., 2023) automatically converts knowledge triples into QA pairs using predefined question templates to construct retrieval data. Similarly, EVQA (Mensink et al., 2023) generates QA pairs automatically from Wikipedia with human verification. Approach like (Lin et al., 2023) employs pseudo-relevance-based strategies for document-label construction. While these methods produce high-quality multimodal retrieval data, there remains room for improvement in annotation accuracy and knowledge extensibility.

3 Methodology

In this work, we propose an automatic multimodal retrieval data construction method based on MLLM preferences, as shown in Figure 2. Specifically, we first filter parametric knowledge by verifying the correctness of the MLLM's initial response to the original query. We then obtain the retrieval distribution and proceed to the answer filtering module, subsequently deriving MLLM preferences distribution by concatenating the original query with retrieval results. Our objective is to minimize the distributional divergence between the retriever and the MLLM preferences, aligning them to introduce accurate and extensible relevance labels.

3.1 MLLM Preferences

3.1.1 Parameter Knowledge Filtering

To enhance the interpretability of document annotations and better enable MLLMs to distinguish between utilizing internal parametric knowledge and external knowledge, we designed a filtering module called **Parameter Knowledge Filtering**. Specifically, we split the original datasets into two subsets: $Q = \{Q^-, Q^+\}$, where Q^- and Q^+ respectively represent samples for which the base model generates incorrect and right responses using only its parametric knowledge. This stage enables more effective identification of potentially relevant documents and improves the interpretability of relevance labels.

3.1.2 Retrieval Distribution Acquisition

We then acquire the distribution of the multimodal retriever which takes image-text pair $q = \{I, T\}$ as input query and computes similarity scores against document tensors from an external knowledge base D. In our work, we employ a textual knowledge base following the PreFLMR (Lin et al., 2024) configuration, which incorporates fine-grained late interaction for precise retrieval. The similarity is formalized as follows:

$$sim(d, q) = \sum_{i=1}^{L_q} \max_{j=1}^{l_d} q_i d_j^T \quad (1)$$

where l_q and l_d denote the length dimensions of the query features q and document features d respectively. We then compute the retrieval distribution of retrieved document d:

$$P_R(d \mid q) = \frac{e^{sim(d,q)/\gamma}}{\sum_{d' \in D'} e^{sim(d',q)/\gamma}} \quad (2)$$

where $D' = \{d_1, d_2, ..., d_k\} \in D$ denotes the retrieved top-k relevant documents and γ is a hyperparameter controls the temperature of the softmax.

3.1.3 Answer Filtering

For each query $q' \in Q^-$, we retrieve top-k documents d with probability $P_R(d \mid q')$. We then proceed to the **Answer Filtering** module, where we select documents that contain correct answers from the retrieval results of query q' as relevant documents for the target query q'. This step is crucial for eliminating noise in the retrieval results and

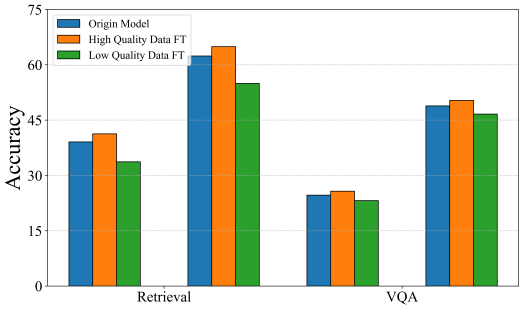

Figure 3: The figure demonstrates the performance variations of the retriever (on retrieval tasks) and the generator (on VQA tasks) when the retriever is fine-tuned with high-quality versus low-quality retrieval data.

focusing on high-quality candidates. At this stage, we improve the accuracy of relevant document labels and enable the retriever to more effectively comprehend the retrieved information.

3.1.4 MLLM Preferences Alignment

For MLLM preferences alignment, we first acquire the preferences distribution of the MLLM. The MLLM processes multimodal input $q' = \{I, T\}$ through an autoregressive generation mechanism. At each timestep t, the model predicts the probability distribution for the next token conditioned on the input q' and previously generated tokens:

$$\log P_M(y \mid d \circ q') = \sum_{t=1}^{n} \log P(y_t \mid y_{<t}, d \circ q') \quad (3)$$

To select high-quality documents that enhance response quality when incorporated into the model's knowledge, we identify candidates with higher relevance probabilities in MLLM preferences distribution which formally defined as:

$$P_M(d \mid q', y) = \frac{P_M(y \mid d \circ q') P_M(d \mid q')}{P_M(y \mid q')} \quad (4)$$

where y denotes ground-truth answer for input q'. However, directly computing $P_M(y \mid q')$ requires enumerating over entire knowledge base D, which is computationally intractable. To enable a feasible approximation, we instead utilize top-k document set $D' = \{d_1, d_2, ..., d_k\}$ retrieved by retriever R to approximate full knowledge base. We make following assumptions: 1) the prior probability $P_M(d \mid q')$ is uniform over retrieved documents, and 2) the contribution of unretrieved documents to generation of y is negligible. Under these assumptions, the marginalization in Equation (4) can be approximated by a sum over retrieved set D', and prior term $P_M(d \mid q')$ cancels out. Consequently, we obtain a normalized document preference distribution based on MLLM's generation probabilities:

$$\begin{aligned} P_M(d \mid q', y; D') &\approx \frac{e^{\log P_M(y \mid d \circ q')/\beta}}{\sum_{d' \in D'} e^{\log P_M(y \mid d' \circ q')/\beta}} \\ &= \frac{e^{\mathrm{Score}_M(d; q', y)/\beta}}{\sum_{d' \in D'} e^{\mathrm{Score}_M(d'; q', y)/\beta}} \end{aligned} \quad (5)$$

where $\mathrm{Score}_M(d; q', y) = \log P_M(y \mid d \circ q')$ quantifies the extent to which document d supports the generation of the correct answer y, and β is a temperature hyperparameter that controls the smoothness of the resulting distribution.

