

Retrieval-Augmented Guidance for Inclusive Teaching in Higher Education

William Massami Costa Harada¹, Tiago Eugênio de Melo¹, Fábio Santos da Silva¹

¹Escola Superior de Tecnologia – Universidade do Estado do Amazonas (EST-UEA)
Grupo de Pesquisa em Sistemas Inteligentes

{wmch.eng22, tmelo, fssantos}@uea.edu.br

Abstract. *To support inclusive teaching practices in higher education, this work presents a Retrieval-Augmented Generation system that provides educators with assistance supported by peer-reviewed literature. The system combines a curated corpus that is automatically expanded as new research is discovered, with a document ingestion procedure that uses a VLM to preserve document structure. We benchmarked seven embedding models and six open-weights LLMs using the RAGAS framework, and the best configuration achieved a document-level Recall@10 of 0.91 and a Faithfulness score of 0.92, suggesting that RAG is a promising approach for supporting educators in adopting inclusive practices for teaching students with disabilities.*

1. Introduction

Inclusion in higher education takes place, to a large extent, in the classroom, where the teaching practices adopted by educators directly influence the opportunities for participation, learning, and academic success of students with disabilities. Data from the latest United Nations Disability and Development Report (2024) [UNDESA 2025] show that only 6 % of persons with disabilities in participating countries completed higher education, compared with 16 % of persons without disabilities in the same group. This disparity was already evident in the previous report of 2018 [UNDESA 2019], and remains one of the challenges addressed by the Sustainable Development Goals (SDGs)[United Nations 2015], which promotes inclusive, equitable, and high-quality education, as well as lifelong learning opportunities for all. Reducing these disparities depends not only on institutional commitments but also on providing educators with the knowledge and resources needed to effectively implement inclusive practices, a challenge that has driven efforts aimed at developing tools to support teachers in inclusive education [Lopez-Gazpio 2025].

Among these efforts, artificial intelligence has been increasingly explored as a way to help educators adapt their pedagogical practices to the needs of students with disabilities. Within this field, generative models, particularly Large Language Models (LLMs), have been widely adopted in educational applications [Pan et al. 2025], as they are trained on large amounts of text and can answer natural-language questions fluently, making them useful for educational support across multiple topics and domains. However, LLMs produce responses based solely on learned representations encoded in their model weights, which makes their internal knowledge difficult to update as new data becomes available, since incorporating it requires expensive retraining due to the large number of parameters, while these models also remain prone to hallucinations. To address these limitations,

Retrieval-Augmented Generation (RAG) has been proposed, coupling the LLM with an external knowledge source so that, at inference time, relevant passages are retrieved and provided as context, grounding the generated response in retrieved evidence instead of relying solely on the model weights [Lewis et al. 2020].

Based on this approach, this work proposes RAGIT, a RAG-based system that grounds LLM responses in a curated corpus of peer-reviewed literature on inclusive higher education that can automatically expand as new research is discovered, allowing the knowledge base to stay updated as research advances without retraining the model. We also developed a mobile application that allows educators to interact with the system and inspect the sources that support each generated answer.

To detail the proposed approach, this paper is organized as follows: Section 2 reviews related work, Section 3 describes the methodology, covering the ingestion, retrieval, and generation modules, Section 4 reports the experimental results and presents the application developed, and Section 5 discusses the results, limitations and future work.

2. Related Work

A literature review was conducted focusing on applications of RAG in educational contexts and approaches for extracting and structuring information from visually complex documents such as academic papers. The main contributions identified are discussed throughout this section.

In the context of RAG usage to provide educational assistance in higher education contexts, [Hennekeuser et al. 2025] investigated the use of LLMs combined with RAG to support university lecturers in higher education. Their system integrates lecture scripts and textbooks in PDF format into a retrieval pipeline that indexes their semantic embeddings into a vector database, which is queried with the embeddings of the user query, and retrieves the most relevant content based on semantic similarity, that is further injected into the context of an LLM to generate responses based on it. The authors conducted workshops with lecturers to refine prompt engineering strategies and evaluated the system through controlled experiments and qualitative interviews with nine university lecturers, and found that explainability, controllability, reliability, and trustworthy source attribution are considered the most important requirements for adopting RAG-based educational assistants.

More specifically focused on disability support, [dos Santos et al. 2025] proposed TEAChat, a Brazilian chatbot based on RAG designed to support the education and inclusion of individuals with Autism Spectrum Disorder. The system indexes articles, clinical guidelines, and educational resources into a vector database, after preprocessing and embedding generation, to provide personalized responses adapted to educators and family members. The retrieved context is incorporated into prompts sent to the LLM, allowing the system to generate responses based on trusted sources while adapting response tones according to the user profile category. The authors further discuss how conversational AI systems may support inclusion and access to reliable information in educational contexts.

