

An Optimization Framework for Retrieval Augmented Generation in Indonesian Educational Question Answering

I Ketut Resika Arthana¹, Nyoman Gunantara², Made Sudarma³, I Made Sukarsa⁴

¹Dept. of Informatics Universitas Pendidikan Ganesha, Indonesia

^{2,3,4}Faculty of Engineering, Universitas Udayana, Indonesia

Email: resika@undiksha.ac.id

Article Info

Article history:

Received June 05, 2026

Revised June 14, 2026

Accepted July 16, 2026

Keywords:

RAG

LLM

Chunking

Embedding

RAGAS

ABSTRACT

The Retrieval-Augmented Generation (RAG) approach has been widely adopted to produce responses that are more closely aligned with a predefined knowledge context. However, many RAG implementations have not undergone systematic optimization of their retrieval and generation components, resulting in outputs that do not always correspond accurately to the reference context. This study developed a RAG optimization framework for Indonesian-language educational question answering using a Human-Computer Interaction learning corpus as a case study. In the retrieval stage, the study evaluated chunking strategies, multilingual embedding models, and the use of a reranker. Evaluation was conducted using Mean Reciprocal Rank (MRR), Normalized Discounted Cumulative Gain (nDCG@K), and Hit@K. In the generation stage, candidate Large Language Models (LLMs) were assessed using RAGAS metrics, namely Context Precision (CP), Context Recall (CR), Faithfulness (F), Answer Relevancy (AR), and Answer Correctness (AC). Experimental results showed that the GTE configuration with fixed-size chunking and a reranker yielded the best retrieval performance, achieving an MRR of 0.9082, nDCG@5 of 0.9215, and Hit@5 of 0.9655. In the generation stage, Gemma 4 E4B exhibited the most balanced answer quality. The resulting framework provides a procedure for selecting retrieval and generation settings for a given corpus.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

I Ketut Resika Arthana,

Dept. of Informatics

Universitas Pendidikan Ganesha, Indonesia

Email: resika@undiksha.ac.id

1. INTRODUCTION

In educational settings, Question Answering systems must be grounded in the learning materials used in the classroom. The responses generated should correspond to the instructional content and be traceable to reference documents, as answers that appear generally correct may not necessarily reflect the specific material being studied. LLMs offer the potential to build more interactive Question Answering systems, given their capacity to interpret queries and formulate responses in natural language. However, the direct use of LLMs does not guarantee alignment between generated answers and specific learning materials, since the model's knowledge derives from a broad, general-purpose training corpus. Issues such as factual hallucination, knowledge recency, and limited domain proficiency remain persistent challenges in LLM deployment [1]. The risk of hallucination is further associated with factual inaccuracies and output inconsistencies relative to the knowledge source [2], [3]. In educational Question Answering systems, this concern requires evaluation, as a linguistically well-formed response does not necessarily have a basis in the learning materials being used.

These limitations require that LLM-based question answering systems incorporate a mechanism capable of grounding responses in reference documents. RAG addresses this by retrieving relevant information from an external corpus prior to response generation, enabling the model to draw upon external knowledge at inference time [4]. Through this mechanism, responses are guided by context retrieved from the learning materials rather than solely the LLM's internal knowledge. RAG has been considered particularly relevant for online learning [5], educational chatbots [6], and question answering, personalized learning, and tutoring systems [7]. In the higher education context, various RAG configurations have been evaluated within college chatbot settings to examine retrieval configuration effects on response quality [8], though explicit optimization techniques and challenges related to retrieval accuracy, reasoning, and system integration remain underexplored.

The quality of a RAG system depends on the processes that precede response generation, including chunking, embedding, retrieval, reranking, and generation, so evaluation should account for the role of each stage [9]. If the configuration does not match the document characteristics and task requirements, errors may arise from unretrieved relevant context, poor ranking, noisy context, or the inability of the LLM to use the retrieved evidence. Even in RAG settings, LLMs may still struggle with noisy documents, unsupported questions, multi document information integration, and incorrect external information [1]. Therefore, retrieval and generation need to be evaluated separately: retrieval can be assessed using MRR, nDCG@5, and Hit@5, while generation can be evaluated through context precision, context recall, faithfulness, answer relevancy, and answer correctness [5]. In the retrieval stage, chunking and embedding should be tested together because their combination affects retrieval quality, especially in systems using open source or locally hosted LLMs [10]. Chunking defines the information units retrieved by the system, whereas embedding determines how semantic similarity between queries and document content is represented. Retrieval granularity requires a balance between contextual completeness and informational precision [11], and benchmarking studies show that segmentation strategies may perform differently across corpus types and retrieval tasks [12]. This issue is more pronounced in learning materials, which contain topics, subtopics, definitions, examples, and procedures. For complex documents, structure aware chunking can preserve information relationships during retrieval [13], while structure based chunking helps maintain semantic units in educational documents [14]. The choice of embedding model and its consistent use during indexing and querying also influence retrieval precision in RAG systems [15].

The Indonesian language context requires specific evaluation because linguistic characteristics, terminological variation, and document structure can affect chunking, embedding, retrieval, and generation. Prior work shows that chunking strategies and top-K settings influence explicit retrieval performance in Indonesian-language scientific articles, with different methods showing different strengths across exact match, semantic similarity, and textual structure measures [16]. Therefore, RAG configurations should be tested against the target language and document type. Generator selection also needs evaluation because answer quality depends on the model's ability to interpret queries, use retrieved context, and maintain fidelity to the source. Local LLMs such as Gemma, LLaMA, and Phi are suitable for local deployment, although this study does not assess operational costs, data security, or infrastructure governance. Since small language models combined with RAG have been proposed for resource constrained educational settings [15], each local model still needs to be evaluated for instruction following, context use, and source faithful answer generation.

This study proposes a RAG optimization framework for Indonesian-language educational Question Answering. The framework separates retrieval evaluation from generation evaluation so that errors can be traced to the relevant stage of the pipeline. The retrieval stage compares chunking strategies, multilingual embedding models, and reranking, while the generation stage evaluates local LLMs using RAGAS metrics. The framework is tested on a Human-Computer Interaction learning corpus. The study contributes an evaluation procedure for RAG configuration selection, empirical evidence on chunking, embedding, and reranking in an Indonesian educational corpus, and a comparison of local LLMs for source-grounded answer generation.