By concatenating each candidate document $d \in D'$ with the query q' and feeding the input $d \circ q'$ into the MLLM, we compute $\mathrm{Score}_M(d; q', y)$, thereby obtaining MLLM preference distribution defined in Equation (5). Our core objective is to align retrieval distribution of retriever R with MLLM preference distribution. Specifically, we minimize KL divergence between two distributions:

$$min\ KL(P_R(d \mid q') \mid\mid P_M(d \mid q', y; D')) \quad (6)$$

where $P_R(d \mid q')$ denotes the retriever distribution defined in Equation (2). Minimizing this KL divergence encourages the retriever to assign higher probabilities to documents that the MLLM deems more supportive of generating the correct answer y. In this manner, the retriever learns to emulate the MLLM's judgment of document relevance, thereby achieving preference alignment between retrieval and generation. Crucially, the relevance labels for documents are implicitly determined by the correctness of the MLLM's answers when augmented with the corresponding knowledge—i.e., grounded in the MLLM's post-retrieval generation performance. Our approach enables effective and efficient retriever fine-tuning while offering enhanced interpretability and accuracy.

Notably, our data construction approach is question-answer pair driven and independent of any specific knowledge base. Consequently, our method not only identifies more potentially relevant documents from the original database, but also enables migration of additional knowledge bases to relevant visual task datasets, thereby achieving knowledge base expansion and facilitating construction of higher-quality retrieval data. To better

understand the impact of retrieval data quality on model performance, we conducted experiments on two datasets, as illustrated in Figure 3. The results demonstrate high-quality retrieval data can positively influence both retriever and generator,

3.2 Training Strategy

We employ contrastive loss for fine-tuning the retriever. Slightly different from (Lin et al., 2023), our strategy is as follows: In the dataset $\mathcal{S}$, a randomly selected relevant document for query q is used as the positive sample, while negative samples are randomly chosen from the irrelevant document set $\mathcal{N}(q)$, which excludes any documents relevant to the query. This is because our method may match multiple relevant documents to each sample, and this strategy can prevent negative sampling failure. The d^+ and d^- denote the positive and negative sample:

$$\mathcal{L} = -\sum_{(q,d^+)\in\mathcal{S}} \log\left(\frac{e^{\mathrm{sim}(q,d^+)}}{e^{\mathrm{sim}(q,d^+)} + \sum\limits_{d^-\in\mathcal{N}(q)} e^{\mathrm{sim}(q,d^-)}}\right) \quad (7)$$

4 Experiments

4.1 Datasets and Metrics

Datasets. We employ InfoSeek and EVQA as the visual task evaluation datasets. Both datasets require incorporating external knowledge for visual question answering, demanding models to possess capabilities in visual understanding, knowledge retrieval, and reasoning. We follow the same experimental setup as (Lin et al., 2024).

Knowledge base. To ensure fair comparison, we conduct experiments using the Wiki knowledge base (Lin et al., 2024). This database collects all Wikipedia passages about common objects and concepts. Each sample in the VQA dataset corresponds to at least one relevant document.

Metrics. Following the setup in (Lin et al., 2024), we evaluate the retriever using Pseudo-Recall@K (PR@K) and Recall@K (R@K). PR@K measures whether the retriever can find relevant documents containing any candidate answer to the question within the top-k retrieved documents. R@K requires retrieved documents to exactly match the original ground truth to be counted as relevant. For generator evaluation, we follow each dataset's original evaluation protocol: The InfoSeek employs diverse metrics based on question types - exact match for string answers, while allowing tolerance ranges for numerical/temporal answers. The InfoSeek evaluation provides three metrics: performance on new questions, performance on new entities, and a combined final score. EVQA use BEM (Bulian et al., 2022) to evaluate answers, which uses a BERT model fine-tuned to determine equivalent answers for a given question. We report 5-run average accuracy of the generator.

4.2 Implementation Details

To validate our method, we conduct experiments on the state-of-the-art multimodal retriever PreFLMR. During the retrieval data construction phase, we employ LLaVA-1.5-7B (Liu et al., 2023) to build relevant documents for 5k data samples in each dataset with BEM as judge. For the answer generators, we report results using LLaVA-1.5-7B. Detailed configurations are provided in A.1.

4.3 Results

To evaluate AutoRaC, we first conduct performance tests on both the retriever and generator using its constructed data. Then we conduct ablation experiments on filtering modules, followed by data analysis, and finally verify the extensibility with an external knowledge base.