Recent work has also investigated the broader adoption of generative AI to support teaching activities while preserving teacher autonomy in the Brazilian context. [Matos et al. 2024] argue that AI-generated educational content should remain under

teacher supervision because pedagogical decisions depend on contextual knowledge of students and require verifiable information. Similarly, [Reis-Andersson 2024] investigated teachers' perceptions of generative AI and found that educators value these tools for lesson planning, assessment preparation, and content creation, while expressing concerns regarding source reliability, teacher training, and the ability to address students' individual needs. Together, these studies indicate that AI should function as a decision-support technology, providing trustworthy information while preserving educators' central role in pedagogical decision-making.

Beyond retrieval and response generation, the quality of RAG systems also depends on how documents are ingested and represented in the knowledge base. Academic publications present particular challenges in this regard, as their layouts frequently include two-column formats, floating tables, and embedded figures that standard text extraction tools tend to mishandle. Vision-Language models (VLMs) have emerged as a more robust alternative to traditional OCR pipelines in this context, and the work of [Galfre et al. 2025] compares OCR+LLM pipelines with end-to-end VLMs from Qwen2.5 family for extracting structured information, from medical leave certificates, using a dataset of 183 real-world documents. The results showed that the VLMs achieved the best overall performance, reaching an average accuracy score of 81.37 %, compared with 64.25 % for the baseline. The authors also reported that this approach was substantially faster, reducing average inference time from 14.96 seconds in the OCR+LLM pipeline to 3.09 seconds with Qwen2.5-VL-7B. Other works combine VLMs with RAG more directly, but embedding document pages as visual representations rather than extracted text [Yu et al. 2025].

Existing work indicates that improving educational RAG systems depends on multiple complementary aspects. First, the quality of the knowledge base directly affects the reliability of generated responses, motivating the use of curated sources. Second, document ingestion should preserve the semantic structure of academic publications to improve retrieval quality. Third, transparency is considered essential for educator trust and adoption. However, existing systems generally assume static corpora and conventional document parsing, leaving document acquisition and corpus maintenance outside the RAG pipeline.

Our work addresses these challenges by combining (i) automated literature discovery and acquisition to support continuous knowledge base updates, (ii) A VLM-based document ingestion approach that preserves document hierarchy during chunk construction and (iii) metadata-enriched retrieval to improve traceability. Together, these components extend previous educational RAG architectures beyond retrieval and response generation, encompassing the complete knowledge acquisition and maintenance workflow.

3. Methodology

RAGIT is implemented as a RAG pipeline composed of three main modules, illustrated in Figure 1: an offline ingestion module responsible for collecting and updating the document corpus (Section 3.1), a retrieval module that parses and indexes the documents into a searchable vector database (Section 3.2), and a generation module that synthesizes answers grounded in the retrieved content (Section 3.3).

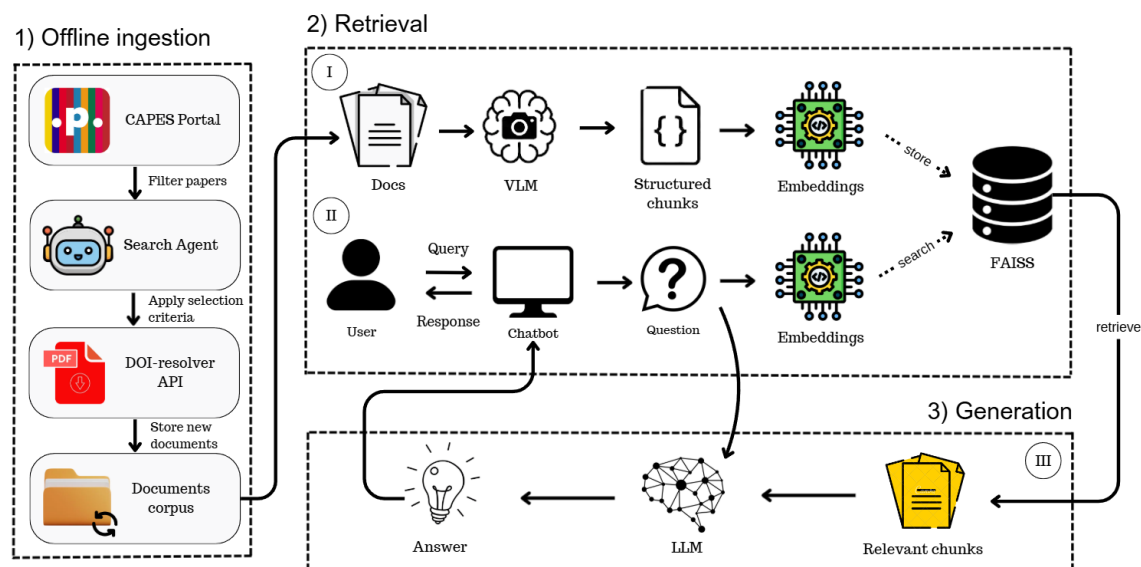


Figure 1. System architecture diagram.

