Journal on Interactive Systems, 2026, 17:1
ISSN: 2763-7719 • doi: 10.5753/jis.2026.8133

Research Paper

Evaluating Incremental Context Strategy in RAG-Based Chatbots for Hospital Document Retrieval: A Trade-off Between Quality and Cost

Murilo Vargas da Cunha [Federal University of Pelotas & Federal Institute of Rio Grande do Sul | *mvcunha@inf.ufpel.edu.br*]
Marilia Rosa Silveira [Federal University of Pelotas | *mrsilveira@inf.ufpel.edu.br*]
Brenda Salenave Santana [Federal University of Pelotas | *bssalenave@inf.ufpel.edu.br*]
Larissa Astrogildo Freitas [Federal University of Pelotas | *larissa@inf.ufpel.edu.br*]
Ulisses Brisolara Corrêa [Federal University of Pelotas | *ulisses@inf.ufpel.edu.br*]

Center for Technological Development, Federal University of Pelotas, Rua Gomes Carneiro, 01, Balsa, Pelotas, RS, 96010-610, Brazil.

Abstract.
Applications in the healthcare field where access to contextualized information anchored in reliable documents has been crucial have increasingly driven the adoption of Retrieval-Augmented Generation (RAG) systems. In this context, this work presents the development and evaluation of an RAG-based chatbot for retrieving normative documents from a university hospital. Traditional RAG approaches typically rely on fixed-size contexts, which may lead to excessive token consumption and degradation in response quality due to the inclusion of irrelevant information. Building upon a previously optimized retrieval pipeline based on the *intfloat/multilingual-e5-small* embedding model and Reciprocal Rank Fusion (RRF), this study investigates a novel incremental context strategy that progressively expands the input to the language model's during response generation. Experiments were conducted on a hybrid dataset of 192 question-answer pairs, combining synthetic queries and expert-generated questions. Four large language models (Gemini Flash 2.5, GPT-4.1-mini, Sabiá 3.1, and DeepSeek 3.2) were evaluated under fixed and incremental context settings, using BERTScore, LLM-as-a-Judge metrics, and token consumption as evaluation criteria. The results show that the incremental approach significantly reduces token usage, achieving up to 88.68% savings, while maintaining or improving semantic quality in most cases. These findings demonstrate that incremental context construction is an effective strategy for improving efficiency and response quality in RAG-based chatbots, while mitigating context dilution effects. The proposed approach is model-agnostic and readily applicable to real-world scenarios, particularly in resource-sensitive and high-stakes domains such as healthcare.

Keywords: Retrieval-Augmented Generation, Chatbots, Large Language Models, Incremental Context, Token Efficiency

Edited by: Windson Viana de Carvalho | **Received:** 01 May 2026 • **Accepted:** 04 September 2026 • **Published:** 08 September 2026

1 Introduction

Recently, the way conversational systems are developed and implemented has undergone significant changes with the advent of Large Language Models (LLMs). According to Cederlund *et al*. [2024], with the advent of LLMs, a significant potential for evolution has emerged in digital support services, which traditionally operated in a conventional manner. For Zhu *et al*. [2024], driven by this rapid technological advancement, models such as GPT-3 and BERT have gained prominence in multiple Natural Language Processing (NLP) tasks. However, despite their promising performance, LLMs still face challenges in generating inaccurate, generic, or factually incorrect responses, known as hallucinations Ji *et al*. [2023b]. This phenomenon can lead to outdated information, misinterpretations, or gaps in highly specialized areas that directly affect the safety of using these systems in sensitive contexts, such as the hospital sector [Zhu *et al*., 2024; Abo El-Enen *et al*., 2025].

In addition, Ji *et al*. [2023a] highlights that hallucination in NLP is particularly problematic, as it impairs performance and raises safety concerns in real-world applications. For example, in medical applications, a hallucinatory summary generated from a patient information form could pose a life-threatening risk if the automatically generated medication instructions are hallucinatory.

In this scenario, users engaged in dialogues with LLMs occasionally encounter instances of misinformation or incorrect information presented in such a persuasive manner as if it were fact [Perković *et al*., 2024]. As emphasized by Joshi *et al*. [2024], there is evidence that these LLMs are inadvertently prone to hallucinations, degrading system performance and failing to produce accurate results in generative AI.

Seeking to overcome these limitations in coverage and accuracy of LLMs, traditional information retrieval techniques, previously employed mainly in search engines, have been integrated into generative models, resulting in more robust and effective architectures for reading and understanding documents [Raza *et al*., 2025]. This approach is called Retrieval-Augmented Generation (RAG), which combines the generative capabilities of LLMs with information retrieval mechanisms from external knowledge bases [Ayala and Bechard, 2024; Yu *et al*., 2025].

However, this strategy applied in LLMs also presents additional limitations, especially related to processing large volumes of context, where the efficient incorporation of external knowledge during inference in long or dynamic contexts remains a challenge [Taguchi *et al*., 2025]. In this sense, fixed-

context approaches tend to retrieve a predefined number of segments or chunks and concatenate them into a single input for the language model, which can introduce irrelevant or conflicting information. Furthermore, this strategy does not guarantee that the language model will effectively utilize the retrieved data during generation, resulting in responses that may still be inaccurate or inconsistent [Asai *et al*., 2023]. Thus, the use of a fixed context may become inadequate in scenarios with extensive and heterogeneous documentary databases.

In order to mitigate this limitation, this work proposes an incremental context-building strategy method, in which the retrieved fragments are progressively provided to the LLM, following the order of relevance. In this way, the process begins with a small subset (Top-5) and is iteratively expanded only when necessary, being interrupted as soon as a valid response is obtained. Therefore, unlike the traditional fixed-context approach, this strategy avoids the cumulative sending of large volumes of information, seeking to improve the effective use of context and reduce the interference of irrelevant segments. Furthermore, the method does not require fine-tuning of the models or modifications to the LLM architecture, being directly applicable to different models and scenarios.

In this context, this article presents a revised and expanded version of the work [da Cunha *et al*., 2025], which describes the development and evaluation of a chatbot based on RAG, exploring an incremental context-building strategy applied to normative documents from a university hospital, composed of Standard Operating Procedures (SOPs) and technical manuals in Portuguese. Thus, this study expands and complements the previous work, as it expands the dataset with question-answer pairs created by experts and emphasizes the evaluation of the generation stage, comparing multiple language models under two context usage strategies: fixed and incremental. In this sense, the evaluation considers not only the quality of the responses through metrics such as BERTScore and LLM-as-a-Judge, but also the computational cost, measured by token consumption. Therefore, token consumption serves as an indicator of computational and financial costs, particularly in API-based LLM use cases. Thus, this work contributes to understanding the impacts of context size and selection on generation quality, as well as mitigating effects such as context dilution in real-world applications in the healthcare domain.

The main contributions of this work are:

- Proposal of an incremental context construction strategy for RAG-based chatbots;
- Evaluation of multiple LLMs under fixed and incremental context settings;
- Demonstration of significant token reduction (up to 88%) without compromising response quality;
- Analysis of context dilution effects in large context scenarios;
- A multi-dimensional evaluation framework combining semantic, qualitative, and cost-based metrics.

The remainder of this paper is organized as follows. Section 2 presents the related work, discussing prior studies positioning this study within the context of chatbots, RAG and LLMS. Section 3 introduces the theoretical foundations, covering LLMs and RAG. Section 4 describes the proposed methodology, including dataset construction, retrieval and re-ranking strategies, and the incremental context approach. Section 5 presents and discusses the experimental results, with emphasis on the trade-off between response quality and token consumption. Section 6 outlines the limitations of this study, and Section 7 concludes the paper with final remarks and directions for future research.

2 Related Works

This section positions the present study with respect to its target problem: the inefficient construction of context in fixed-context RAG systems, which retrieve and concatenate a predefined number of chunks into a single input. This strategy may introduce irrelevant or conflicting information (context dilution), increase token cost, and does not guarantee that the retrieved information is effectively used during generation. Building on this problem, we prioritize works that explore mechanisms for retrieving and generating document-grounded answers, with attention to efficiency, quality, and the context-construction strategy.

To select the related works, a search was carried out in the IEEE Xplore, Scopus, and Google Scholar databases, covering the period 2024–2025. The search combined the concept of retrieval-augmented generation with document-grounded conversational/question-answering systems, using the query: ("retrieval-augmented generation" OR "RAG") AND (chatbot OR "question answering" OR "document retrieval" OR "information access"). The results were then screened according to two inclusion criteria: (i) the work applies RAG to retrieval/generation over documents; and (ii) it discusses the efficiency or the quality of context construction. We excluded works restricted to the design or usability evaluation of chatbots that do not address document-based retrieval. The initial set was complemented by backward/forward citation chaining (snowballing) from the selected studies.