4.3.1 Results on Documents Retrieval

We first compare the retriever fine-tuned using the dataset constructed by AutoRaC against other baselines, with results shown in Table 1. The construction methods for different datasets are as follows (corresponding to the numbering in the table): (5) Origin: The retriever is fine-tuned using the complete original dataset. (6) Random: Relevant documents for samples are randomly selected from the knowledge base. (7) Origin*: Fine-tuning is performed using samples from the original dataset that correspond to our dataset. (8) Ours: The retriever is fine-tuned with the dataset constructed via AutoRaC. (9) REPLUG: We adapted the text-based REPLUG to the multimodal domain as our baseline, simulating the method by removing the two-stage filtering module. For fair comparison, Random, Origin*, and Ours maintain identical Q-A pairs across samples, differing only in their relevant document annotations. The experimental results demonstrate that fine-tuning the retriever using the complete original dataset yields significant performance improvements. Conversely, when employing randomly selected relevant documents for fine-tuning, the retriever performance shows sub-

No.	Retriever	Method	Infoseek		EVQA	
			PR@5	R@5	PR@5	R@5
(1)	CLIP	✗	17.10	–	10.40	–
(2)	FLMR	✗	47.10	–	–	–
(3)	Google Lens	✗	–	–	62.50	–
Comparative Experiments						
(4)	PreFLMR	✗	57.43	39.10	72.00	62.40
(5)	PreFLMR	Origin	61.05	41.29	73.97	64.93
(6)	PreFLMR	Random	53.46	33.71	67.09	54.96
(7)	PreFLMR	Origin*	**58.24**	**39.70**	72.43	**62.83**
(8)	PreFLMR	AutoRaC(ours)	58.09	39.53	**72.51**	62.80
(9)	PreFLMR	REPLUG	57.56	38.95	72.00	62.27

Table 1: Retrieval performance of PR@5 and R@5 on InfoSeek and E-VQA datasets. Baseline results for CLIP, FLMR, and Google Lens are sourced from existing literature.

stantial degradation. These findings indicate that high-quality, sufficient retrieval data positively impacts retriever performance, while low-quality data exerts negative effects. Furthermore, fine-tuning with the dataset constructed by AutoRaC leads to improved retriever performance. While showing a marginal gap compared to the Origin* on InfoSeek, it achieves comparable or even superior results on EVQA when using the same quantity of high-quality data (e.g., from 72.00 to 72.51). These results demonstrate our method can generate high-quality retrieval data to positively influence the retriever, validating the effectiveness of AutoRaC.

4.3.2 Results on Answer Generation

Since the ultimate objective of retrieval is to augment generation, we subsequently evaluate the performance of different datasets on VQA task. The results are presented in Table 2. Here, Oracle represents the theoretical upper bound for RAG, where for each sample we concatenate one randomly selected ground-truth relevant document with the input for answer generation. The experimental results demonstrate that incorporating retrieval results leads to higher answer accuracy. The Oracle performance significantly surpasses all other configurations. These findings confirm the effectiveness of retrieval-augmented generation using Wiki as the external knowledge base on both datasets, while indicating substantial room for improvement in retriever performance. Furthermore, fine-tuning the retriever with Origin and Random datasets respectively yields considerable performance gains and declines in generator performance, demonstrating that high-quality datasets positively impact the generator's final VQA task performance, whereas low-quality data has detrimental effects. Finally, the experimental results show that fine-tuning with the AutoRaC-constructed dataset improves the generator's VQA performance, achieving comparable or even superior results to high-quality datasets of the same size (e.g., 49.33 vs 49.45). This demonstrates that our method can produce high-quality retrieval data that simultaneously enhances both retriever and generator, thereby further validating the effectiveness of AutoRaC.

Moreover, we additionally evaluate our approach using different MLLMs, including Qwen2.5-VL-7B-Instruct (Bai et al., 2025) and GPT-5.1 (OpenAI, 2025), which are capable of understanding and reasoning over both images and text and distinct from the one used during data construction. The results consistently demonstrate the effectiveness and superiority of our method.

4.3.3 Ablation Study

We further conducted ablation studies on the two filtering modules, as shown in Table 3. Results show that using both modules together achieves the best performance for both retriever and generator, while removing either module degrades performance. Interestingly, on InfoSeek, removing both modules unexpectedly outperformed removing only one, although single-module removal still caused performance drops, indicating a synergistic effect between the two modules. Moreover, removing the F-PK module caused only minor degradation on EVQA's VQA task, while our method

No.	Retriever	MLLM	Method	KB	Infoseek			EVQA
					Unseen-Q	Unseen-E	Final	Final
Without Retrieval								
(1)	✗	LLaVA	✗	✗	15.69	10.57	12.63	20.80
(2)	✗	LLaVA	Oracle	Wiki	48.79	50.53	49.64	66.91
(3)	✗	Qwen	✗	✗	27.91	30.70	29.24	24.53
(4)	✗	GPT	✗	✗	16.96	20.26	18.46	33.39
With Retrieval								
(5)	PreFLMR	LLaVA	✗	Wiki	24.99	24.32	24.65	48.87
(6)	PreFLMR	LLaVA	Origin	Wiki	26.07	25.38	25.72	50.34
(7)	PreFLMR	LLaVA	Random	Wiki	23.57	22.79	23.17	46.64
(8)	PreFLMR	LLaVA	Origin*	Wiki	**25.80**	24.33	25.04	49.33
(9)	PreFLMR	LLaVA	AutoRaC(ours)	Wiki	25.10	**25.07**	**25.08**	**49.45**
(10)	PreFLMR	LLaVA	REPLUG	Wiki	24.84	24.53	24.68	49.18
(11)	PreFLMR	Qwen	✗	Wiki	38.52	37.78	38.15	52.76
(12)	PreFLMR	Qwen	Origin*	Wiki	38.64	**38.12**	**38.38**	52.85
(13)	PreFLMR	Qwen	AutoRaC(ours)	Wiki	**38.90**	37.82	38.35	**53.01**
(14)	PreFLMR	GPT	✗	Wiki	19.61	23.22	21.26	57.19
(15)	PreFLMR	GPT	Origin*	Wiki	**21.22**	22.08	21.64	57.25
(16)	PreFLMR	GPT	AutoRaC(ours)	Wiki	19.61	**24.42**	**21.75**	**57.44**

Table 2: Results of the generator on VQA tasks. The best results are highlighted in bold.