3.1. Offline Ingestion

The offline ingestion module prepares the document corpus before it is indexed and queried by RAGIT. As shown in Figure 1, this module operates before user interaction and produces the set of PDF documents that will later be processed by the retrieval pipeline.

The module is coordinated by a Search Agent that queries the CAPES Journals Portal using a configurable search string, publication window, document-type filter, and access constraints. The agent first retrieves candidate records and then is prompted to apply the predefined selection criteria in Table 1 to perform preliminary screening based on their abstracts. The inclusion criteria were defined to ensure that the knowledge base was composed of peer-reviewed papers directly relevant to inclusive teaching in higher education. The criteria were established based on the scope of the proposed assistant, whose objective is to provide guidance to higher education educators, prioritizing publications that present pedagogical strategies, accessibility practices, or educational interventions. For candidates that satisfy these criteria, the agent resolves the DOI and queries an open-access API to locate and download the PDF, which is downloaded and stored in the knowledge base for later indexing. This process is intended to support continuous database updating as new research is published.

Since the Search Agent relies on an LLM-based decision, its decisions are inherently probabilistic and may introduce classification errors during large-scale corpus expansion. False positives may incorporate studies that do not satisfy the inclusion criteria, increasing noise in the vector index and potentially reducing retrieval precision, whereas false negatives may exclude relevant publications, limiting the coverage of the knowledge base. These challenges are inherent to automated literature screening and should be considered when deploying continuously updated RAG systems, as they directly affect the quality of the indexed corpus and, consequently, generation performance.

For the experiments reported in the next sections, the document corpus used for

evaluation was manually collected from the CAPES Journals Portal¹, following the criteria in Table 1. This process resulted in a ground-truth corpus of 55 documents in the English language, comprising empirical studies, theoretical frameworks, and policy analyses. The relatively small corpus size reflects the scarcity of available literature that simultaneously meets all inclusion criteria, yet this approach ensures that the retrieval index contains only validated documents.

Table 1. Inclusion and Exclusion Criteria for Knowledge Base Composition

Dimension	Inclusion	Exclusion
Topic	Pedagogical strategies or methodologies for the inclusion of students with disabilities in higher education; challenges encountered by disabled students in academic settings.	Studies focused solely on legislation without pedagogical application.
Population	Any specific disability category or generalist cross-population approaches.	Clinical health studies lacking a clear pedagogical link.
Availability	Full text accessible.	Abstract-only available or restricted access.
Content	Studies with clearly described methodology and results presented with analysis or discussion.	Research proposals, results lacking analysis or discussion, and bibliographic reviews.

3.2. Retrieval

The retrieval module transforms the PDF documents stored in the corpus into a searchable vector index. As illustrated in Figure 1, this module receives the document corpus produced by the offline ingestion stage, parses each paper into a structured representation, segments the resulting content into metadata-enriched chunks, and stores their embeddings in a FAISS (Facebook AI Similarity Search) [Douze et al. 2026] index.

To parse the documents before indexing, we apply a VLM-based ingestion pipeline. Each PDF page is first converted into a high-resolution image and processed by Qwen2.5-VL. Unlike OCR engines, which primarily extract text line by line, the VLM analyzes the visual structure of each page to identify elements such as section headers, paragraphs, lists, tables, and figure captions according to their layout. This enables the model to preserve the logical organization of the document rather than only its textual content. Figure 2 illustrates this workflow, in which each PDF page is transformed into a structured JSON representation that preserves the document’s section hierarchy.

The JSON output contains normalized fields describing the document structure, including section titles and their corresponding textual contents. During ingestion, the page-level outputs are merged into a single hierarchical representation of the article and enriched with metadata (paper title, publication year, and authors) obtained during the document acquisition stage. This structured representation serves as the basis for the chunk generation process, ensuring that retrieval units are created within section boundaries rather than from arbitrary text windows. Consequently, the semantic context of each passage is preserved while remaining robust to variations in formatting and layout across different academic publications.