In this context, when considering more recent approaches based on RAG and integration with LLM models, the following works stand out, exploring mechanisms for retrieving and generating answers based on documents. The study by Obaid and Bawany [2024] presents SeerahGPT, a system built on the Llama-2-7b architecture that employs RAG to retrieve data from a corpus of Islamic texts (Quran and Hadith). This work applies the metrics Mean Average Precision (MAP) and Mean Reciprocal Rank (MRR) to evaluate the retrieval system and uses others such as Bilingual Evaluation Understudy (BLEU), Recall-Oriented Understudy for Gisting Evaluation (ROUGE), and Metric for Evaluation of Translation with Explicit Ordering (METEOR) to assess response generation. As a result, SeerahGPT outperformed the baseline Llama-2-7b model with a BLEU score of 0.132 (vs. 0.130), as well as superiority in the ROUGE-L (0.230 vs. 0.149) and METEOR (0.259 vs. 0.229) metrics. These results demonstrate that dynamic recovery mitigates hallucinations inherent to LLMs.

While the authors of Wijaya and Purwarianti [2024] propose a tutoring system for programming, based on documents originally written in Indonesian, they compared several open-source LLMs, including models from the Llama, Gemma,

DeepSeek, and Mistral families, combined with embedding models such as e5-large-v2, jina-embeddings-v2, and gte-en-v1.5. The evaluation was conducted qualitatively, in addition to using the MAP metric to assess document retrieval at different chunking sizes.

Following the same line of evaluation of the retrieval stage, the study by Nai *et al*. [2024] employs NDCG@10 to perform an extensive comparison between multiple *embedding* models, including BM25, Cohere, and several OpenAI models. The work by Maryamah *et al*. [2024] uses other ranking metrics and opts for an evaluation based on classic recall and precision metrics to compare its embedding models. Despite the diversity of metrics and analytical focuses, a common point among these works is the absence of a re-ranking layer.

In healthcare applications, the use of RAG has also been extensively investigated as a strategy to mitigate hallucinations. In hospital settings, the work of Son *et al*. [2025] proposes a RAG chatbot to assist in operational queries in Electronic Patient Record (EPR) systems, employing contrast-learning-tuned multilingual embeddings and a commercial LLM via API. The results demonstrate high retrieval performance, with Top-K accuracy exceeding 97%, with the analysis focusing primarily on the retrieval stage.

In the context of patient education, the study by Baur *et al*. [2025] presents a German-language RAG chatbot for orthopedics and traumatology, built from validated clinical guidelines and educational materials. The system combines semantic search with the Qdrant vector database and generation with OpenAI's GPT (Global Path of Technology), being evaluated by automatic metrics and user studies, which indicate high acceptance and perceived quality. Another work that explores languages other than English is that of [Zhang *et al*., 2025], in which the authors integrate RAG with medical knowledge graphs for clinical question-and-answer systems in Chinese, demonstrating consistent gains in accuracy in clinical benchmarks. However, these works reinforce the practical applicability of RAG, without exploring adaptive mechanisms for context control and token use.

In contrast, the present work adopts an approach based on LLM models integrated with information retrieval mechanisms through RAG, using context-building strategies, analyzing their impact on response quality and computational cost through the use of tokens.

3 Theoretical Foundations

In this section, to provide a better foundation for the topics covered in this work, we describe theories and concepts about LLMs and RAG.

3.1 Large Language Models - LLMs

LLMs are typically deep neural networks with a large number of parameters, trained on large volumes of textual data, sometimes encompassing large portions of all publicly available text on the internet. Models like this usually have tens or even hundreds of billions of parameters, which are the adjustable weights in the network that are optimized during training to predict the next word in a sequence [Raschka, 2024].

Thus, these LLMs possess an enormous number of parameters, typically in the billions, with examples such as GPT-3 at 175 billion and PaLM at 540 billion [Zhao *et al*., 2023]. In addition to fitting the definition of Generative AI focused on text generation, this massive scaling causes them to exhibit different behaviors from smaller models, demonstrating emergent abilities and capabilities to solve complex tasks that are not observed in smaller-scale models [Wei *et al*., 2022].

One emerging ability in LLMs occurs when the model receives instructions and pairs of input/output examples to perform a task. This approach is known as in-context learning (or few-shot prompt) [Brown *et al*., 2020]. What defines this capability is the possibility of using LLMs without any additional training that updates their parameters through gradient optimization. Instead of this specific adjustment, the models can leverage their pre-training and be used with natural language instructions, prompts, and/or task demonstrations provided as examples. In this way, the models curiously tackle tasks for which they were not explicitly trained. It is in this scenario that RAG emerges, feeding the LLM prompts with external content not used in the model's training.

3.2 Retrieval-Augmented Generation - RAG

Within the context of LLMs, RAG has numerous applications, including, for example, dialogue generation, question answering, and summarization [Gummadi *et al*., 2024]. The RAG model typically consists of two main components [Devi *et al*., 2024]:

- **Retrieval Component:** This part of the model is responsible for retrieving relevant information from a large corpus of documents or knowledge sources.
- **Generation Component:** Once the relevant information has been retrieved, the generation component produces a response or output in natural language.

The retrieval stage can employ different types of retrieval mechanisms and levels of information granularity. Retrieval mechanisms can be sparse, based on term matching methods such as TF-IDF and BM25, which are simple and do not require training, but are limited to lexical similarity [Ramos *et al*., 2003], or dense retrieval mechanisms, which use model embeddings such as BERT, bi-encoders and pre-trained retrieval mechanisms, allowing for a more flexible and adaptable semantic search [Reimers and Gurevych, 2019].

Embeddings are dense vector representations that capture the semantic information of unstructured text, which can overcome problems such as the curse of dimensionality and the lack of syntactic and semantic information in [Kalyan and Sangeetha, 2020] representations. Thus, the system must measure the similarity between embeddings to retrieve the most relevant vectors from a vector database. This metric can be a function that receives two vectors and returns a real value, indicating the similarity or distance.

However, even though embeddings capture semantic similarity, they usually have difficulties with long texts and lose domain-specific details [Zhou *et al*., 2024]. Therefore, many experiments adopt hybrid approaches that combine sparse and dense retrieval techniques.

The granularity of retrieval defines the unit of information, which can be the entire document, chunks, individual

tokens, or entities/mentions. Chunk-level retrieval is the most common in current RAG systems due to its balance between efficiency and relevance, while other granularities are used for specific purposes, such as searching for rare patterns (tokens) or knowledge-centric tasks (entities) [Fan *et al.*, 2024].

Chunking is the process of dividing information into smaller, semantically coherent units called "chunks," thus determining whether responses are contextually coherent or fragmented, directly shaping what the RAG retrieves. Effective chunking ensures that these segments are meaningful enough to provide context, yet concise enough to fit within the LLM context window [Gomez-Cabello *et al.*, 2025].

The generation stage in RAG systems is intrinsically linked to the subsequent retrieval task and predominantly uses two types of generator models. Generators with accessible parameters, such as encoder-decoder architectures, for example, T5 and BART, and decoder-only models, such as open-source models from the GPT family, which allow fine-tuning and directly integrate retrieved documents to improve generation, as well as proprietary language models, for example, GPT-4 and Claude, which do not allow adjustment of their weights [Raschka, 2024]. In these cases, optimization focuses on prompt engineering, enriching them with retrieved context through techniques such as contextual learning and document compression to improve the efficiency and quality of the response [Fan *et al.*, 2024].

3.2.1 Structure of a Basic RAG System

The operation of a basic RAG system with vector retrieval is based on the dynamic integration of external knowledge, starting with the conversion of the database into chunks which are, in turn, transformed into embeddings and stored in a vector database. This operation can be described as follows [Lewis *et al.*, 2020; Oliveira *et al.*, 2025]:

1. **Pre-processing:** Documents are divided into parts Chunks and transformed into embeddings.
2. **Indexing:** Embeddings are indexed to allow for efficient retrieval, which is essential for RAG systems. In some cases, vector databases and Approximate Nearest Neighbor (ANN) indexing are used, although they are not strictly necessary.
3. **Query:** A user's question is first converted into an embedding. In some cases, query transformation or expansion techniques are applied, such as rephrasing or splitting the question into multiple subqueries.
4. **Retrieval:** The query embedding is used to retrieve semantically similar chunks, typically from a vector database. This is usually followed by a reclassification step to reorder the retrieved results based on more refined relevance criteria.
5. **Generation:** The retrieved fragments are combined with the user's original query to form a prompt and passed to the LLM as context to generate the final response.

This pipeline allows the generative model to access an external knowledge base without the need for retraining, which is one of the main advantages of the RAG architecture. The flowchart of the Basic RAG Pipeline is shown in Figure 1.