2. RESEARCH METHOD

This study adopts a Research and Development (R&D) approach, focused on developing a RAG optimization framework for an Indonesian-language educational question answering system. The research stages consist of Dataset Preparation, Retrieval Optimization Configuration, Generation Optimization Configuration, and Integration. The workflow of these stages is illustrated in Figure 1.

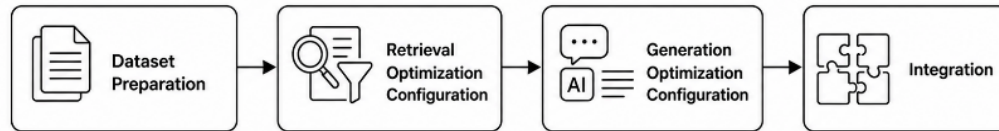


Figure 1. Research Stages

2.1 Dataset Preparation

In the dataset preparation stage, the learning source documents were prepared as the contextual basis for the RAG system and as the source for constructing the evaluation dataset. The documents were cleaned, standardized, and segmented into context-based chunks to preserve semantic coherence. To improve reproducibility, the question-answer generation process was configured using the OpenAI model identifier *gpt-5.1*. At runtime, the generation process used *temperature = 1.0* and *max_completion_tokens = 3000*, with the output requested in JSON array format. Each generated item consisted of *bloom_level*, *question*, *answer*, *evidence_span*, and *context_list*. The prompt consisted of a system instruction that directed the model to act as a Bloom's Taxonomy question generator and to produce only answerable questions grounded in the provided context. The user prompt supplied the anchor chunk, the allowed Bloom levels, and, for higher-order questions, one or two semantically related neighboring chunks selected through cosine similarity. Each generated item was also required to include an *evidence_span* copied directly from the source context so that the answer could be traced back to the learning material.

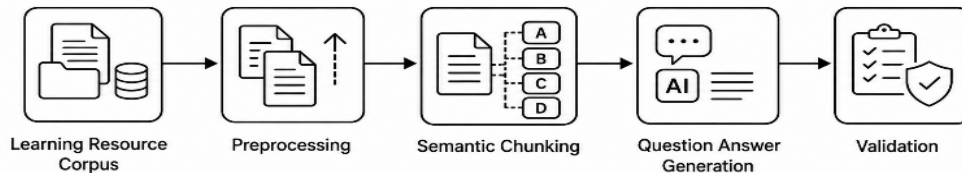


Figure 2. Dataset Preparation

2.1.1 Dataset Validation

Dataset validation was conducted in stages to ensure the suitability of the experimental items. First, all generated items were assessed by an LLM-based evaluator using two decision categories, namely KEEP and DROP, to examine the alignment between the source context, question, and answer. To evaluate the consistency of this validation process, a human expert audit was conducted on 200 items sampled from the initial LLM validation output, covering both KEEP and DROP decisions. The audit involved four human experts, with each expert reviewing 50 items using the same two decision categories. Agreement between the initial LLM-based evaluator and the human expert audit was analyzed using Cohen's Kappa, since the comparison was conducted between two decision sources for each audited item, namely the LLM-based evaluator and the corresponding human expert [17]. In Equation (1), (P_o) denotes the observed agreement, (P_e) denotes the agreement expected by chance, and (κ) represents the Cohen's Kappa coefficient.

$$\left[\kappa = \frac{P_o - P_e}{1 - P_e} \right] \quad (1)$$

This validation design follows the general principle used in Indonesian QA dataset construction, where automatic dataset construction or filtering is complemented by human evaluation to address scalability while still preserving quality control. Prior work on AC-IQuAD also argues that manual evaluation by human annotators is essential for dataset quality assessment, while automated evaluation can help address scalability constraints [18]. The generated answers were checked against the corresponding source chunks during the validation stage by assessing whether the answer was supported by the provided context and whether the question-answer pair was aligned with the source document.

2.2 Retrieval Optimization Configuration

The Retrieval Optimization Configuration stage aims to identify the best-performing retrieval configuration through a comparative evaluation of chunking and embedding technique combinations. This stage comprises the development of a vector database based on each chunking-embedding combination, and the evaluation of retrieval performance by querying the evaluation dataset items against the vector database. Each component is elaborated further as follows.

2.2.1. Vector Database Development

This stage aims to determine the best-performing combination of chunking strategy and embedding model in supporting the retrieval process within the RAG framework. As illustrated in *Figure 3*, the learning resource corpus is processed through three main stages: chunking, embedding, and storage into a vector database. Evaluation is performed across combinations of both variables, as they directly influence the relevance of the context retrieved during the retrieval stage.

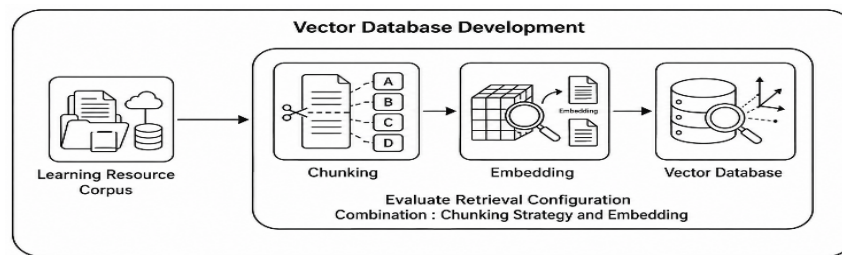


Figure 3. Vector Database Development

a. Chunking Technique Selection

This study compares three chunking techniques representing distinct segmentation strategies: (a) Fixed Size Chunking (chunk size of 400 tokens with an overlap of 80 tokens), serving as a stable and straightforward baseline; (b) Sentence-Based Chunking, which segments documents at sentence boundaries to preserve linguistic coherence; and (c) Semantic Chunking, which groups consecutive paragraphs based on semantic similarity using cosine similarity with a threshold of 0.80 applied to BAAI/bge-m3 embeddings. The use of semantic chunking in RAG-based Question Answering has also been employed to maintain semantic proximity across document sections [19].

b. Embedding Technique Selection

The selection of embedding models refers to the Massive Text Embedding Benchmark (MTEB) and the Retrieval Text Embedding Benchmark (RTEB) from HuggingFace, with a focus on multilingual retrieval performance [20]. Given that the experiments were conducted on a local NVIDIA RTX 3080 GPU with 10 GB VRAM, model selection was restricted to open-source SentenceTransformers-based models with fewer than one billion parameters. The five selected models are presented in *Table 1*.