¹<https://www.periodicos.capes.gov.br>

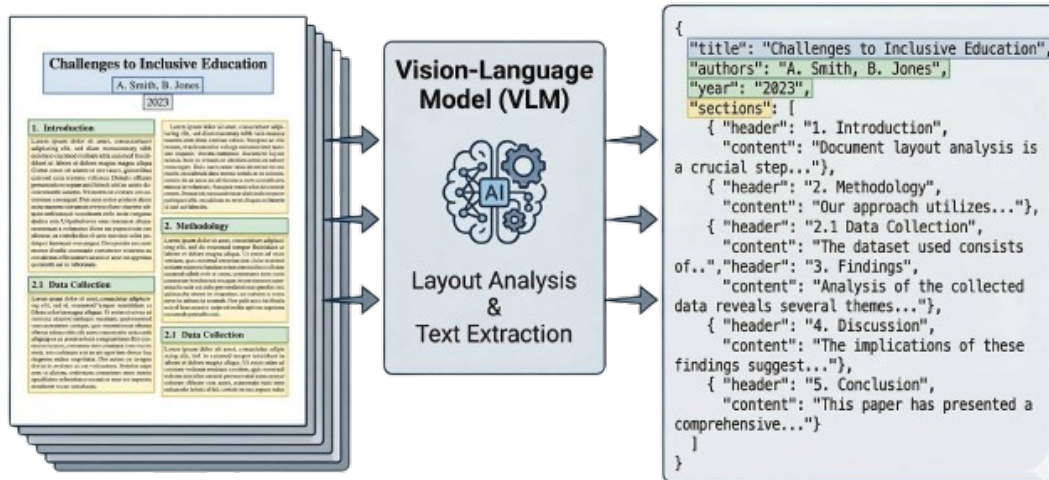


Figure 2. Visual-Language Document Ingestion pipeline. A PDF page image is processed by a VLM, which identifies semantic boundaries and produces a structured JSON representation preserving the document's section hierarchy.

After parsing, each section of the structured document is processed independently to generate the retrieval units. First, the text is segmented according to the paragraph boundaries identified by the VLM. Consecutive paragraphs are then aggregated until a maximum length of 512 tokens. To reduce information loss at chunk boundaries, adjacent chunks share an overlap of 50 tokens, preserving contextual continuity and improving the semantic coherence of consecutive retrieval units.

Each chunk constitutes a retrieval unit and is enriched with metadata, that is later injected into the LLM context to provide source traceability. The enriched chunks are then encoded by an embedding model into fixed-length dense vectors, whose dimensionality depends on the selected embedding model. The resulting vectors are stored in the vector database, which uses inner-product similarity to rank the most relevant chunks during retrieval. This indexed collection forms the searchable knowledge base queried at runtime whenever an educator submits a question.

3.3. Generation

The generation module is responsible for producing the final answer shown to the user. As illustrated by the steps (ii) and (iii) in Figure 1, the interaction begins when the user submits a query through the chatbot interface. The question is encoded using the same embedding model used during indexing, and then a nearest-neighbor search is performed over the FAISS index to retrieve the top- k most relevant chunks. Each retrieved chunk is provided to the LLM together with its paper title, publication year, and section title.

After retrieval, the LLM synthesizes the retrieved chunks into a response. The system prompt constrains the model to answer only using the provided context, as illustrated below:

Assistant prompt

You are a pedagogical assistant. Your task is to answer the given question using **ONLY** the provided content, extracted from academic papers.

To proceed with the task, you shall follow the rules below:

1. Every sentence must end with a citation (Author, Year).
2. Report findings exactly as described in the text.
3. If the context is empty or irrelevant, say “I don’t have enough knowledge to answer this.”

Now, proceed with the task:

Context: **{sections_with_metadata}**

Question: **{question}**

Answer:

4. Results and Discussion

The experiments described in this section were conducted on an Acer Veriton GN100 AI Workstation equipped with 128GB unified memory. The vector index was implemented with FAISS, and model inference was performed locally using vLLM².

4.1. Document ingestion

To characterize the computational requirements of the proposed ingestion pipeline, we measured the processing time, token consumption, and number of inference attempts required to convert the collected documents into structured JSON representations. Across the 865 processed pages, the VLM pipeline required an average of 94.7 s per page, consuming approximately 2,305 vision input tokens and 1,029 output tokens per page. The average number of processing attempts was 1.04 per page, indicating that nearly all pages were successfully processed in a single inference. Since the ingestion stage is executed only during the initial corpus indexing and the updates with the publications identified by the Search Agent, the computational cost of this stage does not affect query-time latency.

4.2. Retrieval Evaluation

To evaluate retrieval performance, we tested seven open-weight embedding models over our document corpus. For each embedding model, a separate FAISS index was constructed, containing the embeddings of the 2692 chunks extracted from the 55 collected documents. Performance was measured using Mean Reciprocal Rank (MRR@k), Precision@k, and Recall@k, with $k \in \{3, 5, 10\}$.

