FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Retrieval-Augmented Generation (RAG) Systems and Large Language Models (LLMs) in Clinical Pharmacology Decision-Making

João Pedro Oliveira Sequeira

Mestrado em Engenharia Informática e de Computação

Supervisor: Prof. Carlos Ferreira

Second Supervisor: Prof. Rui Camacho

July 17, 2026

Retrieval-Augmented Generation (RAG) Systems and Large Language Models (LLMs) in Clinical Pharmacology Decision-Making

João Pedro Oliveira Sequeira

Mestrado em Engenharia Informática e de Computação

Approved in oral examination by the committee:

President: Prof. Gabriel de Sousa Torcato David
Examiner: Prof. José Paulo Neves Correia Marques dos Santos
Member: Prof. Carlos Ferreira

July 17, 2026

Resumo

A Farmacologia Clínica desempenha um papel central nos sistemas de saúde modernos, exigindo a síntese sistemática de grandes volumes de informação biomédica heterogénea para suportar decisões terapêuticas fundamentadas em evidência. Na Unidade de Farmacologia Clínica do Hospital de São João, este processo é realizado manualmente por farmacologistas clínicos, sendo moroso, intensivo em recursos e sujeito a inconsistências decorrentes do crescente volume de literatura publicada.

Esta dissertação propõe, implementa e avalia um sistema baseado em Inteligência Artificial para a geração automática de pareceres técnicos de farmacologia clínica. A arquitetura desenvolvida combina técnicas de *Retrieval-Augmented Generation* com um Grafo de Conhecimento construído a partir de fontes regulatórias, seguindo uma estratégia de busca de informação híbrida: factos farmacológicos estruturados são obtidos através do grafo, enquanto literatura biomédica relevante é recuperada por pesquisa semântica vetorial sobre documentos indexados provenientes de repositórios públicos, incluindo PubMed, ClinicalTrials.gov, EMA e NICE. O sistema suporta três categorias de pareceres, entre elas, Avaliação de Fármaco Único, Avaliação Comparativa e Critérios de Iniciação, Continuação e Suspensão e opera de forma modular, gerando cada secção do relatório de forma independente com base em configurações específicas de recuperação de informação e instruções de geração. A pipeline completa foi codificada num Docker container, permitindo a sua disponibilização num ambiente hospitalar sem requisitos complexos de infraestrutura.

A avaliação do sistema segue uma abordagem de dupla fase: avaliação automática com métricas de pipeline (*Ragas*), métricas de semelhança textual e semântica (BERTScore, ROUGE-L, Med-F1 e Similaridade Semântica) e avaliação humana conduzida por dois farmacologistas clínicos da unidade de farmacologia clínica do Hospital de São João. Os resultados automáticos demonstram que os relatórios gerados preservam o significado clínico dos documentos de referência, com valores de Similaridade Semântica entre 0,88 e 0,89 em todas as categorias. As métricas de pipeline revelam um desempenho estável na relevância das respostas geradas mas uma precisão de contexto variável, indicando que o ruído na recuperação de informação, e não falhas na geração, constitui o principal fator restritivo. As secções fundamentadas no Grafo de Conhecimento apresentaram consistentemente melhor desempenho do que as que dependem exclusivamente de recuperação vetorial. A avaliação humana confirmou o potencial do sistema enquanto ferramenta de apoio à decisão, tendo ambos os avaliadores reconhecido a sua utilidade prática após discussão detalhada da arquitetura e das suas restrições atuais.

O sistema demonstra que é possível gerar automaticamente pareceres farmacológicos semanticamente fundamentados e estruturalmente coerentes a partir de fontes abertas, reduzindo a carga de trabalho associada à pesquisa e redação, e preservando o farmacologista como decisor final do processo clínico.

Palavras-chave: Geração Aumentada via Recuperação • Modelos de Linguagem de Grande Escala • Farmacologia Clínica • Sistemas de Apoio à Decisão Clínica • Processamento de Linguagem Natural Biomédica • Recuperação de Informação Médica • Inteligência Artificial Generativa • GraphRAG • IA na Saúde

Abstract

Clinical Pharmacology plays a central role in modern healthcare systems, requiring the systematic synthesis of large volumes of heterogeneous biomedical information to support evidence-based therapeutic decisions. At the Clinical Pharmacology Unit of Hospital de São João, this process is carried out manually by clinical pharmacologists, making it time-consuming, resource-intensive, and susceptible to inconsistencies driven by the ever-growing volume of published literature.