Table 1. Embedder Models

No	Model Name	Mean Retrieval Score	Approx. Parameters
1	BGE-M3	69.59	~568M
2	SNOWFLAKE-ARCTIC-EMBED-L-V2.0	66.53	~335M
3	EMBEDDING GEMMA-300M	66.20	~300M

No	Model Name	Mean Retrieval Score	Approx. Parameters
4	GTE-MULTILINGUAL-BASE	64.17	~278M
5	MULTILINGUAL-E5-BASE	62.76	~278M

Each combination of chunking technique and embedding model is treated as an independent retrieval configuration, yielding a total of 15 configuration combinations (3 chunking techniques $\times$ 5 embedding models). Each combination produces a dedicated vector database, in which all chunks from the learning resource corpus are processed using the corresponding embedding model and stored in a separate vector index. This approach ensures that each configuration is evaluated independently without cross-combination interference, enabling a fair and controlled comparison of retrieval performance.

2.2.2. Retrieval Optimization Configuration

The retrieval evaluation stage aims to assess the ability of each configuration to retrieve context relevant to the questions in the RAG evaluation dataset. As illustrated in *Figure 4*, the process begins with the RAG Evaluation Dataset as input, after which the system performs retrieval against each of the vector databases constructed in the preceding stage. Each vector database represents a single chunking-embedding configuration combination, and retrieval is therefore executed separately for each combination.

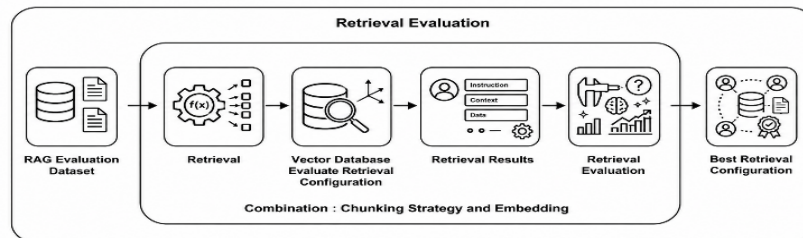


Figure 4. Retrieval Evaluation

During the retrieval stage, each question in the dataset is converted into a query embedding using the embedding model corresponding to the configuration under evaluation. The system then performs a search against the vector database to obtain the most relevant candidate chunks. The output of this stage consists of retrieval results, comprising the selected chunks along with their relevance scores for each question. The retrieval results are subsequently evaluated by computing retrieval metrics encompassing MRR, nDCG@K, and Hit@K [21]. Evaluation results at the question level are aggregated into a performance summary per experimental configuration. Through this process, all 15 configurations are compared to identify the best retrieval configuration, which is then carried forward to the subsequent stage.

BGE-RERANKER-V2-M3 was selected as the reranking model because it belongs to the BGE model family, which has been designed for multilingual retrieval and has shown strong multilingual and cross-lingual retrieval performance. The BGE-M3 model supports more than 100 languages and multiple retrieval functionalities, making it relevant for Indonesian-language retrieval settings [22]. In addition, cross-encoder reranking has been widely used to refine top-k candidates retrieved by first-stage retrievers, improving ranking quality at the cost of additional computation [23]. Therefore, the reranking experiment in this study was designed to examine the effect of adding a reranking stage after initial retrieval, rather than to provide a comprehensive comparison across multiple reranker models. In the retrieval experiment, each configuration was evaluated using hybrid search, combining BM25-based lexical retrieval and embedding-based dense retrieval before the optional reranking stage was applied to the selected configurations.

2.2.3. Retrieval Evaluation Metrics

Retrieval evaluation employs three rank-based metrics. Each question was associated with one or more ground-truth chunks recorded in the `context_list` field. For factual or lower-order questions, the ground truth typically consisted of the anchor chunk. For higher-order questions, particularly Analyze, Evaluate, and Create items, the ground truth could include multiple related

chunks selected during the context enrichment process. Therefore, retrieval metrics were computed by checking whether any of the ground-truth chunks appeared in the retrieved top-k results, while ranking-based metrics used the highest-ranked relevant ground-truth chunk.

Mean Reciprocal Rank (MRR) (2) measures ranking quality based on the position of the first relevant context; nDCG@K (3) assigns higher weight when relevant context appears at the top of the ranked list up to position K; and Hit@K (4) assesses whether relevant context is successfully retrieved within the top-K results, irrespective of its exact position. The use of top-K and rank-based metrics in RAG evaluation has been discussed in a RAG evaluation survey by [24], while retrieval quality evaluation in RAG is further addressed by [25]. In this study, the retrieval evaluation used ($K = 5$); therefore, Hit@K is reported as Hit@5 in the retrieval result tables.

$$MRR = \left(\frac{1}{N}\right) \sum_{i=1}^N \left(\frac{1}{rank_i}\right) \quad (2)$$

$$nDCG@K = \frac{DCG@K}{IDCG@K} \quad (3)$$

$$Hit@K = \frac{1}{N} \sum_{i=1}^N \mathbb{I}(R_i^K \cap G_i \neq \emptyset) \quad (4)$$

Where N denotes the total number of questions, $rank_i$ is the position of the highest-ranked relevant ground-truth chunk for question i , r the position at which the ground truth chunk appears, R_i^K is the set of chunks retrieved within the top-K results, G_i is the set of ground truth chunks, and $\mathbb{I}[\cdot]$ is an indicator function. All three metric values range from 0 to 1; nDCG@K and Hit@K take a value of 0 when the ground truth chunk is not found within the top-K results.