To query each index, a set of questions was manually constructed from the collected papers. For each document, a human annotator wrote three queries that could be answered using the paper’s content and that reflected realistic questions an educator might plausibly ask the system, while also identifying the paper expected to contain the relevant content. In total, the test set comprised 165 queries derived from the 55 collected documents.

²<https://vllm.ai>

Although retrieval is performed over paragraph-level chunks, evaluation is carried out at the document level, since the relevant information for a query is not always restricted to a single chunk and may instead be distributed across multiple sections of an academic paper, with different chunks providing partial or complementary information that is still useful. For this reason, retrieved chunks were mapped back to their source papers before scoring, aligning the evaluation with the objective of assessing whether the retriever identifies the relevant source documents. Among the reported metrics, Recall@k is of particular importance, as the generative model can ignore irrelevant chunks but cannot recover relevant sources that were never retrieved.

Table 2. Embedding Model Performance Comparison

Model	Dim	MRR@k	Precision@k	Recall@k
<i>k = 3</i>				
jina-embeddings-v2-base-en [Günther et al. 2023]	768	0.6626	0.2465	0.7394
e5-base-v2 [Wang et al. 2022]	768	0.6949	0.2586	0.7758
nomic-embed-text-v1.5 [Nussbaum et al. 2024]	768	0.7152	0.2626	0.7879
all-MiniLM-L12-v2 [Wang et al. 2020]	384	0.6071	0.2384	0.7152
bge-small-en-v1.5 [Xiao et al. 2023]	384	0.6485	0.2424	0.7273
all-MiniLM-L6-v2 [Wang et al. 2020]	384	0.6475	0.2485	0.7455
all-mpnet-base-v2 [Song et al. 2020]	768	0.6111	0.2384	0.7152
<i>k = 5</i>				
jina-embeddings-v2-base-en [Günther et al. 2023]	768	0.6772	0.1612	0.8061
e5-base-v2 [Wang et al. 2022]	768	0.7065	0.1648	0.8242
nomic-embed-text-v1.5 [Nussbaum et al. 2024]	768	0.7245	0.1661	0.8303
all-MiniLM-L12-v2 [Wang et al. 2020]	384	0.6198	0.1539	0.7697
bge-small-en-v1.5 [Xiao et al. 2023]	384	0.6688	0.1636	0.8182
all-MiniLM-L6-v2 [Wang et al. 2020]	384	0.6626	0.1624	0.8121
all-mpnet-base-v2 [Song et al. 2020]	768	0.6247	0.1552	0.7758
<i>k = 10</i>				
jina-embeddings-v2-base-en [Günther et al. 2023]	768	0.6861	0.0879	0.8788
e5-base-v2 [Wang et al. 2022]	768	0.7132	0.0879	0.8788
nomic-embed-text-v1.5 [Nussbaum et al. 2024]	768	0.7344	0.0903	0.9030
all-MiniLM-L12-v2 [Wang et al. 2020]	384	0.6321	0.0867	0.8667
bge-small-en-v1.5 [Xiao et al. 2023]	384	0.6823	0.0915	0.9152
all-MiniLM-L6-v2 [Wang et al. 2020]	384	0.6727	0.0885	0.8848
all-mpnet-base-v2 [Song et al. 2020]	768	0.6388	0.0885	0.8848

Table 2 shows the results for the evaluated models. `bge-small-en-v1.5` obtained the highest Recall@10, reaching 0.915, followed closely by `nomic-embed-text-v1.5` with 0.903. In terms of MRR@k, `nomic-embed-text-v1.5` performed best, with a score of 0.734 at $k = 10$, while `e5-base-v2` achieved the second-highest value, 0.713. Precision@k showed relatively small differences across models, particularly at larger values of k , with `bge-small-en-v1.5` achieving the highest Precision@10. Overall, the results indicate that smaller 384-dimensional models, such as `bge-small-en-v1.5` and `all-MiniLM-L6-v2`, can perform competitively with 768-dimensional models as the number of chunks increases.

4.3. Generation Evaluation

To assess the quality of the generated responses, we conducted a quantitative evaluation using automatic metrics from the RAGAS framework [Es et al. 2024]. The evaluation focused on the internal quality of the pipeline, assessing how faithfully the generative model synthesizes an answer based on the context retrieved.

A test set of 42 queries was manually constructed from the close reading of a representative subset of papers in the corpus, chosen based on layout diversity. For each query, a reference answer was composed using excerpts from the source articles, providing a ground-truth baseline for evaluation. Two metrics from the RAGAS framework were selected for this evaluation.