In this pipeline, the RAG system indexes the entire collection of documents that make up the knowledge base. This

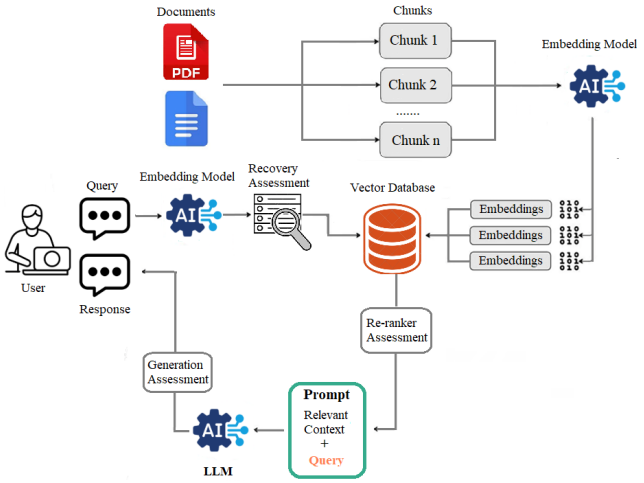

Figure 1. Basic RAG Pipeline Flowchart.

indexing involves dividing the document text into several chunks, which are converted into embeddings and stored in a vector database. Through a retrieval process, the system identifies the chunks most relevant to the user's question by vector similarity, usually using cosine similarity. Then, the system combines the context of the original question with the information extracted from the relevant documents, forming a prompt that is sent to the LLM. This prompt enriches the LLM with contextual information, allowing it to generate a more accurate and complete answer. Thus, relevance is determined at query time by the retrieval stage, and not assumed at indexing.

Therefore, by incorporating external data, the RAG system significantly expands the knowledge base of LLMs. In this way, this technique not only makes them more capable of addressing complex and current issues, but also gives them the ability to access and process information in a much more dynamic way than traditional models.

Given this context, research in RAG began to focus on providing better information so that LLMs could respond to more complex tasks requiring greater knowledge during the inference phase, leading to a rapid development in RAG studies [Gao *et al.*, 2023].

4 Methodology

This section describes the methodology adopted for the development and evaluation of the chatbot in this work, which advances to the use of new techniques beyond basic RAG. The methodology was structured in sequential steps, allowing for the separate evaluation of information retrieval, response generation, and the proposed strategy for adaptive context control. The complete flowchart of the adaptive context control technique, which we call Incremental Retrieval, is presented in Figure 2.

Given the user's question, the system retrieves and ranks candidate chunks by combining BM25 and e5 embeddings via RRF. Next, the iteration counter is initialized to zero and the initial position is set to the first block of five chunks (Top-5). The chunk block at the current position is obtained and sent, together with the question, to the LLM, which processes the context and generates an answer. The answer is then subjected to the stopping criterion: if it is considered valid — that is, if

the model produces an answer without signaling the absence of information in the context — it is returned; otherwise, the system checks whether the maximum number of iterations has been reached. If not, the counter is incremented, the next block of five chunks is appended to the context, and the process returns to the retrieval step; if so, the system returns the indication that no valid answer was found. Finally, the generated answers are submitted to the evaluation stage, which considers BERTScore, the number of tokens consumed, and LLM-as-a-Judge.

4.1 Dataset

This study expands on the previous research [da Cunha *et al.*, 2025] by adding 118 questions/answers created by experts. This forms a hybrid set of questions and answers, totaling 192 instances, divided into:

- **Synthetic dataset(derived from previous work):** composed of 74 questions/answers automatically generated with Gemini Flash 1.5, and manual identification of the identifier (ID) of the chunk containing the correct answer in the database. This subset allows for an objective and controlled evaluation of the retrieval step;
- **Expert dataset:** composed of 118 questions/answers developed by hospital professionals in a preliminary qualitative evaluation phase of the system.

This expansion aims to increase the realism and complexity of the queries, bringing the experimental scenario closer to real-world applications. Table 1 presents representative examples of questions and answers that illustrate the linguistic structure and informational granularity addressed in the evaluation.

The construction of the synthetic dataset and the preparation of the corpus for indexing in the previous study aimed to enable controlled testing of the retrieval and reordering modules before evaluation in a real-world scenario, as well as to establish a traceable link between each question and its documentary evidence, necessary for the ranking metrics used in the retrieval evaluation stage.

To assemble the current dataset, 107 internal normative documents were selected from five hospital departments: Occupational Health and Safety Unit, Human Resources Division, Hospital Infection Control Service, Teaching and Research Management Department, and Information Technology and Digital Health Sector [Empresa Brasileira de Serviços Hospitalares (EBSERH), 2025]. These documents were chosen because they represent frequent institutional routines and present normative language (rules, exceptions, and procedures) aligned with the target scenario of the RAG system.

The documents were segmented into smaller units (chunks) for indexing in the vector database. We adopted chunks of approximately 100 tokens, applying a soft limit: upon reaching this value, the cut was moved to the next endpoint, preserving syntactic cohesion. This configuration proved more suitable than alternatives based on line or page division, and also approximated the length of a paragraph, often sufficient to contain the necessary evidence in normative documents. To increase traceability, we also stored the full text of the source page of each chunk in a dedicated field in the database. We acknowledge that contextual/semantic segmentation strategies, such as the hybrid chunker of the docling library, tend to outperform purely structural segmentation by better preserving semantic boundaries; their comparative evaluation in this domain is indicated as future work.

4.1.1 Questions and Answers Generated by Experts

After selecting the embeddings model and the reordering strategy in the quantitative phase of the study, we configured the system for evaluation in a usage scenario. In this phase, we used Gemini Flash 2.0 as the generator model and submitted the chatbot to evaluation by 12 hospital specialists, with the aim of verifying its perceived usefulness and collecting real interactions to expand the initial dataset.

Each specialist formulated questions related to their routine and evaluated the answer as satisfactory ("Yes") or unsatisfactory ("No"). When the answer was rejected, the system requested the "expected answer," allowing for the recording of a reference question/answer pair and supporting failure analysis. In the end, we collected 139 interactions.

In absolute terms, 118 responses were classified as "Satisfactory" (Yes), indicating that the system was able to retrieve and synthesize the correct information in the vast majority of cases presented. Therefore, these 118 questions and answers approved by experts were incorporated into the synthetic dataset to compose a hybrid dataset, totaling 192 question-answer pairs used in the comparative evaluation of generative models.

4.2 Evaluation of Information Retrieval from Embeddings and Reordering Models

As previously described, the retrieval assessment stage was conducted in the [da Cunha *et al.*, 2025] study, in which three embedding models were analyzed: *all-MiniLM-L6-v2*, *paraphrase-multilingual-MiniLM-L12-v2*, and *intfloat/multilingual-e5-small*. These models were used to represent questions and *chunks* in dense vectors, enabling semantic retrieval via cosine similarity Kalyan and Sangeetha [2020]. In addition to the dense retrievers, a sparse retriever based on BM25 was employed, producing a lexical ranking to be combined with the dense ranking in the fusion step.

The quality of the ranking was evaluated using the metrics Mean Reciprocal Rank (MRR) and Normalized Discounted Cumulative Gain (NDCG@10), considering as relevant the chunk previously identified as containing evidence of the response. For each of the 74 questions in the synthetic dataset, the 100 most similar chunks were retrieved, ensuring sufficient coverage for further refinement.

Due to the negative impact of using large volumes of context, both in terms of token cost and response quality, a reordering step was introduced to prioritize the most relevant chunks.

The strategies evaluated were: (i) the Cross-Encoder model, which calculates joint relevance between the question and the chunk Puthenputhussery *et al.* [2025]; (ii) Reciprocal Rank Fusion (RRF), which combines rankings from different methods Cormack *et al.* [2009]; and (iii) LLM-based reordering, using Gemini Flash 1.5 Carraro and Bridge [2025].

The results showed that, after reordering, the differ-

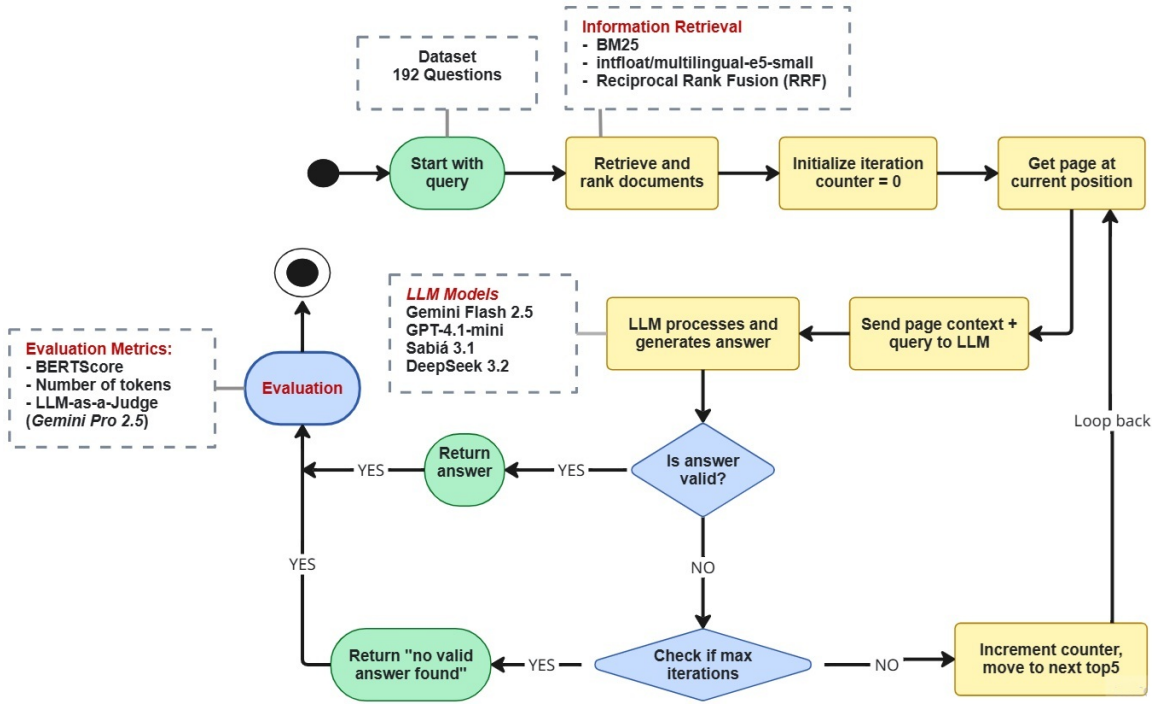

Figure 2. Flowchart of the proposed incremental retrieval pipeline.