Method	Retrieval				VQA			
	Infoseek		EVQA		Infoseek			EVQA
	PR@5	R@5	PR@5	R@5	Unseen-Q	Unseen-E	Final	Final
Ours	**58.09**	**39.53**	**72.51**	**62.80**	**25.10**	**25.07**	**25.08**	**49.45**
w/o F-PK	57.35	38.85	72.32	62.59	24.91	25.02	24.96	49.44
w/o F-Ans	57.56	39.12	72.29	62.53	25.07	24.47	24.76	49.25
w/o F-PK, F-Ans	57.56*	38.95*	72.00	62.27	24.84	24.53*	24.68	49.18

Table 3: Results of both the retriever and generator after removing two filter modules during data construction. F-PK and F-Ans respectively represent Parameter Knowledge Filtering and Answer Filtering module.

significantly reduced dataset size, effectively filtering irrelevant documents and improving retriever fine-tuning efficiency.

4.3.4 Data Analysis

We first analyze the data volume of datasets constructed through different methods, as shown in Table 4. Here, we maintain identical sample sizes for Random, Origin*, and AutoRaC to ensure fair comparison. The dataset constructed with both modules achieves a 30%-50% reduction in data size. This demonstrates that AutoRaC can effectively filter irrelevant documents to maintain fine-tuning quality, while simultaneously improving retriever fine-tuning efficiency.

To further validate capability of current retriever in ranking relevant documents, we analyzed the retrieval results obtained from original PreFLMR model during data construction, with the accuracy distribution shown in Figure 4. Here, the x-axis represents the ranked positions of retrieved documents in the training set, while the y-axis indicates the count of correctly answered queries when concatenated with each document. The ideal distribution, corresponding to the relevance probability $P_R(d' \mid q')$, should exhibit a decaying trend as the rank position increases. However, the actual results demonstrate relatively flat curves that gradually approach a uniform distribution at higher ranks. These findings indicate substantial room

Dataset\Method	Random	Origin*	AutoRaC	w/o F-PK	w/o f-ans	w/o F-PK, f-ans
Infoseek	2248	2248	2248	2585	2782	3175
EVQA	1852	1852	1852	2715	2835	3942

Table 4: Statistics of data volume under different data construction settings

No.	Retriever	Fine-tuning data of retriever				OKVQA		
		Wiki	GS	Random	AutoRaC	PR@1	PR@3	PR@5
Baselines: With original knowledge base								
(1)	FLMR (Lin et al., 2023)	✗	✗	✗	✗	-	-	68.1
(2)	PreFLMR (Lin et al., 2024)	✗	✗	✗	✗	-	-	68.6
(3)	PreFLMR	✓	✗	✗	✗	-	-	70.9
Comparison: With additional knowledge base								
(4)	PreFLMR	✓	✓	✗	✗	55.15	76.65	83.85
(5)	PreFLMR	✓	✓	✓	✗	50.14	72.26	80.30
(6)	PreFLMR (ours)	✓	✓	✗	✓	**56.16**	**77.57**	**84.50**

Table 5: The performance changes of multimodal retriever after adding an additional knowledge base.

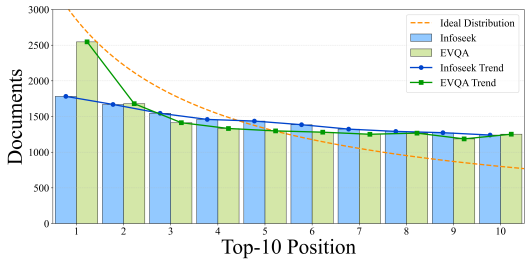

Figure 4: This figure presents the actual distribution of relevant documents in retrieval results and their ideal distribution. The blue and green bars represent the distributions of relevant documents for the InfoSeek and EVQA datasets respectively, while the orange curve indicates the ideal distribution of relevant documents.

for improvement in the retriever's performance on the knowledge base, while simultaneously demonstrating that our proposed method can effectively exploit the retriever's inherent potential and establish positive feedback for retrieval optimization.

4.3.5 Knowledge Base Expansion

Our previous experiments have demonstrated effectiveness of AutoRaC and its theoretical applicability for knowledge base expansion through retrieved data. To verify whether our method can genuinely extend retrieval capabilities by incorporating new knowledge bases, we introduced Google Search Corpus (GS) (Luo et al., 2021) and conducted tests on the OKVQA (Marino et al., 2019). Following (Lin et al., 2023), we employed pseudo-recall rate PR@K as the evaluation metric.

As shown in Table 5, we have some findings on what works: (1) Incorporating additional external knowledge bases alongside the original one and performing fine-tuning leads to significant retrieval improvement, demonstrating effectiveness of knowledge expansion. (2) When conducting additional data construction on the merged database, the Random method introduces irrelevant documents that misguide the retriever, causing performance degradation. (3) Compared to random construction, AutoRaC, which is grounded in multimodal queries and outcome-oriented design, can generate higher-quality data. (i.e., from 83.85 to 84.50).