Answer Relevancy measures the degree to which the generated response directly addresses the input query. Given a generated answer a , N synthetic questions $\{q_1, q_2, \dots, q_N\}$ are derived from a by an LLM (gpt4o-mini[Hurst et al. 2024]). The metric is then computed as the mean cosine similarity between the embedding of the original question q and each generated question:

$$\text{Answer Relevancy}(q, a) = \frac{1}{N} \sum_{i=1}^N \text{sim}(\mathbf{e}_q, \mathbf{e}_{q_i}) \quad (1)$$

where $\mathbf{e}_q$ and $\mathbf{e}_{q_i}$ denote the embedding vectors of q and q_i , respectively. In our experiments, we set $N = 3$, meaning that three synthetic questions were generated for each answer. A high score indicates that the generated answer is likely semantically relevant to the user's query.

Faithfulness quantifies the extent to which the generated response is grounded in the retrieved context. An LLM judge first decomposes the answer a into a set of atomic claims $\{c_1, c_2, \dots, c_M\}$. Each claim is then verified against the retrieved context C , and the metric is defined as the proportion of supported claims:

$$\text{Faithfulness}(a, C) = \frac{|V|}{M} \quad (2)$$

where $V \subseteq \{c_1, \dots, c_M\}$ is the subset of claims supported by C , and M is the total number of claims extracted from the generated answer a . This is the primary metric for our system, as it directly penalizes claims that cannot be supported by the provided chunks.

Six open-weights instruction-tuned models of varying parameter size were evaluated as candidates for the generation module, the retriever used was bge-small-en-v1.5, with $k = 10$ since it presented the best performance during retrieval. The results are presented in Table 3.

The results show that the three best-performing models in terms of Faithfulness (qwen-2.5-7b, llama-3.1-8b, and mistral-8b) all exceeded 0.90, while the remaining three scored below 0.89. Answer Relevancy follows a similar trend, with llama-3.1-8b achieving the highest score and mistral-small-3.1-24b the lowest. Overall, the results indicate that smaller models can achieve both strong factual grounding and high query relevance, suggesting that generation quality is not solely determined by model size.

Table 3. Generation quality evaluation using the RAGAS framework.

Model	Params	Answer Relevancy	Faithfulness
Llama-3.1-8B [Grattafiori et al. 2024]	8B	0.8417	0.9020
Qwen-2.5-7B [Qwen Team 2024]	7B	0.8332	0.9259
Ministral-8B [Liu et al. 2026]	8B	0.7813	0.9007
Qwen3.5-27B [Qwen 2025]	27B	0.7934	0.8873
GPT-oss-20B [OpenAI 2025]	20B	0.7842	0.8822
Mistral-small-3.1-24b [Mistral AI 2025]	24B	0.7692	0.8251

4.4. Prototype app

To validate the technical integration of the proposed modules and provide an interface for educators, we developed a functional prototype that connects the retrieval and generation pipeline to a mobile application. The implementation was organized into two main components: (i) a backend service responsible for executing the RAG pipeline, and (ii) an Android application responsible for user interaction.

The backend was implemented with FastAPI³ and is responsible for orchestrating the retrieval and generation pipeline. Upon receiving a user query, the backend encodes the query into embeddings, retrieves the most relevant chunks from the vector index, and injects the retrieved passages into the prompt sent to the LLM, and the generated answer is then returned through a REST API consumed by the mobile application. Each response also includes inline source citations containing the title and publication year of the referenced papers, allowing users to identify the academic sources supporting the generated content. In addition to the inference pipeline, the backend is also responsible for managing the indexed document collection and exposing the metadata associated with the retrieved papers.

The client application was built with Jetpack Compose⁴ and provides a means for interacting with the system. Through a chat interface, educators can submit natural language queries and receive responses grounded in the retrieved academic literature. The interaction flow was designed to resemble traditional conversational AI systems in order to reduce operational complexity and keep the focus on the retrieved content. Alongside the conversational interface, the application also includes a library section containing the latest papers incorporated by the Search Agent, allowing educators to inspect the sources available in the knowledge base and view related metadata, such as title and publication year, this functionality is intended to improve transparency regarding the materials used by the system.

By deploying the system on a mobile platform, we aim to demonstrate the viability of the proposed pipeline in a practical scenario. The app functionality is illustrated in Figure 3.