2.3 Generation Optimization Configuration

The generation selection stage aims to evaluate candidate LLMs in terms of their ability to produce relevant and accurate responses based on context retrieved through the best retrieval configuration. The candidate LLMs under comparison are the local LLMs listed in Table 2. In addition to these local LLMs, deepseek-chat was included as a service-based baseline to provide a comparative reference for generation quality. As illustrated in Figure 5, the process begins with the RAG Evaluation Dataset as the primary input, after which the system performs retrieval using the best retrieval configuration established in the preceding stage to ensure that the retrieved context remains consistent and controlled across all experiments. Once context is obtained through retrieval, the system performs prompt construction by combining the question and retrieved context into a predefined prompt format. This prompt is then passed to each candidate LLM to generate a response. The generation process is executed iteratively across all questions in the dataset, yielding a collection of responses from each model under evaluation.

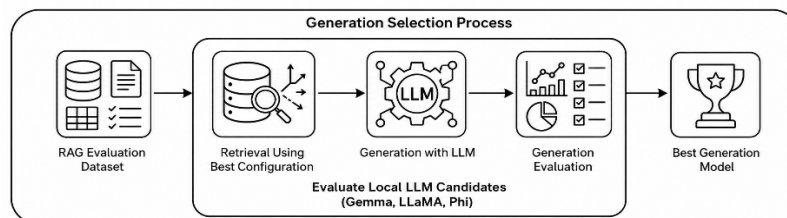


Figure 5. Generation Selection Process

The generated responses are subsequently evaluated using the RAGAS framework, which is designed to assess the quality of both retrieval and generation in RAG systems [26]. This evaluation examines several key aspects, including Context Precision, Context Recall, Faithfulness, Answer Correctness, and Answer Relevancy. The evaluation results across all candidate LLMs are then aggregated and compared to identify the best generation model, which is subsequently adopted as the generative component of the RAG system.

Table 2. Compared Local LLMs

Model Name	Model ID	Developer	Parameter Size
Phi 4	microsoft/phi-4	Microsoft	14B
Llama 3.1 8B	meta-llama/Llama-3.1-8B	Meta	8B
Gemma 4 E4B	google/gemma-4-E4B	Google	E4B

2.3.1 Generation Optimization Metrics

Generation quality is evaluated using the RAGAS framework through five metrics. Context Precision (CP) assesses whether relevant context is ranked at the top of the retrieved results; Context Recall (CR) measures the extent to which claims in the reference answer are supported by the retrieved context; Answer Relevancy (AR) evaluates the degree to which the generated response addresses the intent of the question; Faithfulness (F) measures the proportion of claims in the generated answer that can be traced back to the retrieved context; and Answer Correctness (AC) assesses the accuracy of the generated answer relative to the reference answer. All metrics are scaled from 0 to 1, with higher values indicating better quality.

$$CP@K = \frac{\sum_{k=1}^K (\text{Precision}@k \times v_k)}{\sum_{k=1}^K v_k} \quad (5)$$

$$CR = \frac{C_{supported}}{C_{reference}} \quad (6)$$

$$AR = \frac{1}{N} \sum_{i=1}^N \cos(E_{g_i}, E_q) \quad (7)$$

$$F = \frac{C_s}{C_a} \quad (8)$$

where (v_k) is the relevance indicator at position (k), ($\text{Precision}@k$) is the ratio of relevant context up to position (k), ($C_{supported}$) and ($C_{reference}$) are the number of reference-answer claims supported by the retrieved context and the total number of reference-answer claims, respectively, (E_{g_i}) and (E_q) are the embeddings of the (i)-th generated answer and the original question, respectively, ($\cos(\cdot)$) denotes cosine similarity, and (C_s) and (C_a) are the number of generated-answer claims supported by the retrieved context and the total number of generated-answer claims, respectively. Answer Correctness follows the standard RAGAS definition as a combined measure of factual and semantic similarity between the generated answer and the reference answer.

2.4 Integration

In the Integration stage, the best retrieval configuration and the best generation model obtained from the preceding stages are combined into a complete Question Answering system. The vector database constructed using the optimal chunking and embedding configuration serves as the context source, while the LLM with the highest generation performance functions as the answer generation component. The integration of these two components yields a RAG system optimized under the tested configuration, in which each submitted question undergoes relevant context retrieval followed by end-to-end answer generation.

3. RESULTS AND DISCUSSION

3.1. Dataset Development Results

The construction of the RAG evaluation dataset was carried out through three main stages as illustrated in *Figure 6*: semantic chunking, AI-based dataset item generation, and validation. In the semantic chunking stage, the three source documents were segmented into contextual units based on semantic coherence across sentences. File 1 produced 78 chunks, File 2 produced 111 chunks, and File 3 produced 95 chunks, yielding a total of 284 chunks from the entire corpus. Each chunk was subsequently used as the basis for automated dataset item generation using AI, whereby File 1 produced 690 items, File 2 produced 984 items, and File 3 produced 851 items. This process resulted in 2,525 initial dataset items, which then proceeded to the validation stage. Initial

validation was performed by an LLM-based evaluator using the deepseek-chat model, in which each item was assessed and categorized as either KEEP or DROP. Of the 2,525 items evaluated, 1,882 were classified as KEEP and retained as the research dataset.