5 Conclusions

Current multimodal retrieval data construction methods struggle to balance annotation accuracy and diversity. To address this issue, we propose AutoRaC, an automated retrieval data construction framework based on MLLM preferences with two effective filtering modules. Experiments show that: (1) AutoRaC generates accurate and interpretable relevance annotations for high-quality multimodal retrieval data; (2) It enhances knowledge base utilization while supporting additional knowledge integration for more diverse data construction; (3) Retrievers trained on AutoRaC data achieve performance comparable to existing high-quality datasets on both retrieval and VQA tasks. We hope this work provides new insights into multimodal retrieval data construction.

6 Limitations

Limited by available computational resources, we leave further investigations as future work: (1) This method can be further validated on a wider range of datasets and models. (2) By increasing the retrieval volume, more high-quality data can be obtained to achieve better performance. (3) Although the dual-filtering module in our method has demonstrated clear benefits, the ablation studies still exhibit a small number of outliers. We identify this as a direction for future investigation. (4) The existing evaluation protocols for multimodal retrievers exhibit inconsistencies with those used in VQA evaluation. Based on this paper, we propose an improved method. However, since this is not related to the main content of this article, we have included it in the appendix as a direction for future work.

7 Potential Risks and Ethical Considerations

Our method relies on MLLM preferences to construct retrieval data, which may inadvertently propagate biases present in the underlying model (e.g., cultural, gender, or factual biases). If deployed without careful validation, AutoRaC could amplify these biases in downstream RAG systems. Additionally, while our approach reduces reliance on human annotation, it does not eliminate the need for quality control—automatically constructed data may still contain inaccuracies or hallucinated knowledge. We recommend human-in-the-loop validation for high-stakes applications.

We use AI for minor language polishing. All scientific content is written by human researchers.

References

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, and 8 others. 2025. Qwen2.5-vl technical report. *Preprint*, arXiv:2502.13923.

Jannis Bulian, Christian Buck, Wojciech Gajewski, Benjamin Börschinger, and Tal Schuster. 2022. Tomayto, tomahto. beyond token-level answer equivalence for question answering evaluation. *arXiv preprint arXiv:2202.07654*.

Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. 2021. Autoregressive entity retrieval. *Preprint*, arXiv:2010.00904.

Yang Chen, Hexiang Hu, Yi Luan, Haitian Sun, Soravit Changpinyo, Alan Ritter, and Ming-Wei Chang. 2023. Can pre-trained vision and language models answer visual information-seeking questions? *arXiv preprint arXiv:2302.11713*.

Lianghao Deng, Yuchong Sun, Shizhe Chen, Ning Yang, Yunfeng Wang, and Ruihua Song. 2025. MuKA: Multimodal knowledge augmented visual information-seeking. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 9675–9686, Abu Dhabi, UAE. Association for Computational Linguistics.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, and 1 others. 2022. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.

Hexiang Hu, Yi Luan, Yang Chen, Urvashi Khandelwal, Mandar Joshi, Kenton Lee, Kristina Toutanova, and Ming-Wei Chang. 2023. Open-domain visual entity recognition: Towards recognizing millions of wikipedia entities. *Preprint*, arXiv:2302.11154.

Soyeong Jeong, Kangsan Kim, Jinheon Baek, and Sung Ju Hwang. 2025. Videorag: Retrieval-augmented generation over video corpus. *Preprint*, arXiv:2501.05874.

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.

Fanheng Kong, Jingyuan Zhang, Yahui Liu, Hongzhi Zhang, Shi Feng, Xiaocui Yang, Daling Wang, Yu Tian, Victoria W., Fuzheng Zhang, and Guorui Zhou. 2025. Modality curation: Building universal embeddings for advanced multimodal information retrieval. *Preprint*, arXiv:2505.19650.

Weizhe Lin, Jinghong Chen, Jingbiao Mei, Alexandru Coca, and Bill Byrne. 2023. Fine-grained late-interaction multi-modal retrieval for retrieval augmented visual question answering. *Advances in Neural Information Processing Systems*, 36:22820–22840.

Weizhe Lin, Jingbiao Mei, Jinghong Chen, and Bill Byrne. 2024. Preflmr: Scaling up fine-grained late-interaction multi-modal retrievers. *arXiv preprint arXiv:2402.08327*.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916.

Man Luo, Yankai Zeng, Pratyay Banerjee, and Chitta Baral. 2021. Weakly-supervised visual-retriever-reader for knowledge-based question answering. *arXiv preprint arXiv:2109.04014*.

Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. 2019. Ok-vqa: A visual question answering benchmark requiring external knowledge. In *Proceedings of the IEEE/cvf conference on computer vision and pattern recognition*, pages 3195–3204.

Lang Mei, Siyu Mo, Zhihan Yang, and Chong Chen. 2025. A survey of multimodal retrieval-augmented generation. *Preprint*, arXiv:2504.08748.

Dheeraj Mekala, Tu Vu, Timo Schick, and Jingbo Shang. 2022. Leveraging qa datasets to improve generative data augmentation. *Preprint*, arXiv:2205.12604.

Thomas Mensink, Jasper Uijlings, Lluis Castrejon, Arushi Goel, Felipe Cadar, Howard Zhou, Fei Sha, André Araujo, and Vittorio Ferrari. 2023. Encyclopedic vqa: Visual questions about detailed properties of fine-grained categories. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3113–3124.