³<https://fastapi.tiangolo.com>

⁴<https://developer.android.com/compose>

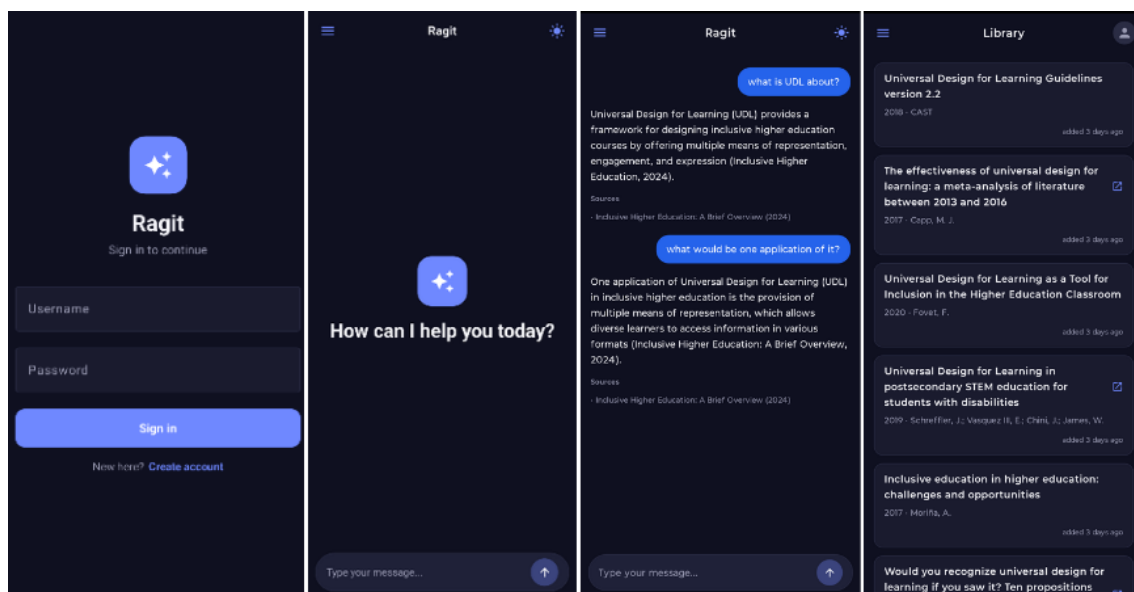


Figure 3. Android application. When the user inputs a query, the app transmits the request to the system backend and displays its outcomes.

5. Final considerations

This work presented RAGIT, a retrieval-augmented system that supports educators in obtaining guidance on inclusive education literature in higher education. The proposed solution comprises an automated document acquisition pipeline, a VLM-based document ingestion process that preserves the structure of academic papers, a retrieval and generation pipeline, and a mobile application through which educators can interact with the system. Experimental results showed that the proposed approach achieved a Recall@10 of 0.91 in retrieval and a Faithfulness score of 0.92 in generation, indicating that relevant sources can be retrieved with high coverage and used to generate responses that remain grounded in the retrieved evidence.

Our work contributes a curated corpus of studies on inclusive higher education, assembled according to explicit selection criteria and designed to be continuously expanded as new research becomes available. The proposed solution further demonstrates how RAG can be combined with automated literature acquisition and VLM-based document processing to facilitate access to reliable educational content from the literature. Finally, the developed mobile application demonstrates the practical feasibility of deploying the proposed approach in real-world educational contexts.

One limitation of our work should be acknowledged. The evaluation was conducted on a relatively small document collection, which may favor recall, as performance may degrade on larger corpora. As future work, we intend to evaluate the system on larger document collections, investigate the impact of the automated Search Agent on corpus expansion, and conduct user studies with higher education educators to assess the effectiveness of the proposed solution in a realistic scenario.

AI Transparency

Figures 2 and 1 were generated using Gemini-3-Pro-Image. The text was grammatically revised by Gemini-3-Pro and reviewed by the authors to ensure text accuracy.

Code Availability

We provide the complete source code for the RAG pipeline, the ingestion module, and the Android application, along with the documents chunks used in this work at <https://github.com/h4rad4/sbie2026>.