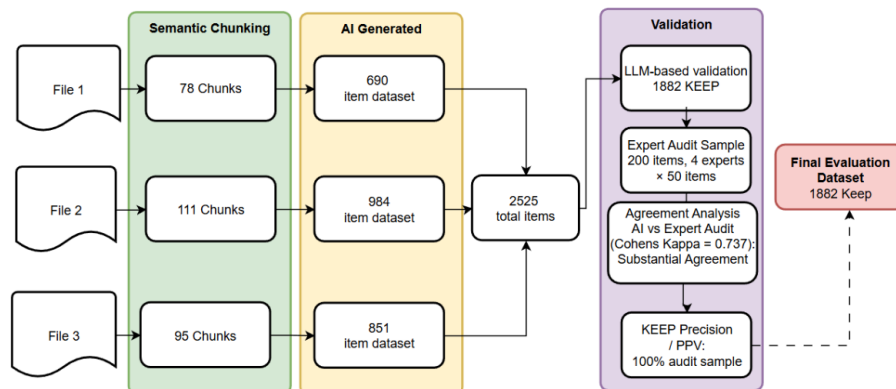


Figure 6. Dataset Development Results

To examine the reliability of the LLM-based validation decisions, an expert audit was conducted on a sample of 200 items drawn from the initial validation output. The audit involved four human experts, each reviewing 50 items using the same two decision categories, namely KEEP and DROP. The pooled comparison between the initial LLM-based evaluator and the human expert audit panel yielded a Cohen's Kappa value of $\kappa = 0.737$, with an observed agreement of 0.910 and an expected agreement of 0.658. Based on the Landis and Koch interpretation scale, this value falls within the substantial agreement category [27]. Pairwise comparisons between the LLM-based evaluator and individual human experts also showed substantial to almost perfect agreement, with Cohen's Kappa values ranging from 0.615 to 0.803 and percentage agreement ranging from 90.0% to 92.0%. The agreement values show that the LLM-based evaluator produced decisions that were generally consistent with the assessments of the human experts.

Table 3. Confusion Matrix of LLM-based Validation and Expert Audit

	Expert_KEEP	Expert_DROP
LLM_KEEP	148	0
LLM_DROP	18	34

Since the retained items labeled as KEEP by the LLM-based evaluator were the items carried forward to the retrieval and generation experiments, an additional class-specific validation was conducted using Positive Predictive Value (PPV), also referred to in this study as KEEP Precision. This metric was used to examine whether items labeled as KEEP by the LLM-based evaluator were also judged as KEEP by the human experts. Based on the audit confusion matrix (

Table 3), 148 items were labeled KEEP by both the LLM-based evaluator and the experts, while no item labeled KEEP by the LLM-based evaluator was changed to DROP by the experts. Therefore, the KEEP Precision reached 100%, indicating that, within the audited sample, the items retained by the LLM-based evaluator were consistently accepted by the human experts. The 18 cases labeled as DROP by the LLM-based evaluator but judged as KEEP by the experts suggest that the automated validator applied a stricter filtering criterion. This tendency may reduce dataset coverage by excluding some acceptable items, but it does not reduce the precision of the retained KEEP items, since no LLM-labeled KEEP item was judged as DROP by the experts.

3.2. Retrieval Optimization Results

Retrieval optimization was conducted to identify the configuration most capable of ranking relevant context at the top of the retrieved results. All evaluations employed hybrid search combining BM25 and embedding-based retrieval to capture both lexical and semantic matches for each query [28]. Hybrid search was selected on the grounds that user queries may relate to the

context through either semantic similarity or term overlap. A hybrid retrieval approach in RAG can help balance semantic and lexical search when documents exhibit terminological variation [29].

Table 4. Optimization Retrieval Results

Embedding Model	Chunking Technique	MRR	nDCG@5	HIT@5
E5	Fixed Size (400, 80)	0.7413	0.7738	0.8884
E5	Sentence Based	0.7455	0.7789	0.8831
E5	Semantic	0.7105	0.7484	0.8565
GTE	Fixed Size (400, 80)	0.5908	0.6291	0.7572
GTE	Sentence Based	0.5811	0.6202	0.7428
GTE	Semantic	0.5541	0.5980	0.7269
Snowflake	Fixed Size (400, 80)	0.7178	0.7530	0.8698
Snowflake	Sentence Based	0.7189	0.7551	0.8672
Snowflake	Semantic	0.6853	0.7263	0.8438
Gemma-300m	Fixed Size (400, 80)	0.7435	0.7746	0.8815
Gemma-300m	Sentence Based	0.7465	0.7812	0.8879
Gemma-300m	Semantic	0.7205	0.7551	0.8533
BGE M3	Fixed Size (400, 80)	0.7245	0.7575	0.8725
BGE M3	Semantic	0.7102	0.7479	0.8565
BGE M3	Sentence Based	0.7443	0.7765	0.8783

Evaluation was conducted in two stages. The first stage tested all combinations of embedding models and chunking techniques without a reranker, so as to reduce computational overhead during the initial selection process. The results presented in Table 4 indicate that sentence-based chunking was generally the most consistent technique, yielding the best configuration for four out of five embedding models tested. Gemma-300m with sentence-based chunking recorded the highest performance in this stage, achieving an MRR of 0.7465 and an nDCG@5 of 0.7812, followed closely by e5 and BGE M3, which produced nearly identical values. Snowflake exhibited a similar pattern, albeit with slightly lower scores. An exception was observed with GTE, where fixed-size chunking outperformed both sentence-based and semantic chunking. Meanwhile, semantic chunking consistently yielded the lowest scores across all models, indicating that the variable chunk lengths produced by the semantic approach are suboptimal for the dataset used in this experiment. The addition of the BGE-RERANKER-V2-M3 reranker improved performance across all selected configurations. The role of reranking in RAG has also been discussed as a mechanism for refining the ordering of context candidates prior to generation [29].

Table 5. Evaluation Retrieval with Reranking

Embedding Model	Chunking Technique	MRR	nDCG@5	HIT@5
E5	Sentence Based	0.8818	0.9003	0.9554
GTE	Fixed Size (400, 80)	0.9082	0.9215	0.9655
Snowflake	Sentence Based	0.8823	0.9006	0.9554
Gemma-300m	Sentence Based	0.8818	0.9003	0.9554
BGE M3	Sentence Based	0.8812	0.8996	0.9543

The results of the second evaluation stage presented in Table 5 demonstrate that the addition of the BGE-RERANKER-V2-M3 reranker improved performance across all selected configurations. The greatest improvement was observed in GTE with Fixed Size chunking, which had been the lowest-performing configuration in the first stage but emerged as the best configuration after reranking, achieving an MRR of 0.9082, an nDCG@5 of 0.9215, and a Hit@5 of 0.9655. The increase of +0.3174 in MRR indicates that, although GTE was suboptimal in ranking candidates during the initial stage, it was nonetheless capable of including relevant context within the top-K results, enabling the reranker to effectively refine the ordering. Other configurations, including E5, Gemma-300m, Snowflake, and BGE M3, also experienced substantial improvements, with MRR gains ranging from +0.1353 to +0.1634, and all four reaching nearly equivalent final scores following reranking. These findings indicate that the reranker substantially improves retrieval ranking quality, particularly for configurations that underperform during initial retrieval.