Table 1. Example questions and answers, with their origin (Synthetic = automatically generated; Expert = authored by a hospital professional).

Question	Answer	Origin
Quais escalas são utilizadas para avaliar o risco de quedas em pacientes adultos e crianças, respectivamente, e como são identificados os pacientes adultos com risco alto de quedas? *(What scales are used to assess the risk of falls in adult and child patients, respectively, and how are adult patients at high risk of falls identified?)*	Para crianças é utilizada a Escala Humpty Dumpty e para os adultos a Morse Fall Scale (MSF). Os pacientes adultos, que apresentem Risco Alto para quedas são identificados com uma pulseira amarela. *(The Humpty Dumpty Scale is used for children and the Morse Fall Scale (MSF) for adults. Adult patients at high risk for falls are identified with a yellow bracelet.)*	Synthetic
De acordo com o documento, qual a solução recomendada para a nebulização em caso de coleta de escarro induzido, especificando a composição e advertindo sobre uma possível substância a ser evitada? *(According to the document, what is the recommended solution for nebulization in case of induced sputum collection, specifying the composition and warning about a possible substance to be avoided?)*	Preparar a solução salina 3% (5 ml de Soro Fisiológico 0,9% + 0,5 ml de NaCl 20%) e realizar nebulização durante 15 minutos. Não utilizar solução preparada com água destilada e NaCl devido ao risco de broncoespasmo. *(Prepare the 3% saline solution (5 ml of 0.9% Saline + 0.5 ml of 20% NaCl) and nebulize for 15 minutes. Do not use a solution prepared with distilled water and NaCl due to the risk of bronchospasm.)*	Synthetic
Onde esta disponível o checklist de prevenção de infecção de corrente sanguínea? *(Where is the bloodstream infection prevention checklist available?)*	O checklist de Prevenção da Infecção de Corrente Sanguínea Associada a Cateter Venoso Central (CVC) está disponível no site do hospital no sistema ADS Hospitalar, aba documentos, pasta CCIH, formulário CCIH – 002. *(The Central Venous Catheter (CVC) Associated Bloodstream Infection Prevention Checklist is available on the hospital's website in the ADS Hospitalar system, under the documents tab, CCIH folder, CCIH form – 002.)*	Expert

ences between the embedding models became less significant. Thus, for experimental parsimony, we adopted the *intfloat/multilingual-e5-small* model in conjunction with RRF, which showed better performance. This configuration is used

as a fixed basis in the present work. This choice is also supported by Medeiros and de Oliveira [2025], who compared open and proprietary embedding models and LLMs for RAG over Brazilian Portuguese collections and reported that the Multilingual E5 model achieved the best retrieval performance.

In concrete terms, the adopted retrieval is hybrid, combining two sources of evidence: (i) sparse retrieval, based on term matching, using BM25; and (ii) dense retrieval, based on semantic similarity, using the intfloat/multilingual-e5-small embedding model. The rankings produced by each method are combined through Reciprocal Rank Fusion (RRF), yielding a unified list of candidate chunks that is subsequently re-ranked. This combination leverages the lexical precision of BM25 and the semantic generalization capability of dense embeddings.

4.3 Evaluation of the Generation with Context-Building Strategies

The generation phase was evaluated using different language models with distinct characteristics, namely: Gemini Flash 2.5, GPT-4.1-mini, Sabiá 3.1, and DeepSeek 3.2. Responses were generated through API calls, maintaining a standardized experimental protocol.

To ensure comparability between models, a single base prompt was used, keeping both the input format and the instruction structure constant. No additional techniques, such as fine-tuning or specific prompt adjustments per model, were applied. In all cases, the generation considered the original question and the set of selected chunks as context.

The experiments were conducted on the complete set of 192 questions from the dataset. For each question, two context-building strategies were implemented: (i) fixed context, using the Top-50 chunks, and (ii) incremental context, in which the context is progressively expanded. Both methods were applied to all evaluated models, and the generated responses were stored for later analysis.

In the incremental method, the value of k used in each query, the number of iterations required to generate the final response, and the total number of tokens consumed were also recorded. This data allowed for a detailed analysis of the method's behavior and efficiency.

The consumption of tokens was accounted for considering both input tokens (composed of the prompt and the context) and output tokens (generated response). In the case of the incremental method, the total cost per question was obtained by summing the tokens used in all iterations. This metric was used as an indicator of computational cost.

The evaluation of the responses was conducted in three complementary dimensions. Semantic quality was measured using the BERTScore, which assesses the similarity between the generated response and the answer key. Computational cost was analyzed based on the total number of tokens used, and finally, an LLM-as-a-Judge approach was used to evaluate qualitative aspects of the responses [Gu *et al.*, 2026].

In the LLM-as-a-Judge assessment, the Gemini Pro 2.5 model was used in zero-shot mode, without fine-tuning, evaluating each instance based on the question, answer key, and generated response. The model was instructed to act as a high-precision evaluator, following the prompt detailed in the table 2, which defines criteria of *correctness*, *completeness*, *relevance*, and *consistency*, allowing for an analysis closer to human evaluation.

Table 2. Prompt used in the evaluation with the LLM-as-a-Judge method.

Prompt Type	Content

System Message	You are an expert evaluator for question answering systems.
	Your task is to evaluate a generated answer by comparing it with a reference answer.
	Question: `{question}`
	Reference Answer: `{correct answer}`
	Generated Answer: `{answer}`
	Evaluate the generated answer based on the following criteria:
	• **Correctness**: factual accuracy
	• **Completeness**: coverage of key info
	• **Relevance**: answers the question
	• **Consistency**: no contradictions
	Score each criterion from 1 to 5 and compute the average.
	`{`
	`"correctness": number,`
	`"completeness": number,`
	`"relevance": number,`
	`"consistency": number,`
	`"final_score": number,`
	`"justification": "..."`
	`}`

User Message	`Question: {question}`
	`Reference Answer: {correct answer}`
	`Generated Answer: {answer}`

In this sense, according to the Table 3, the LLM-as-a-Judge evaluation assesses multiple qualitative dimensions of the generated responses. Correctness measures the factual accuracy of the answer with respect to the reference. Completeness evaluates whether the response fully covers the relevant aspects of the expected answer. Relevance captures how directly the response addresses the question, penalizing the inclusion of unnecessary or unrelated information. Finally, Consistency assesses the internal coherence of the response, ensuring that it does not contain contradictions or logical inconsistencies.

Table 3. Dimensions evaluated by the LLM-as-a-Judge approach.

Metric	What it measures	Main focus
Correctness	Factual accuracy of the response	Is it correct?
Completeness	Coverage of relevant information	Is it complete?
Relevance	Pertinence to the question	Is it focused?
Consistency	Internal coherence of the response	Is it logically consistent?

Together, these metrics allow for a multidimensional evaluation of the responses, capturing not only factual accuracy but also aspects related to the quality, clarity, and usefulness of the information generated.

4.3.1 Fixed-Context Generation Assessment

After defining the retrieval and reordering strategy, the generation stage was evaluated using the complete hybrid dataset, composed of 192 questions. The objective of this stage was to analyze the behavior of the language models when exposed to a fixed and relatively extensive context, composed of multiple chunks concatenated into a single inference.

For each question, the 50 most relevant chunks were

selected from the ranking generated by the retrieval method (Top-50), which were used as input context for the models. This configuration follows the common practice in the RAG literature, in which a fixed number of documents is used regardless of the complexity of the query. Each inference considered exclusively the original question and the fixed set of selected chunks.

The main objective of this stage was to investigate how the use of extensive contexts impacts the number of tokens and the quality of the generated responses, especially in terms of the possible inclusion of irrelevant information and context dilution effects. Additionally, it sought to establish a baseline for comparison with the incremental method, allowing for the analysis of trade-offs between the amount of context, response quality, and computational cost.