Arvind Neelakantan, Tao Xu, Raul Puri, Alec Radford, Jesse Michael Han, Jerry Tworek, Qiming Yuan, Nikolas Tezak, Jong Wook Kim, Chris Hallacy, Johannes Heidecke, Pranav Shyam, Boris Power, Tyna Eloundou Nekoul, Girish Sastry, Gretchen Krueger, David Schnurr, Felipe Petroski Such, Kenny Hsu, and 6 others. 2022. Text and code embeddings by contrastive pre-training. *Preprint*, arXiv:2201.10005.

OpenAI. 2025. Gpt-5.1. `https://platform.openai.com/docs/models/gpt-5.1`. Accessed: 2026-04-26.

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

Dustin Schwenk, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh Mottaghi. 2022. A-okvqa: A benchmark for visual question answering using world knowledge. *Preprint*, arXiv:2206.01718.

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Richard James, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. 2024. REPLUG: Retrieval-augmented black-box language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8371–8384, Mexico City, Mexico. Association for Computational Linguistics.

Jiashuo Sun, Jihai Zhang, Yucheng Zhou, Zhaochen Su, Xiaoye Qu, and Yu Cheng. 2024. SURf: Teaching large vision-language models to selectively utilize retrieved information. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7611–7629, Miami, Florida, USA. Association for Computational Linguistics.

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul Bennett, Junaid Ahmed, and Arnold Overwijk. 2020. Approximate nearest neighbor negative contrastive learning for dense text retrieval. *Preprint*, arXiv:2007.00808.

Tianjun Zhang, Shishir G Patil, Naman Jain, Sheng Shen, Matei Zaharia, Ion Stoica, and Joseph E Gonzalez. 2024. Raft: Adapting language model to domain specific rag. In *First Conference on Language Modeling*.

Junjie Zhou, Zheng Liu, Ze Liu, Shitao Xiao, Yueze Wang, Bo Zhao, Chen Jason Zhang, Defu Lian, and Yongping Xiong. 2024. Megapairs: Massive data synthesis for universal multimodal retrieval. *Preprint*, arXiv:2412.14475.

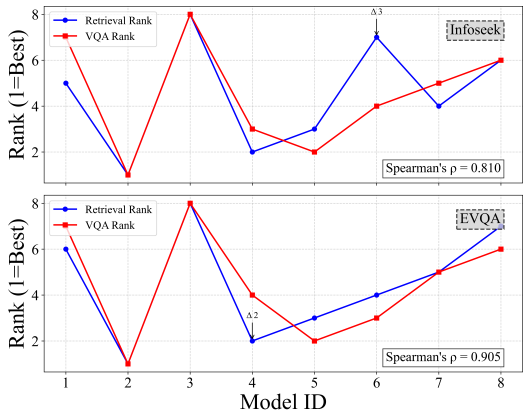

Figure 5: Comparison of ranking similarity between retrieval results and VQA performance across different models on both InfoSeek and EVQA datasets.

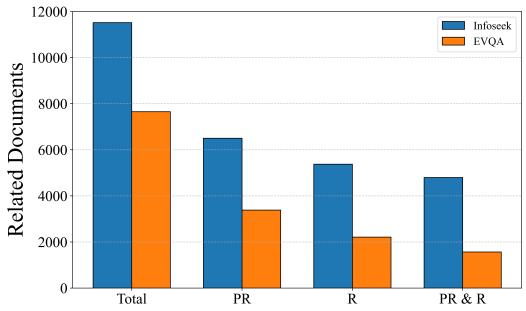

Figure 6: Statistical analysis of relevant documents in the AutoRaC-constructed dataset meeting evaluation criteria for both Pseudo-Recall (PR) and Recall (R). PR & R denotes documents satisfying both metrics simultaneously.

A Appendix

A.1 Implementation Details

This section is a supplement to the experimental details of the paper. All experiments were conducted a single NVIDIA A40 GPU.

A.1.1 Retriever

For all constructed datasets, we report the optimal results of PreFLMR after fine-tuning for 5 epochs on the training split. During fine-tuning, we freeze the visual encoder module while setting the learning rate to 1e-5 for other trainable parameters, which are optimized using the Adam optimizer. All experiments were conducted with a batch size of 8 and a warmup step of 100. The accumulate grad batch set to 8.

A.1.2 MLLM

We apply Low-Rank Adaptation (Hu et al., 2022) to reduce trainable parameters, setting the LoRA rank to 128 and the LoRA alpha to 256. The total batch size is 512. We use the Adam optimizer for fine-tuning, making only the parameters of the multimodal projectors and the LoRA modules trainable, with learning rates respectively set to 2e-5 and 2e-4.

A.2 Discussions

The experimental results presented earlier reveal partial inconsistencies between retrieval metrics and VQA performance. We first conducted a ranking similarity comparison between retrieval results and VQA results, as shown in Figure 5. The results indicate that while their Spearman correlation coefficients reach 0.8-0.9, significant ranking discrepancies persist. This suggests inherent uncertainty in evaluation metrics. Given the higher reliability of VQA assessment methods (exact match, similarity match, etc.), we attribute this primarily to limitations in retriever evaluation. Although pseudo-recall (PR) and ground truth (R) can partially reflect multimodal retrieval performance, the observed inconsistencies demonstrate they may not accurately represent the retriever's ultimate capability, since: (1) judging document relevance solely by answer inclusion may yield false assessments, and (2) relevant information might reside in undiscovered documents. We subsequently analyzed how many AutoRaC-constructed documents actually meet PR/R evaluation criteria (Figure 6), finding only 25%-50% simultaneously or individually satisfy these standards. This confirms substantial room for improvement in current retriever evaluation metrics. Accordingly, we propose preliminary improvements to the evaluation framework based on AutoRaC. Pseudo code can be found in A.4.