3.3. Generation Selection Results

Generation evaluation was conducted to assess the quality of responses produced by each LLM within the RAG system. The retrieval configuration used was the best-performing configuration from the preceding stage, namely GTE with Fixed Size Chunking (400, 80), Hybrid Search BM25, top-K = 5, and the BGE-RERANKER-V2-M3 reranker. The models compared consisted of three local LLMs: Phi 4, Llama 3.1 8B, and Gemma 4 E4B. In addition, deepseek-chat was included as an external service-based baseline to provide a reference point for comparing the generation quality of the local LLMs.

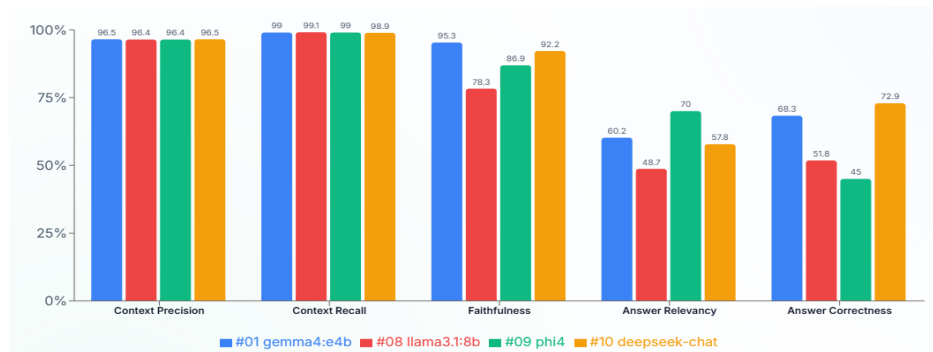


Figure 7. Comparative evaluation results across RAGAS metrics

The evaluation results in Figure 7 show that all models achieved high and uniform CP and CR scores, with CP ranging from 96.40% to 96.53% and CR from 98.88% to 99.13%. As all models employed an identical retrieval configuration, performance differences across models are more pronounced in Faithfulness, AR, and AC. In terms of Faithfulness, Gemma 4 E4B recorded the highest score of 95.29%, surpassing the deepseek-chat baseline at 92.21%, followed by Phi 4 (86.89%) and Llama 3.1 8B (78.27%). For AR, Phi 4 led with 70.04%, while in AC deepseek-chat achieved the highest score of 72.95%, with Gemma 4 E4B in second place at 68.30%. Based on the aggregate mean score across all five metrics as presented in Figure 7, Gemma 4 E4B attained the highest overall score of 83.8%, slightly higher than the deepseek-chat baseline at 83.7%, followed by Phi 4 at 79.5% and Llama 3.1 8B at 74.9%. Despite the narrow margin, Gemma 4 E4B was selected as the best local LLM owing to its more stable trade-off across metrics, particularly in Faithfulness, alongside an AC score closely approaching the deepseek-chat baseline. In the context of educational question answering, answer fidelity to the source is a primary consideration, as the system must produce responses that are not only relevant but also accountable with respect to the learning materials used.

3.4. Integration Results

The integration stage combined the best retrieval configuration, namely GTE with fixed-size chunking and BGE-RERANKER-V2-M3, with the best generation model, Gemma 4 E4B. In a free-form interaction example, the system retrieved five chunks and generated a structured answer grounded in the retrieved context. The interface also displayed real-time RAGAS scores, including CP of 1.00, Faithfulness of 1.00, and AR of 0.98, indicating that the generated answer was precise, supported by the retrieved context, and relevant to the user question. CR and AC were not displayed in this mode because both metrics require a ground-truth answer.

3.5. Discussion

The retrieval results suggest that the first-stage retriever and the reranker played different roles in the RAG pipeline. As shown in Figure 8, the addition of BGE-RERANKER-V2-M3 increased MRR across all selected configurations, with the largest gain appearing in GTE with fixed-size chunking. This pattern suggests that the initial retriever had already included relevant candidates, although not always at the highest ranks. The reranker then improved the ordering of these candidates before they were passed to the generation stage. Before reranking, sentence-based chunking produced more stable retrieval because its segments preserved sentence boundaries and

reduced context fragmentation. After reranking, GTE with fixed-size chunking achieved the best result, indicating that broader fixed-size chunks may provide enough contextual material for the cross-encoder reranker to distinguish relevant evidence more effectively. Thus, the best initial retrieval configuration is not necessarily the best end-to-end retrieval configuration once reranking is introduced.

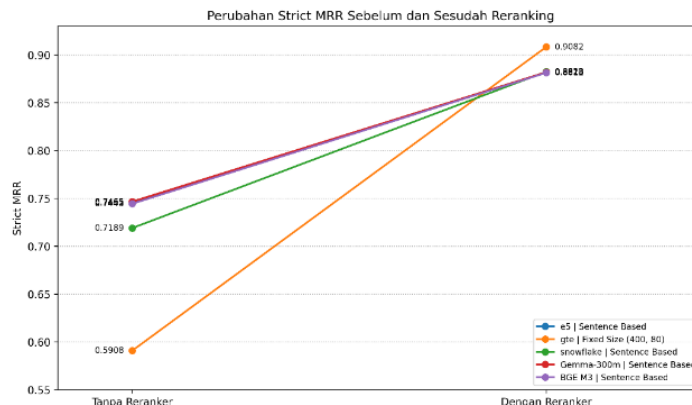


Figure 8. MRR Improvement Following Reranker Application Across Selected Retrieval Configurations

The generation results indicate that retrieval quality alone did not determine answer quality. As shown in Figure 7, all models obtained similarly high CP and CR scores under the same optimized retrieval setting, which means that each model received comparable retrieved context. The main differences appeared in Faithfulness, Answer Relevancy, and Answer Correctness, which depend more on how each model interprets and uses the retrieved evidence during generation. Gemma 4 E4B was therefore selected because it offered a safer trade-off for educational question answering: high source fidelity, competitive answer correctness, and a stable aggregate score relative to the external service-based baseline. In this setting, source fidelity is more important than producing a fluent answer that is only loosely connected to the learning material.