References

- dos Santos, A. B., Santos, R. E. d. S., and da Silva, C. M. R. (2025). Teachat: Promovendo o suporte educacional ao autismo com inteligência artificial conversacional. In *Workshop sobre Educação em Computação (WEI)*, pages 1369–1379. SBC.
- Douze, M., Guzhva, A., Deng, C., Johnson, J., Szilvasy, G., Mazaré, P.-E., Lomeli, M., Hosseini, L., and Jégou, H. (2026). The faiss library. *IEEE Transactions on Big Data*, 12(2):346–361.
- Es, S., James, J., Anke, L. E., and Schockaert, S. (2024). 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*, pages 150–158.
- Galfre, M., Curcio, V., Dardouillet, P., and Le Van, O. (2025). Vision-language models for sick leave certificates: Beyond ocr in real-world form understanding. In *2025 3rd International Conference on Foundation and Large Language Models (FLLM)*, pages 254–261. IEEE.
- Grattafiori, A., Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Vaughan, A., et al. (2024). The llama 3 herd of models. Technical report, Meta AI. [arXiv:2407.21783](https://arxiv.org/abs/2407.21783).
- Günther, M., Ong, J., Mohr, I., Abdessalem, A., Abel, T., Akram, M. K., Guzman, S., Mastrapas, G., Sturua, S., Wang, B., et al. (2023). Jina embeddings 2: 8192-token general-purpose text embeddings for long documents. *arXiv preprint arXiv:2310.19923*.
- Hennekeuser, D., Vaziri, D. D., Golchinfar, D., Schreiber, D., and Stevens, G. (2025). Enlarged education—exploring the use of generative ai to support lecturing in higher education. *International Journal of Artificial Intelligence in Education*, 35(3):1096–1128.
- Hurst, A., Lerer, A., Goucher, A. P., Perelman, A., Ramesh, A., Clark, A., Ostrow, A., Welihinda, A., Hayes, A., Radford, A., et al. (2024). Gpt-4o system card. Technical Report [arXiv:2410.21276](https://arxiv.org/abs/2410.21276), OpenAI.
- Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., et al. (2020). Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474.
- Liu, A. H., Khandelwal, K., Subramanian, S., Jouault, V., Rastogi, A., Sadé, A., Jeffares, A., Jiang, A., Cahill, A., Gavaudan, A., et al. (2026). Ministral 3. *arXiv preprint arXiv:2601.08584*.

- Lopez-Gazpio, I. (2025). Integrating large language models into accessible and inclusive education: access democratization and individualized learning enhancement supported by generative artificial intelligence. *Information*, 16(6):473.
- Matos, F. D., Giroto Junior, G., de Medeiros Arnt, A., and Lippi, A. (2024). A proposal in brazil to use generative ai in education threatens quality and equity. *Patterns*, 5(7):101020.
- Mistral AI (2025). Mistral-small-3.1-24b-instruct-2503 model card. <https://huggingface.co/mistralai/Mistral-Small-3.1-24B-Instruct-2503>. Acesso em: 22 mai. 2026.
- Nussbaum, Z., Morris, J. X., Duderstadt, B., and Mulyar, A. (2024). Nomic embed: Training a reproducible long context text embedder.
- OpenAI (2025). gpt-oss-120b & gpt-oss-20b model card. Model card, OpenAI. arXiv:2508.10925.
- Pan, F., Zhou, Q., Guo, W., and Yang, H. (2025). A survey on retrieval-augmented generation in applications of education and teaching. In *2025 7th International Conference on Computer Science and Technologies in Education (CSTE)*, pages 803–807. IEEE.
- Qwen (2025). Qwen3.5-27b model card. <https://huggingface.co/Qwen/Qwen3.5-27B>. Model card. Acesso em: 22 mai. 2026.
- Qwen Team (2024). Qwen2.5 technical report. Technical report, Alibaba Group. arXiv:2412.15115.
- Reis-Andersson, J. (2024). Ai generative in brazil’s public schools: The teachers’ perspective. *International Conference on AI Research*, 4:374–380.
- Song, K., Tan, X., Qin, T., Lu, J., and Liu, T.-Y. (2020). Mpnet: Masked and permuted pre-training for language understanding. In *Advances in Neural Information Processing Systems*, volume 33, pages 16857–16867.
- UNDESA (2019). *Realizing the Sustainable Development Goals by, for and with Persons with Disabilities*. Stylus Publishing, LLC.
- UNDESA (2025). *Accelerating the Realization of the Sustainable Development Goals by, for and with Persons with Disabilities*. Stylus Publishing, LLC.
- United Nations (2015). Transforming our world: the 2030 agenda for sustainable development. Acesso em: 22 maio 2026.
- Wang, L., Yang, N., Huang, X., Jiao, B., Yang, L., Jiang, D., Majumder, R., and Wei, F. (2022). Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*.
- Wang, W., Wei, F., Dong, L., Bao, H., Yang, N., and Zhou, M. (2020). Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *arXiv*.
- Xiao, S., Liu, Z., Zhang, P., and Muennighoff, N. (2023). C-pack: Packaged resources to advance general chinese embedding.
- Yu, S., Tang, C., Xu, B., Cui, J., Ran, J., Yan, Y., Liu, Z., Wang, S., Han, X., Liu, Z., et al. (2025). Visrag: Vision-based retrieval-augmented generation on multi-modality documents. In *International Conference on Learning Representations*, volume 2025.