A.3 Instruction templates

The MLLM instruction templates designed for different retrieval quantities of documents is shown in the Figure 7.

A.4 Evaluation Improvement

In this section, we propose a novel retrieval evaluation framework called Hybrid Recall (HRecall) based on AutoRaC. The pseudocode is shown as Algorithm 1:

A.5 GPT vs. Qwen Performance Analysis

Our experimental results indicate that the fine-tuned Qwen-VL-7B model achieves superior per-

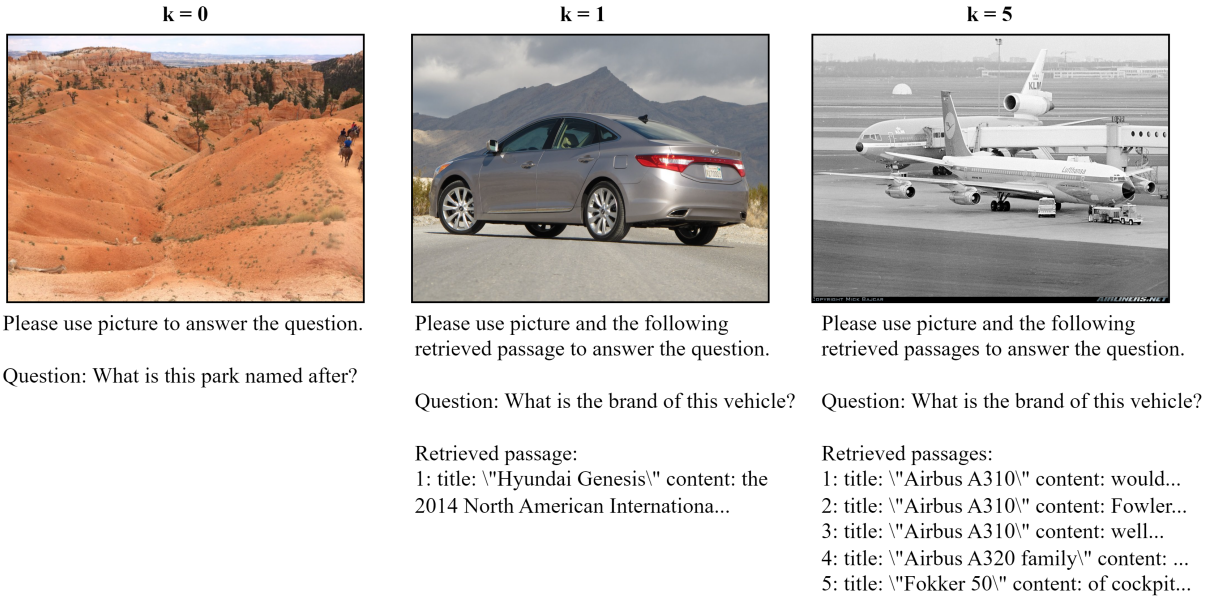

Figure 7: Display of different instruction templates.

formance on the Infoseek benchmark compared to the GPT-5.1 model under direct inference, despite the latter's larger parameter count. We attribute this counter-intuitive result to the task-specific alignment induced by Supervised Fine-Tuning (SFT). While GPT-5.1 is optimized for general-purpose dialogue and tends to generate verbose natural language explanations, the Qwen model, fine-tuned on the Infoseek dataset, learns to prioritize concise, extractive answers that strictly adhere to the evaluation metric (Exact Match). This aligns with the findings of Chen et al. (Chen et al., 2023), which highlight that pre-trained MLLMs often require fine-tuning to effectively elicit fine-grained knowledge for visual information-seeking questions.

This observation resonates with the analysis in the original Infoseek paper. Chen et al. (Chen et al., 2023) demonstrated that even state-of-the-art pre-trained models like PaLI-X struggle with zero-shot performance on Infoseek (10.8% on Human split) and require fine-tuning to activate their latent knowledge. Similarly, our results suggest that GPT-5.1, despite its strong reasoning capabilities, behaves similarly to the 'zero-shot' or 'Q-only' baselines in the Infoseek paper, failing to adapt to the constrained output format required by the benchmark without extensive prompt engineering.

Furthermore, we observed that even with strict prompt constraints, GPT-5.1 frequently fails to suppress its inherent tendency toward elaborative responses. In contrast, the fine-tuned Qwen model exhibits a higher degree of output determinism. This discrepancy highlights a fundamental trade-off: general-purpose MLLMs prioritize helpfulness and verbosity, whereas benchmark-oriented models optimize for precision and brevity. The Relaxed Accuracy metric in Infoseek penalizes extraneous tokens, putting models favoring natural language generation at a systematic disadvantage.

A.6 Dataset Details

In this work, the datasets are distributed under the Apache License 2.0, which permits its use for research purposes. We have adhered to all terms and conditions specified by the license. The data volume statistics are as follows:

Dataset	Train	Dev	Test
Infoseek	676.0k	-	4.7k
EVQA	167.0k	9.9k	3.8k

Table 6: Dataset Statistics

Algorithm 1 Evaluation of Multimodal Retriever (Hybrid Recall)

Require:

$\mathcal{D}$: Test dataset $\{(I_i, Q_i, A_i, K_i)\}_{i=1}^{N}$ ▷ Containing image, question, ground-truth answers and ducuments.