4. CONCLUSION

This study proposed a RAG optimization framework that separates retrieval and generation evaluation for an Indonesian-language educational question answering system, applied to a Human-Computer Interaction learning corpus. The findings show that RAG performance should be optimized as a pipeline process, since retrieval configuration, reranking, and generator selection jointly determine answer quality and source fidelity. In the retrieval stage, sentence-based chunking was the most stable before reranking, while GTE with fixed-size chunking achieved the best performance after applying BGE-RERANKER-V2-M3, with MRR of 0.9082, nDCG@5 of 0.9215, and Hit@5 of 0.9655. In the generation stage, Gemma 4 E4B produced the most balanced performance, with the highest Faithfulness score of 95.29% and an aggregate score of 83.8%. The transferable part of this study is the evaluation procedure: separating retrieval and generation evaluation, comparing chunking and embedding settings, analyzing reranking effects, and selecting the generator using RAGAS metrics. However, the specific optimal configuration should be considered corpus-specific and should not yet be generalized to all Indonesian educational QA systems. Future research should evaluate the framework across broader learning domains, multiple courses, different academic disciplines, and more heterogeneous document types, in line with adaptive RAG studies that emphasize sensitivity to domain, document type, and task requirements [30].

This study has several limitations. Dataset validation relied on an LLM-based evaluator supported by sampled human expert auditing. Although the expanded audit of 200 items involving four human experts showed substantial agreement with the initial LLM-based decisions, with Cohen's Kappa $\kappa = 0.737$ and KEEP Precision of 100% in the audit sample, the validation remains an initial filtering process rather than exhaustive human verification. Bloom's Taxonomy labels

were generated automatically as auxiliary annotations to support question diversity and were not exhaustively verified manually across all retained items. The optimization was also limited to a basic RAG architecture without query rewriting, multi-hop retrieval, or self-reflection. In addition, the reranking evaluation used only BGE-RERANKER-V2-M3, generation evaluation depended on the LLM-based evaluator used in RAGAS, and the corpus was restricted to one course domain, three source documents, and one GPU environment.

ACKNOWLEDGEMENTS

This study forms part of the author's doctoral research. The author expresses sincere appreciation to Universitas Udayana for its academic guidance and support during the doctoral journey. The author also acknowledges Universitas Pendidikan Ganesha for providing institutional facilities and research infrastructure that enabled the completion of this work.

CONFLICT OF INTEREST

The authors declare that they have no conflict of interest.

DATA AVAILABILITY

The data underlying this article will be shared upon reasonable request to the corresponding author.

AUTHOR CONTRIBUTIONS

I Ketut Resika Arthana: Conceptualization; Methodology; Software; Formal Analysis; Investigation; Data Curation; Writing Original Draft; Visualization. Nyoman Gunantara: Conceptualization; Validation; Supervision; Project Administration; Writing Review and Editing. Made Sudarma: Methodology; Supervision; Writing Review & Editing. I Made Sukarsa: Methodology; Software; Supervision; Writing Review and Editing.

DECLARATION OF GENERATIVE AI USE

Generative AI tools were used to support the generation of initial question-answer dataset candidates from source chunks, preliminary LLM-based dataset filtering, and language refinement. Human expert auditing and author review were conducted to verify the dataset quality, analysis, interpretation, and conclusions. The authors take full responsibility for the content of the manuscript.

REFERENCES

- [1] J. Chen, H. Lin, X. Han, and L. Sun, "Benchmarking large language models in retrieval-augmented generation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024, vol. 38, no. 16. doi: 10.1609/aaai.v38i16.29728.
- [2] L. Huang *et al.*, "A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions," *ACM Transactions on Information Systems*, vol. 43, no. 2, pp. 1–55, Mar. 2025, doi: 10.1145/3703155.
- [3] I. K. R. Arthana, N. Gunantara, M. Sudarma, and M. Sukarsa, "LoRA-based fine-tuning of local LLMs for hallucination detection in Indonesian RAG systems," *International Journal of Advanced Computer Science and Applications*, vol. 17, no. 3, 2026, doi: 10.14569/IJACSA.2026.0170389.
- [4] Y. Gao *et al.*, "Retrieval-Augmented Generation for Large Language Models: A Survey," 2024, doi: <https://doi.org/10.48550/arXiv.2312.10997>.
- [5] D. Danter, H. Mühle, and A. Stöckl, "Advanced Chunking and Search Methods for Improved Retrieval-Augmented Generation (RAG) System Performance in E-Learning," 2024. doi: 10.54941/ahfe1005756.
- [6] F. Golla, "Enhancing Student Engagement Through AI-Powered Educational Chatbots: A Retrieval-Augmented Generation Approach," in *2024 21st International Conference on Information Technology Based Higher Education and Training, ITHET 2024*, 2024. doi: 10.1109/ITHET61869.2024.10837678.