4.3.2 Technical Assessment of Incremental Context Retrieval

As a main methodological contribution, this work proposes and evaluates the Incremental Retrieval technique, whose objective is to reduce the consumption of tokens and mitigate the degradation of response quality associated with the use of extended contexts.

Unlike the fixed-context approach, in which a predefined set of chunks is sent to the model in a single inference, Incremental Retrieval builds the context progressively. Initially, only the Top-5 chunks from the retrieval ranking are provided to the language model. If the answer is not found, new blocks of 5 chunks are added iteratively, respecting the ranking order, up to a maximum limit of 50 chunks. In each iteration, the language model (LLM) is triggered with the updated context. The process is interrupted when the model produces a response considered valid or when the maximum number of iterations is reached.

The stopping criterion relies exclusively on the model's own assessment of whether the current context is sufficient to answer the question. The prompt sent at each iteration contains only two pieces of information — the user's question and the retrieved chunks — and instructs the model to produce a single, direct answer or, when the requested information is not present in the provided texts, to reply exactly with the marker "NÃO ENCONTRADO" ("NOT FOUND"). Table 4 reproduces the prompt used. A response is therefore classified as valid when the model returns an actual answer without emitting this marker, and as invalid when the marker is returned. Crucially, the prompt never receives the reference answer: the decision depends only on inputs available at inference time (the question and the retrieved context). For this reason, the stopping criterion is directly applicable in real-world use, and not merely as an evaluation strategy.

Relevance is handled upstream, by the hybrid retrieval and the RRF-based ordering: chunks are appended in decreasing order of relevance, so that the most pertinent evidence is presented first. The stopping check does not verify whether an arbitrary or irrelevant snippet is contained in the answer; rather, it assesses whether the accumulated context is sufficient to answer the question. When the available chunks are not relevant enough to support an answer, the model tends to emit the "NÃO ENCONTRADO" marker, triggering a new expansion of the context. This mechanism controls the growth of the context and avoids unnecessary calls to the model. We acknowledge that this self-assessment depends on the model's own behavior and may fail in ambiguous cases or with partially correct answers, as discussed in the Limitations section.

Table 4. Prompt used at each iteration of Incremental Retrieval (stopping criterion). Unlike the LLM-as-a-Judge prompt, it receives only the question and the retrieved chunks; no reference answer is provided.

Prompt Part	Content
Instruction	You are a document analysis expert. Your task is to answer the question based solely on the provided texts, producing a single, direct final answer.
Rules	• Generate a SINGLE final answer • Do NOT answer per document or write "Document 1", "Document 2", etc. • Do NOT list multiple separate answers or repeat excerpts • Be objective and direct • If the answer is NOT present in the texts, reply exactly with: "NÃO ENCONTRADO" (NOT FOUND)
Inputs	**Question:** `{query}` **Documents:** `{contexto}`

The experiments were conducted allowing for incremental recovery up to position 50 in the ranking, with the following being recorded for each query:

- the number of chunks required to obtain a valid answer;
- the total number of tokens consumed across the iterations;
- the quality of the response generated.

The evaluation was performed using the four language models, allowing for the analysis of the technique's behavior across different architectures. The objective was to compare the efficiency of the incremental approach versus the fixed-context method, considering both the computational cost (measured by the number of tokens) and the quality of the generated responses.

5 Results and Discussion

This section presents the results obtained from the experiments described in the methodology, followed by a comparative analysis and discussion of their implications.

5.1 Performance of embedding and re-ranking models in information retrieval.

The comparative analysis of the three embedding models carried out in the [da Cunha *et al.*, 2025] study revealed a substantial disparity in their robustness. The *intfloat/multilingual-e5-small* model demonstrated clear superiority. As shown in Table 5, the *intfloat/multilingual-e5-small* model achieved an average MRR of 0.6662, considerably higher than the scores of the other two models, both around 0.52.

In addition to the position of the first correct result, the overall quality of the classification was evaluated using the NDCG@10 metric. The results of this analysis, presented in Table 6, further reinforce the advantage of the

Table 5. MRR metric statistics by model

Model	Average	Median	Freq MRR = 1(%)
all-MiniLM-L6-v2	0.5225	0.5	40.5405
paraphrase	0.5200	0.5	40.5405
infloat	**0.6662**	**1.0**	**58.1081**

intfloat/multilingual-e5-small model, which achieved an average NDCG@10 score of 0.6782, surpassing both *paraphrase-multilingual-MiniLM-L12-v2* (0.5625) and *all-MiniLM-L6-v2* (0.5445).

Table 6. NDCG@10 metric statistics by model

Model	Average	Median	Freq NDCG@10 = 1 (%)
all-MiniLM-L6-v2	0.5445	0.6309	36.4865
paraphrase	0.5625	0.6220	37.8378
infloat	**0.6782**	**1.0000**	**52.7027**

After defining the most effective incorporation model, the research focused on optimizing the reclassification step. Analysis of the Mean Reciprocal Rank (MRR) metric revealed a performance hierarchy, as shown in Table 7. Although Gemini Re-ranker presented the highest average MRR, with 0.8612, and the highest frequency of perfect hits (MRR=1), securing first place, the RRF method also demonstrated remarkably strong performance, with an average MRR of 0.7857.

Table 7. MRR metric statistics by reordering model

Model	Average	Median	Freq MRR = 1 (%)
CrossEncoder	0.6584	1.0	54.0541
RRF	0.7857	1.0	67.5676
Gemini Flash 1.5	**0.8612**	**1.0**	**78.3784**

Furthermore, the quality of the classification was evaluated using the NDCG@10 metric. As demonstrated in Table 8, the results confirm the superiority of the Gemini-based reclassifier, which achieved an average NDCG@10 score of 0.8826. In this case, the RRF method showed similar performance, reaching an average of 0.8157.

Table 8. NDCG@10 metric statistics by reordering model

Model	Average	Median	Freq NDCG@10 = 1 (%)
CrossEncoder	0.6906	0.9599	50.0000
RRF	0.8157	1.0000	64.8649
Gemini Flash 1.5	**0.8826**	**1.0000**	**75.6757**

Although the Gemini Flash 1.5 model presented the best results in the MRR and NDCG@10 metrics, the RRF method was chosen as the re-ranking strategy in this work. It is worth noting that RRF is not a retrieval method in itself: it operates by fusing the rankings produced by other retrievers. In this work, RRF combined two rankings — one from BM25 (sparse, term-based retrieval) and one from the intfloat/multilingual-e5-small embedding model (dense, semantic retrieval) — into a single unified ranking. This choice is mainly justified by its deterministic and stable nature, since the fusion does not introduce stochastic variability into the process. Furthermore, because it does not dynamically modify the set of retrieved documents, it is possible to maintain the same top-k of candidates throughout the experiments, which is fundamental for a fair comparison between different response generation models (LLMs).

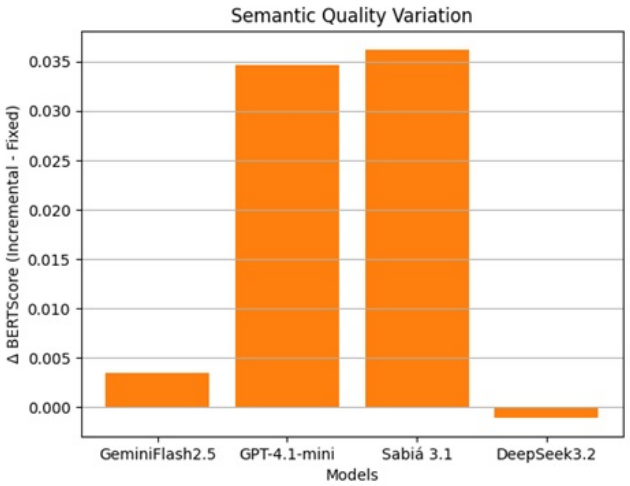

Figure 3. Difference in BertScore between the Incremental and Fixed methods.

5.2 Quality of Response Generation

This subsection presents an analysis of the results obtained in the evaluation of context construction strategies, comparing the fixed context method (Top-50) with the proposed incremental approach. The analysis considers three main dimensions: response quality, computational cost of using tokens, and the behavior of the incremental method.

For the fixed-context method, the table 9 shows the results of evaluating the quality of the responses using the BertScore and LLM-as-a-Judge metrics, which assess: Correctness, Completeness, Relevance, and Consistency.

Table 9. Average results for the fixed context method (Top-50 chunks).

Modelo	BERT	Corr.	Comp.	Rel.	Cons.	Final
Gemini Flash 2.5	0.7987	4.4583	4.0573	4.6615	4.5260	4.4258
GPT-4.1-mini	0.7737	4.6158	4.4579	4.7895	4.7000	4.6408
Sabiá 3.1	0.7452	4.1257	3.7016	4.4503	4.2304	4.1269
DeepSeek 3.2	0.8110	4.6316	4.2368	4.8211	4.7158	4.6013

The results presented in Table 10 demonstrate that the incremental method maintains performance comparable to the fixed context, with variations depending on the model used.