$\mathcal{K}$: Knowledge base

$MLLM$: Multimodal Large Language Model

R: Multimodal retriever

k: Top-k retrieval count

Initialize:

$C \leftarrow \emptyset$ ▷ Indices of correct initial responses

$W \leftarrow \emptyset$ ▷ Indices of incorrect initial responses

$Count \leftarrow 0$ ▷ Total number of correct samples

for $i = 1$ **to** N **do**

  $a_{first} \leftarrow MLLM(I_i, Q_i)$

  if $a_{first}\ in\ A_i$ **then**

    $C \leftarrow C \cup \{i\}$

  else

    $W \leftarrow W \cup \{i\}$

  end if

end for

for $i \in C$ **do**

  $\mathcal{R}_{\text{top-k}} \leftarrow R(I_i, Q_i, \mathcal{K}, k)$

  $isCorrect \leftarrow$ False

  for $j = 1$ **to** k **do**

    if $d_j \in \mathcal{K}_i$ **then**

      isCorrect $\leftarrow$ True

      break

    end if

  end for

  if $isCorrect$ **then**

    $count \leftarrow count + 1$

  end if

end for

for $i \in W$ **do**

  $\mathcal{R}_{\text{top-k}} \leftarrow R(I_i, Q_i, \mathcal{K}, k)$

  $isCorrect \leftarrow$ false

  for $j = 1$ **to** k **do**

    $a_{rag} \leftarrow MLLM(I_i, Q_i, \mathcal{R}_j)$

    if $a_{rag}\ in\ A_i$ **then**

      $isCorrect \leftarrow$ true

      break

    end if

  end for

  if $isCorrect$ **then**

    $Count \leftarrow Count + 1$

  end if

end for

$HRecall \leftarrow \frac{Count}{|\mathcal{D}|}$

return $HRecall$

Reproducibility Checklist

1. General Paper Structure

1.1. Includes a conceptual outline and/or pseudocode description of AI methods introduced (yes/partial/no/NA) yes

1.2. Clearly delineates statements that are opinions, hypothesis, and speculation from objective facts and results (yes/no) yes

1.3. Provides well-marked pedagogical references for less-familiar readers to gain background necessary to replicate the paper (yes/no) yes

2. Theoretical Contributions

2.1. Does this paper make theoretical contributions? (yes/no) no

If yes, please address the following points:

2.2. All assumptions and restrictions are stated clearly and formally (yes/partial/no) Type your response here

2.3. All novel claims are stated formally (e.g., in theorem statements) (yes/partial/no) Type your response here

2.4. Proofs of all novel claims are included (yes/partial/no) Type your response here

2.5. Proof sketches or intuitions are given for complex and/or novel results (yes/partial/no) Type your response here

2.6. Appropriate citations to theoretical tools used are given (yes/partial/no) Type your response here

2.7. All theoretical claims are demonstrated empirically to hold (yes/partial/no/NA) Type your response here

2.8. All experimental code used to eliminate or disprove claims is included (yes/no/NA) Type your response here

3. Dataset Usage

3.1. Does this paper rely on one or more datasets? (yes/no) yes

If yes, please address the following points:

3.2. A motivation is given for why the experiments are conducted on the selected datasets (yes/partial/no/NA) yes

3.3. All novel datasets introduced in this paper are included in a data appendix (yes/partial/no/NA) yes

3.4. All novel datasets introduced in this paper will be made publicly available upon publication of the paper with a license that allows free usage for research purposes (yes/partial/no/NA) yes

3.5. All datasets drawn from the existing literature (potentially including authors' own previously published work) are accompanied by appropriate citations (yes/no/NA) yes

3.6. All datasets drawn from the existing literature (potentially including authors' own previously published work) are publicly available (yes/partial/no/NA) yes

3.7. All datasets that are not publicly available are described in detail, with explanation why publicly available alternatives are not scientifically satisficing (yes/partial/no/NA) NA

4. Computational Experiments

4.1. Does this paper include computational experiments? (yes/no) yes

If yes, please address the following points:

4.2. This paper states the number and range of values tried per (hyper-) parameter during development of the paper, along with the criterion used for selecting the final parameter setting (yes/partial/no/NA) partial

4.3. Any code required for pre-processing data is included in the appendix (yes/partial/no) yes

4.4. All source code required for conducting and analyzing the experiments is included in a code appendix (yes/partial/no) yes

4.5. All source code required for conducting and analyzing the experiments will be made publicly available upon publi-

cation of the paper with a license that allows free usage for research purposes (yes/partial/no) yes

4.6. All source code implementing new methods have comments detailing the implementation, with references to the paper where each step comes from (yes/partial/no) yes

4.7. If an algorithm depends on randomness, then the method used for setting seeds is described in a way sufficient to allow replication of results (yes/partial/no/NA) NA

4.8. This paper specifies the computing infrastructure used for running experiments (hardware and software), including GPU/CPU models; amount of memory; operating system; names and versions of relevant software libraries and frameworks (yes/partial/no) partial

4.9. This paper formally describes evaluation metrics used and explains the motivation for choosing these metrics (yes/partial/no) yes

4.10. This paper states the number of algorithm runs used to compute each reported result (yes/no) yes

4.11. Analysis of experiments goes beyond single-dimensional summaries of performance (e.g., average; median) to include measures of variation, confidence, or other distributional information (yes/no) no

4.12. The significance of any improvement or decrease in performance is judged using appropriate statistical tests (e.g., Wilcoxon signed-rank) (yes/partial/no) no

4.13. This paper lists all final (hyper-)parameters used for each model/algorithm in the paper's experiments (yes/partial/no/NA) partial