- [7] I. K. R. Arthana, N. Gunantara, M. Sudarma, and M. Sukarsa, "A Systematic Literature Review of Retrieval-Augmented Generation Implementation for Enhancing Large Language Models in Education," *Jurnal Nasional Pendidikan Teknik Informatika (JANAPATI)*, vol. 15, no. 1, pp. 91–109, Mar. 2026, doi: 10.23887/janapati.v15i1.112281.
- [8] C. Sarmiento and E. J. M. Lauría, "Investigating Flavors of RAG for Applications in College Chatbots," in *International Conference on Computer Supported Education, CSEDU - Proceedings*, 2025, vol. 2, pp. 421–428. doi: 10.5220/0013468200003932.
- [9] S. Liang, X. Gong, F. Li, and C. Liao, "UniEval-RAG: A unified end-to-end evaluation framework for RAG systems," in *Proceedings - 2025 International Conference on Computer, Internet of Things and Smart City (CIoTSC 2025)*, 2025. doi: 10.1109/CIoTSC67482.2025.11412981.
- [10] J. Bossenz *et al.*, "Evaluation of Chunking and Embedding Strategies for Local Document Retrieval Using an Open-Source LLM in a Hospital," 2025. doi: 10.3233/SHTI251383.
- [11] Y. Song, L. Liu, H. Wang, and J. Liu, "A survey of retrieval granularity in retrieval-augmented generation," *Data Analysis and Knowledge Discovery*, 2026, doi: 10.11925/infotech.2096-3467.2025.0349.
- [12] R. D. Pesl, J. G. Mathew, M. Mecella, and M. Aiello, "Retrieval-Augmented Generation for Service Discovery: Chunking Strategies and Benchmarking," *IEEE Transactions on Services Computing*, pp. 1–15, 2026, doi: 10.1109/TSC.2026.3665441.
- [13] X.-K. Koay, L.-Y. Ong, and P.-Y. Goh, "Structure-Aware Chunking for Complex Tables in Retrieval-Augmented Generation Systems," *Emerging Science Journal*, vol. 10, no. 1, pp. 184–205, Feb. 2026, doi: 10.28991/ESJ-2026-010-01-09.
- [14] Y. Niu and X. Rong, "Improving retrieval-augmented generation for educational policy understanding via structure-aware text chunking," in *IECA 2026*, 2026. doi: 10.1145/3802133.3802307.
- [15] Z. Yu, S. Liu, P. Denny, A. Bergen, and M. Liut, "Integrating Small Language Models with Retrieval-Augmented Generation in Computing Education: Key Takeaways, Setup, and Practical Insights," in *SIGCSE TS 2025 - Proceedings of the 56th ACM Technical Symposium on Computer Science Education*, 2025, vol. 1, pp. 1302–1308. doi: 10.1145/3641554.3701844.
- [16] S. F. Chaerul Haviana, M. Riyadi, and R. Kusumaningrum, "Evaluation of chunking strategies in RAG application for explicit retrieval on Indonesian language scientific papers," in *Proceedings of EECSI 2025*, 2025. doi: 10.1109/EECSI67060.2025.11290624.
- [17] J. Cohen, "A Coefficient of Agreement for Nominal Scales," *Educational and Psychological Measurement*, vol. 20, no. 1, pp. 37–46, Apr. 1960, doi: 10.1177/001316446002000104.
- [18] K. Doxolodeo and A. A. Krisnadhi, "AC-IQuAD: Automatically Constructed Indonesian Question Answering Dataset by Leveraging Wikidata," *Language Resources and Evaluation*, vol. 59, no. 1, pp. 135–160, Mar. 2025, doi: 10.1007/s10579-023-09702-y.
- [19] L. Zhang and Y. Ning, "Improving construction contract question answering through embedding optimization and semantic chunking in large language models," *Advanced Engineering Informatics*, vol. 69, p. 104027, Jan. 2026, doi: 10.1016/j.aei.2025.104027.
- [20] N. Muennighoff, N. Tazi, L. Magne, and N. Reimers, "MTEB: Massive Text Embedding Benchmark," in *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, 2023, pp. 2014–2037. doi: 10.18653/v1/2023.eacl-main.148.
- [21] K. Järvelin and J. Kekäläinen, "Cumulated gain-based evaluation of IR techniques," *ACM Transactions on Information Systems*, vol. 20, no. 4, pp. 422–446, Oct. 2002, doi: 10.1145/582415.582418.
- [22] J. Chen, S. Xiao, P. Zhang, K. Luo, D. Lian, and Z. Liu, "M3-Embedding: Multi-Linguality, Multi-Functionality, Multi-Granularity Text Embeddings Through Self-Knowledge Distillation," in *Findings of the Association for Computational Linguistics ACL 2024*, 2024, pp. 2318–2335. doi: 10.18653/v1/2024.findings-acl.137.

- [23] R. Nogueira and K. Cho, "Passage Re-ranking with BERT," Apr. 2020, doi: <https://doi.org/10.48550/arXiv.1901.04085>.
- [24] H. Yu, A. Gan, K. Zhang, S. Tong, Q. Liu, and Z. Liu, "Evaluation of Retrieval-Augmented Generation: A Survey," Jul. 2024, doi: 10.1007/978-981-96-1024-2_8.
- [25] A. Salemi and H. Zamani, "Evaluating Retrieval Quality in Retrieval-Augmented Generation," in *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, Jul. 2024, pp. 2395–2400. doi: 10.1145/3626772.3657957.
- [26] S. Es, J. James, L. Espinosa Anke, and S. Schockaert, "RAGAs: Automated Evaluation of Retrieval Augmented Generation," in *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, 2024, pp. 150–158. doi: 10.18653/v1/2024.eacl-demo.16.
- [27] J. R. Landis and G. G. Koch, "The Measurement of Observer Agreement for Categorical Data," *Biometrics*, vol. 33, no. 1, p. 159, Mar. 1977, doi: 10.2307/2529310.
- [28] S. Robertson and H. Zaragoza, "The Probabilistic Relevance Framework: BM25 and Beyond," *Foundations and Trends® in Information Retrieval*, vol. 4, no. 1–2, pp. 1–174, Sep. 2009, doi: 10.1561/15000000019.
- [29] T. Şakar and H. Emekci, "Maximizing RAG efficiency: A comparative analysis of RAG methods," *Natural Language Processing*, vol. 31, no. 1, pp. 1–25, Jan. 2025, doi: 10.1017/nlp.2024.53.
- [30] M. Zhou, J. Tang, W. Zeng, and X. Zhao, "INKER: Adaptive dynamic retrieval augmented generation with internal-external knowledge integration," *Information Processing & Management*, vol. 63, no. 3, p. 104534, Apr. 2026, doi: 10.1016/j.ipm.2025.104534.