Table 10. Average results for the context incremental method.

Modelo	BERT	Corr.	Comp.	Rel.	Cons.	Final
Gemini Flash 2.5	0.8021	4.4740	3.8802	4.7813	4.5938	4.4323
GPT-4.1-mini	0.8083	4.4421	4.1790	4.7158	4.5000	4.4592
Sabiá 3.1	0.7814	4.0524	3.9634	4.6545	4.1885	4.2147
DeepSeek 3.2	0.8099	4.4790	3.9526	4.9053	4.6000	4.4842

In terms of BERTScore, it is observed that three of the four models (Gemini Flash 2.5, GPT-4.1-mini, and Sabiá 3.1) showed improvement in the incremental method, with GPT-4.1-mini (+0.0346) and Sabiá 3.1 (+0.0362) standing out, as shown in Table 11 and Figure 3. These results indicate that reducing the context does not compromise the semantic quality of the responses and may even improve it.

Table 11. Difference between incremental and fixed methods.

Modelo	Δ BERTScore	Δ Correctness	Δ Completeness	Δ Relevance	Δ Consistency	Δ Final Score
Gemini Flash 2.5	+0.0034	+0.0156	-0.1771	+0.1198	+0.0677	+0.0065
GPT-4.1-mini	+0.0346	-0.1737	-0.2790	-0.0737	-0.2000	-0.1816
Sabiá 3.1	+0.0362	-0.0733	+0.2618	+0.2042	-0.0419	+0.0877
DeepSeek 3.2	-0.0011	-0.1526	-0.2842	+0.0842	-0.1158	-0.1171

Analysis based on LLM-as-a-Judge reveals more complex behavior. For the Gemini Flash 2.5 model, there was an improvement in correctness, relevance, and consistency, suggesting that incremental context reduces noise and favors more precise responses. Similarly, the Sabiá 3.1 model showed significant gains in completeness and relevance, indicating better alignment with relevant content.

On the other hand, GPT-4.1-mini and DeepSeek 3.2 showed a reduction in some metrics, especially completeness. This behavior suggests that, although incremental context increases precision, it may limit the scope of responses in models that rely on a larger volume of information.

However, in a general analysis, the results indicate that the incremental method not only preserves the quality of responses, but can also mitigate context dilution effects by prioritizing more relevant information.

5.3 Computational Cost and Efficiency

Table 12 and Figure 4 show a significant reduction in token consumption when using the incremental method. Savings ranged from 34.79% (Gemini Flash 2.5) to 88.68% (Sabiá 3.1), with consistent reductions for all models.

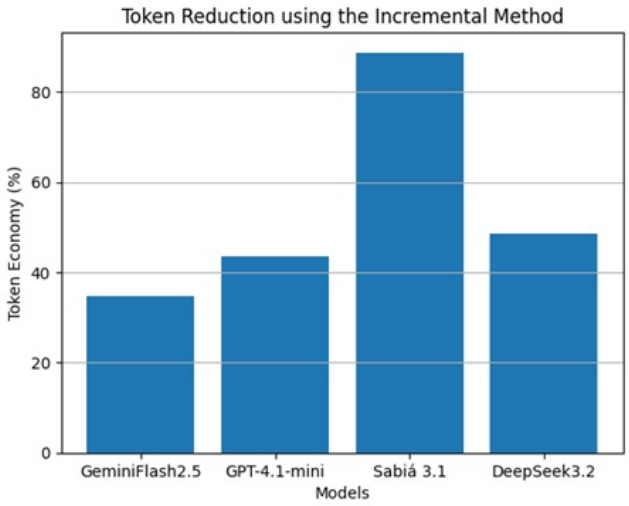

Figure 4. Token economy when using the Incremental method compared to the Fixed method.

This result, analyzed together with the response quality results from tables 9 and 10, confirms one of the main hypotheses of this work: the incremental construction of the context allows for a significant reduction in computational cost without compromising response quality.

Table 12. Total token consumption and percentage savings for each model.

Model	Top-50 Fixed	Incremental	Token economy (%)
Gemini Flash 2.5	11224073	7319075	34.79
GPT-4.1-mini	11224073	6346035	43.46
Sabiá 3.1	11224073	1270020	88.68
DeepSeek 3.2	11224073	5776404	48.54

Figure 5 reinforces this finding by demonstrating the trade-off between cost and quality. It can be observed that the points corresponding to the incremental method shift to regions of lower token consumption, maintaining similar or higher BERTScore values. In the specific case of Sabiá3.1, as shown in tables 9 and 10, there is an increase in BERTScore from 0.7452 in the fixed context to 0.7814 in the incremental context, with an 88.68% reduction in token usage. This indicates greater efficiency in the use of the context.

5.4 Analysis of Context Dilution

The results suggest that using large amounts of context can introduce irrelevant information, impairing the model's ability to identify important evidence. This phenomenon, known as context dilution, is mitigated by the incremental approach.

The consistent improvement in the relevance metric, especially for the Gemini Flash 2.5 and Sabiá 3.1 models, as can be seen in Figure 6, reinforces this interpretation. By providing only progressive subsets of information, the model is able to focus on the most relevant segments, reducing interference.

Furthermore, the reduction in completeness observed in some models can be interpreted as a side effect of the smaller volume of context, highlighting a trade-off between precision and coverage.

5.5 Analysis of the Behavior of the Incremental Method

Table 13 shows the frequency of the k values used by the incremental method for each model. In general, there is a strong concentration on the lower k values, indicating that, in many cases, the models choose to generate responses with a reduced amount of context.

Table 13. Frequency of the value of k used in the incremental method.

k	Gemini Flash 2.5	GPT-4.1-mini	Sabiá 3.1	DeepSeek 3.2
5	154	172	184	170
10	15	3	8	5
15	4	4	0	4
20	4	1	0	2
25	1	0	0	0
30	1	0	0	1
35	0	0	0	0
40	0	0	0	0
45	0	1	0	3
50	13	11	0	7

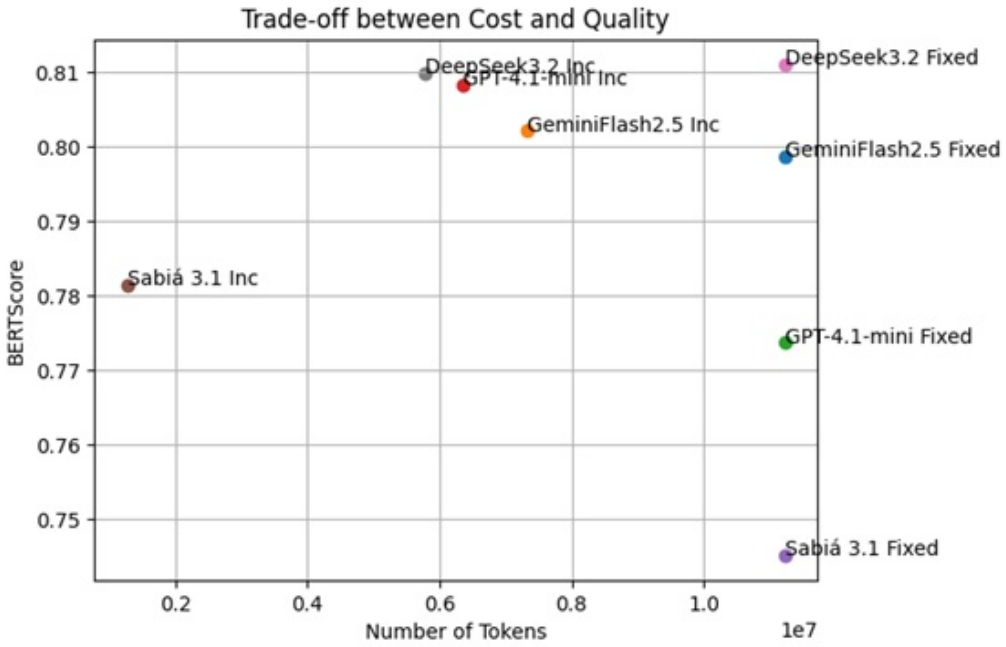

Figure 5. Comparison between BertScore quality and token usage.

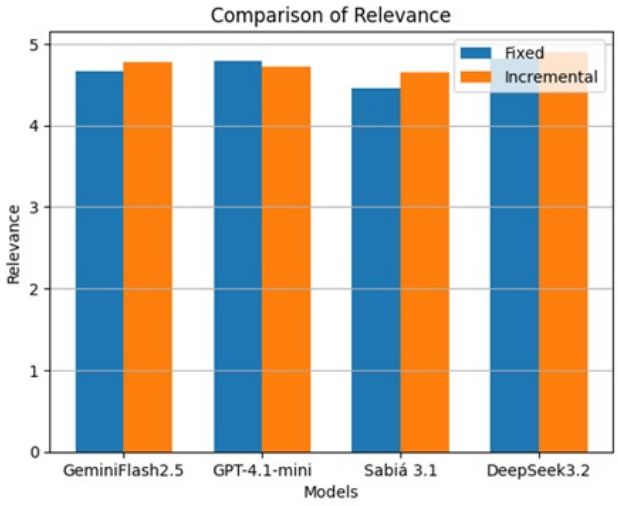

Figure 6. Comparison relevance.

An important observation from this step is that, although low values of k were common in the experiments, this should not be interpreted as evidence that using only a small fixed context (e.g., Top-5) is sufficient. The results indicate that the models sometimes respond too early, before examining the entire retrieved context.

This interpretation is reinforced by the joint analysis with quality metrics. For example, the Sabiá 3.1 model presents the highest frequency of responses with $k = 5$, but obtains lower BERTScore values, as shown in Table 10, compared to the other models. This result suggests that the model tends to over-rely on the first chunks retrieved, which can lead to incomplete responses or responses that are less semantically aligned with the template. In other words, an early stopping decision can compromise the quality of the response.

On the other hand, higher values of k, although less common, suggest that some questions required the models to expand the retrieved context before finding relevant information. The cases where $k = 50$ in Table 13 are especially important because they show that some models were still unable to use the necessary information even after receiving many additional chunks. This difference across models may reflect variations in how they handle long contexts and combine information from multiple retrieved passages.

The results show that the incremental method can reduce unnecessary context usage while also exposing differences in how the models behave. Some models tend to answer quickly using only a small amount of context, whereas others are better at incorporating additional retrieved information when needed. These findings suggest that the main advantage of the incremental approach is its ability to adjust the amount of retrieved context according to the demands of each query, avoiding both excessive use and premature use of insufficient information.

5.6 Summary of Results

The results obtained allow us to draw three main conclusions, which go beyond purely quantitative gains and highlight relevant behavioral aspects of the models in enhanced recovery scenarios:

- The incremental method significantly reduces token consumption, with savings of up to 88%, by avoiding the unnecessary inclusion of irrelevant context. This gain is achieved without compromising the model's ability to access essential information when needed, highlighting the efficiency of the progressive expansion mechanism.
- The quality of the responses is maintained or, in some cases, improved in terms of semantic similarity and relevance. However, detailed analysis reveals that this quality depends not only on the amount of context used, but also on how the models interact with it. In particular, it is observed that premature generation decisions can lead

to under utilization of the available context, negatively impacting the performance of some models;
- The progressive construction of context acts as an adaptive mechanism that balances efficiency and quality, mitigating both the excessive inclusion of irrelevant information and limitations associated with processing long contexts. Specifically, the method helps reduce the effects of attention saturation, in which increasing the context does not translate into proportional performance gains, while encouraging a more appropriate exploration of the available evidence.

Taken together, these findings indicate that the effectiveness of the incremental method lies not only in reducing computational cost, but also in its ability to dynamically regulate the context utilization process. Therefore, the proposed approach presents itself as a robust alternative to the use of extensive fixed contexts, being particularly relevant for applications in RAG systems, where there is a critical trade-off between cost, quality, and reliability of responses.

6 Limitations

The study presents promising results; however, it has some limitations that should be considered when interpreting them. Firstly, the evaluation was conducted within a specific domain of normative documents from a university hospital, which may limit the generalization of the findings to other contexts and types of information.

Furthermore, the use of the LLM-as-a-Judge method for qualitative evaluation of responses was a valuable addition compared to the previous study, allowing for an analysis closer to human evaluation. However, it depends on the behavior of the evaluation model itself, which may introduce biases in the assignment of scores.

In this study, the number of tokens was used as an estimate of computational and financial cost. Although token usage is commonly adopted in API-based evaluations, it does not account for other factors that also affect cost, such as latency, energy consumption, or pricing differences across providers.

Additionally, the adopted segmentation was size-based with preservation of syntactic cohesion; contextual/semantic chunking strategies (e.g., the docling hybrid chunker), which may improve chunk delimitation and, consequently, retrieval, were not evaluated. This comparison constitutes a direction for future work.

Finally, the incremental method adopted depends on a stopping criterion based on the identification of valid responses by the model itself, which may not be perfectly robust in all cases, especially in ambiguous situations or with partially correct responses, requiring other detection techniques.

7 Conclusion

This work presented the development and evaluation of a RAG-based chatbot for retrieving normative documents in a hospital environment, focusing on the analysis of context-building strategies during response generation.

As a main contribution, an incremental context approach was proposed and evaluated, in which the context is progressively expanded as needed. The results demonstrated that this strategy is capable of significantly reducing token consumption, with savings of up to 88%, while maintaining or improving the quality of responses in terms of semantic similarity and relevance.

Furthermore, the results reinforce the importance of context selection and organization for the performance of LLM-based systems, especially in sensitive applications such as healthcare.

Future work intends to investigate more robust strategies for the stopping criterion of the incremental method, explore different context expansion policies, and evaluate the performance of a qualitative assessment with hospital end-users to analyze aspects such as usefulness, clarity, reliability, and adequacy of responses in real-world use. Finally, the possibility of incorporating more comprehensive cost metrics, including latency and energy consumption, is also noteworthy.

Declarations

Acknowledgements

This work was supported by Instituto Federal do Rio Grande do Sul (IFRS), Empresa Brasileira e Serviços Hospitalares (EBSERH) and Hospital Escola da UFPEL. This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001. We would like to thank the FAPERGS - Brasil for Financial Support, Award Agreement 22/2551-0000598-5. We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan X Pascal GPU used for this research.

Authors' Contributions

All authors read, contributed equally, and approved the final manuscript.

Competing interests

The authors declare they have no competing interests.

Availability of data and materials

The experimental scripts used in this study are available at: `https://github.com/Murilovcunha/Incremental-Context-Strategies-in-RAG-Based-Chatbots`. Due to the sensitive nature of hospital documents, the original corpus cannot be publicly shared; however, evaluation artifacts are provided to support reproducibility.

Further relevant information

This research adheres to the ethical guidelines established our institution with approval from the Research Ethical Committee of the university, under protocol CAAE 88037225.9.0000.5317. The authors acknowledge the use of generative AI for grammar enhancement, paragraph adjustment, and language improvement during the preparation of this manuscript.

References

Abo El-Enen, M., Saad, S., and Nazmy, T. (2025). A survey on retrieval-augmentation generation (rag) models for healthcare applications. *Neural Computing and Applications*, 37(33):28191–28267. DOI: https://doi.org/10.1007/s00521-025-11666-9.

Asai, A., Wu, Z., Wang, Y., Sil, A., and Hajishirzi, H. (2023). Self-rag: Self-reflective retrieval augmented generation. In *NeurIPS 2023 Workshop on Instruction Tuning and Instruction Following*.

Ayala, O. and Bechard, P. (2024). Reducing hallucination in structured outputs via retrieval-augmented generation. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 6: Industry Track)*, pages 228–238. Association for Computational Linguistics. DOI: https://doi.org/10.18653/v1/2024.naacl-industry.19.

Baur, D., Ansorg, J., Heyde, C.-E., and Voelker, A. (2025). Development and evaluation of a retrieval-augmented generation chatbot for orthopedic and trauma surgery patient education: Mixed-methods study. *JMIR AI*, 4:e75262. DOI: https://doi.org/10.2196/75262.

Brown, T. B. *et al*. (2020). Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Carraro, D. and Bridge, D. (2025). Enhancing recommendation diversity by re-ranking with large language models. *ACM Transactions on Recommender Systems*, 4(2):1–40. DOI: https://doi.org/10.1145/3700604.

Cederlund, O., Alawadi, S., and Awaysheh, F. M. (2024). Llmrag: An optimized digital support service using llm and retrieval-augmented generation. In *Proceedings of the 9th International Conference on Fog and Mobile Edge Computing (FMEC 2024)*, pages 54–62, Malmö, Sweden. IEEE. DOI: https://doi.org/10.1109/FMEC62297.2024.10710181.

Cormack, G. V., Clarke, C. L., and Buettcher, S. (2009). Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 758–759. ACM. DOI: https://doi.org/10.1145/1571941.1572114.

da Cunha, M. V., Silveira, M. R., Santana, B. S., Freitas, L. A., and Corrêa, U. B. (2025). Optimizing and evaluating a retrieval-augmented generation system for normative document retrieval in hospital settings. In *Anais do XXXI Simpósio Brasileiro de Sistemas Multimídia e Web (WebMedia)*, pages 385–393, Porto Alegre, Brasil. SBC. DOI: https://doi.org/10.5753/webmedia.2025.16029.

Devi, S., Dhar, G., Bharadwaj, C., and M, A. (2024). Retrieval augmented medlm. In *Proceedings of the 2024 IEEE Conference on Artificial Intelligence (CAI 2024)*, pages 1220–1221, Singapore. IEEE. DOI: https://doi.org/10.1109/CAI59869.2024.00217.

Empresa Brasileira de Serviços Hospitalares (EBSERH) (2025). Estrutura administrativa — hu-ufsc. `https://www.gov.br/ebserh/pt-br/hospitais-universitarios/regiao-sul/hu-ufsc/governanca/estrutura-administrativa`. Accessed on 1 September 2026.

Fan, W., Ding, Y., Ning, L., Wang, S., Li, H., Yin, D., Chua, T.-S., and Li, Q. (2024). A survey on rag meeting llms: Towards retrieval-augmented large language models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 6491–6501, Barcelona, Spain. ACM. DOI: https://doi.org/10.1145/3637528.3671470.

Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., Guo, Q., Wang, M., *et al*. (2023). Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*. DOI: https://doi.org/10.48550/arXiv.2312.10997.

Gomez-Cabello, C. A., Prabha, S., Haider, S. A., Genovese, A., Collaco, B. G., Wood, N. G., Bagaria, S., and Forte, A. J. (2025). Comparative evaluation of advanced chunking for retrieval-augmented generation in large language models for clinical decision support. *Bioengineering*, 12(11):1194. DOI: https://doi.org/10.3390/bioengineering12111194.

Gu, J., Jiang, X., Shi, Z., Tan, H., Zhai, X., Xu, C., Li, W., Shen, Y., Ma, S., Liu, H., *et al*. (2026). A survey on llm-as-a-judge. *The Innovation*, 6(9):101253. DOI: https://doi.org/10.1016/j.xinn.2025.101253.

Gummadi, V., Udayaraju, P., Sarabu, V. R., Ravulu, C., Seelam, D. R., and Venkataramana, S. (2024). Enhancing communication and data transmission security in rag using large language models. In *Proceedings of the 4th International Conference on Sustainable Expert Systems (ICSES 2024)*, pages 612–617. IEEE. DOI: https://doi.org/10.1109/ICSES63445.2024.10763024.

Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y. J., Madotto, A., and Fung, P. (2023a). Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38. DOI: https://doi.org/10.1145/3571730.

Ji, Z., Yu, T., Xu, Y., Lee, N., Ishii, E., and Fung, P. (2023b). Towards mitigating llm hallucination via self reflection. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 1827–1843. Association for Computational Linguistics. DOI: https://doi.org/10.18653/v1/2023.findings-emnlp.123.

Joshi, P., Gupta, A., Kumar, P., and Sisodia, M. (2024). Robust multi model rag pipeline for documents containing text, table & images. In *Proceedings of the 3rd International Conference on Applied Artificial Intelligence and Computing (ICAAIC 2024)*, pages 993–999, Salem, India. IEEE. DOI: https://doi.org/10.1109/ICAAIC60222.2024.10574972.

Kalyan, K. S. and Sangeetha, S. (2020). Secnlp: A survey of embeddings in clinical natural language processing. *Journal of Biomedical Informatics*, 101:103323. DOI: https://doi.org/10.1016/j.jbi.2019.103323.

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., Riedel, S., and Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive nlp tasks. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H., editors, *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc.

Maryamah, M., Irfani, M. M., Tri Raharjo, E. B., Rahmi, N. A., Ghani, M., and Raharjana, I. K. (2024). Chatbots in academia: A retrieval-augmented generation approach for improved efficient information access. In *Proceedings of the 16th International Conference on Knowledge and Smart Technology (KST 2024)*, pages 259–264, Krabi, Thailand. IEEE. DOI: https://doi.org/10.1109/KST61284.2024.10499652.

Medeiros, L. S. F. and de Oliveira, H. T. A. (2025). Comparação de modelos de embeddings e llms para

geração aumentada por recuperação em português. In *Anais do LII Seminário Integrado de Software e Hardware (SEMISH)*, pages 429–440. SBC. DOI: https://doi.org/10.5753/semish.2025.9027.

Nai, R., Sulis, E., Fatima, I., and Meo, R. (2024). Large language models and recommendation systems: A proof-of-concept study on public procurements. In *Natural Language Processing and Information Systems (NLDB 2024), Part II*, pages 280–290, Turin, Italy. Springer. DOI: https://doi.org/10.1007/978-3-031-70242-6_27.

Obaid, S. and Bawany, N. Z. (2024). Seerahgpt: Retrieval augmented generation based large language model. In *Proceedings of the 18th International Conference on Open Source Systems and Technologies (ICOSST 2024)*, pages 1–7, Lahore, Pakistan. IEEE. DOI: https://doi.org/10.1109/ICOSST64562.2024.10871159.

Oliveira, S. S. T., Fazzioni, D., and Ferreira, D. O. C. (2025). Grandes modelos de linguagem. In *Kudo, T. N. et al. (Org.)*. Cegraf UFG, Goiânia. E-book (254 p.). ISBN 978-85-495-1096-9. Accessed on 1 September 2026.

Perković, G., Drobnjak, A., and Botički, I. (2024). Hallucinations in llms: Understanding and addressing challenges. In *Proceedings of the 47th MIPRO ICT and Electronics Convention (MIPRO 2024)*, pages 2084–2088, Opatija, Croatia. IEEE. DOI: https://doi.org/10.1109/MIPRO60963.2024.10569238.

Puthenputhussery, A., Kang, C., Magnani, A., Zhang, T., Shang, H., Yadav, N., Chandran, P., Madhani, B., Fu, Y.-T., Wang, H., *et al*. (2025). Large scale deployment of bert based cross encoder model for re-ranking in walmart search engine. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 4365–4369. ACM. DOI: https://doi.org/10.1145/3726302.3731965.

Ramos, J. *et al*. (2003). Using tf-idf to determine word relevance in document queries. In *Proceedings of the First Instructional Conference on Machine Learning*, volume 242, pages 29–48, New Jersey, USA.

Raschka, S. (2024). *Build a Large Language Model (From Scratch)*. Manning Publications.

Raza, M., Jahangir, Z., Riaz, M. B., Saeed, M. J., and Sattar, M. A. (2025). Industrial applications of large language models. *Scientific Reports*, 15:13755. DOI: https://doi.org/10.1038/s41598-025-98483-1.

Reimers, N. and Gurevych, I. (2019). Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992. Association for Computational Linguistics. DOI: https://doi.org/10.18653/v1/D19-1410.

Son, N., Kang, I., Kim, I., Lee, K., Nam, S., and Lee, D. (2025). Development and evaluation of a retrieval-augmented generation-based electronic medical record chatbot system. *Healthcare Informatics Research*, 31(3):218–225. DOI: https://doi.org/10.4258/hir.2025.31.3.218.

Taguchi, C., Maekawa, S., and Bhutani, N. (2025). Efficient context selection for long-context qa: No tuning, no iteration, just adaptive-k. *arXiv preprint arXiv:2506.08479*. DOI: https://doi.org/10.48550/arXiv.2506.08479.

Wei, J., Tay, Y., Bommasani, R., Raffel, C., Zoph, B., Borgeaud, S., Yogatama, D., Bosma, M., Zhou, D., Metzler, D., Chi, E. H., Hashimoto, T., Vinyals, O., Liang, P., Dean, J., and Fedus, W. (2022). Emergent abilities of large language models. *Transactions on Machine Learning Research*. Survey Certification.

Wijaya, O. C. and Purwarianti, A. (2024). An interactive question-answering system using large language model and retrieval-augmented generation in an intelligent tutoring system on the programming domain. In *Proceedings of the 2024 11th International Conference on Advanced Informatics: Concept, Theory and Application (ICAICTA)*, pages 1–6, Singapore. IEEE. DOI: https://doi.org/10.1109/ICAICTA63815.2024.10763263.

Yu, H., Gan, A., Zhang, K., Tong, S., Liu, Q., and Liu, Z. (2025). Evaluation of retrieval-augmented generation: A survey. In Zhu, W., Xiong, H., Cheng, X., Cui, L., Dou, Z., Dong, J., Pang, S., Wang, L., Kong, L., and Chen, Z., editors, *Big Data (BigData 2024), Communications in Computer and Information Science*, pages 102–120, Singapore. Springer. DOI: https://doi.org/10.1007/978-981-96-1024-2_8.

Zhang, D., Du, H., Wang, X., Zhu, M., Pang, X., Wei, D., and Wang, X. (2025). Cmedragbot: A chinese medical chatbot based on graph rag and large language models. *Interdisciplinary Sciences: Computational Life Sciences*. DOI: https://doi.org/10.1007/s12539-025-00715-5.

Zhao, W. X. *et al*. (2023). A survey of large language models. *arXiv preprint arXiv:2303.18223*. DOI: https://doi.org/10.48550/arXiv.2303.18223.

Zhou, Y., Dai, S., Cao, Z., Zhang, X., and Xu, J. (2024). Length-induced embedding collapse in transformer-based models. *arXiv preprint arXiv:2410.24200*. DOI: https://doi.org/10.48550/arXiv.2410.24200.

Zhu, Z., Wang, Y., Liu, H., Chen, X., and Zhang, M. (2024). Enhancing large language models with knowledge graphs for robust question answering. In *Proceedings of the 30th IEEE International Conference on Parallel and Distributed Systems (ICPADS 2024)*, pages 262–269, Belgrade, Serbia. IEEE. DOI: https://doi.org/10.1109/ICPADS63350.2024.00042.