This dissertation proposes, implements, and evaluates an Artificial Intelligence-based system for the automated generation of technical clinical pharmacology reports. The developed architecture combines Retrieval-Augmented Generation techniques with a Knowledge Graph constructed from regulatory sources, following a hybrid retrieval strategy: structured pharmacological facts are obtained from the graph, while relevant biomedical literature is retrieved through semantic vector search over indexed documents from public repositories, including PubMed, ClinicalTrials.gov, European Medicines Agency (EMA), and National Institute for Health and Care Excellence (NICE). The system supports three report categories, those being, Single Drug Evaluation, Comparative Evaluation, and Initiation, Continuation or Suspension Criteria and operates in a modular fashion, generating each report section independently according to specific retrieval configurations and prompt instructions. The entire pipeline is containerized using Docker, enabling straightforward deployment in a hospital environment without complex infrastructure requirements.

System evaluation follows a two-phase approach: automated assessment using pipeline metrics (*Ragas*), textual and semantic similarity metrics (BERTScore, ROUGE-L, Med-F1, and Semantic Similarity), and human evaluation conducted by two clinical pharmacologists from the unit. Automated results demonstrate that the generated reports preserve the clinical meaning of reference documents, with Semantic Similarity scores between 0.88 and 0.89 across all report categories. Pipeline metrics reveal stable Answer Relevancy but variable Context Precision, indicating that retrieval noise rather than generation failure is the primary bottleneck. Sections grounded in the Knowledge Graph consistently outperformed those relying solely on vector retrieval across all experimental configurations. Human evaluation confirmed the system's potential as a decision-support tool, with both evaluators acknowledging its practical utility following a detailed discussion of the architecture and its current constraints.

The system demonstrates that it is possible to automatically generate semantically grounded and structurally coherent pharmacology reports from open-access sources, reducing the workload associated with literature retrieval and report drafting, while preserving the clinical pharmacologist as the final decision-maker in the clinical process.

Keywords: Retrieval-Augmented Generation • Large Language Models • Clinical Pharmacology • Clinical Decision Support Systems • Biomedical Natural Language Processing • Medical Information Retrieval • Generative Artificial Intelligence • GraphRAG • AI in Healthcare

The United Nations Sustainable Development Goals

The United Nations Sustainable Development Goals (SDGs) provide a global framework for achieving a better and more sustainable future for all. It includes 17 goals (https://sdgs.un.org/goals) to address the world's most pressing challenges, including poverty, inequality, climate change, environmental degradation, peace, and justice. This dissertation contributes to the following Sustainable Development Goals through the development of an AI-powered system for the automated generation of clinical pharmacology reports, supporting evidence-based medicine at Hospital de São João.

SDG 3 Ensure healthy lives and promote well-being for all at all ages.

SDG 9 Build resilient infrastructure, promote inclusive and sustainable industrialization and foster innovation.

SDG 17 Strengthen the means of implementation and revitalize the global partnership for sustainable development.

SDG	Target	Contribution	Performance Indicators and Metrics
3	3.8	Automating the drafting of technical pharmacology reports reduces the time burden on clinical pharmacologists, enabling faster and more consistent evidence-based recommendations to support clinical decision-making at UFC/HSJ.	Reduction in report drafting time; semantic similarity of generated reports to expert-written ground truth (Semantic Similarity ≥ 0.88 across all report types).
	3.d	Improving the reliability and traceability of pharmacological evidence synthesis strengthens the capacity of health institutions to manage drug evaluation requests, supporting safer prescribing decisions.	Faithfulness scores in RAGAS evaluation; rate of hallucination detection flagged during human evaluation.
9	9.5	The proposed hybrid RAG architecture integrating Knowledge Graphs, vector databases, and LLMs constitutes an innovative contribution to the application of AI in clinical NLP, advancing research infrastructure for medical document automation.	Number of configurations evaluated (10); adoption of open-source and reproducible pipeline components (Neo4j, PostgreSQL/pgvector, Ollama).
17	17.6	The collaboration between FEUP, ISEP, and HSJ/UFC instantiates a multi-institutional knowledge-sharing partnership between academia and a public health system, fostering cross-sector technology transfer in AI-assisted clinical decision support.	Three institutional affiliations involved (FEUP, ISEP, HSJ); co-supervision structure spanning engineering and clinical pharmacology expertise.
	17.16	The system relies exclusively on open-access data sources and open-source tools (PubMed, ClinicalTrials.gov, EMA, FDA, Neo4j, PostgreSQL), promoting a multi-stakeholder model for the development of sustainable health technology.	Number of open-access data sources integrated (5+); system stack built entirely on open-source components with no proprietary dependencies.

Acknowledgements

Looking back at my path as a student, I see a journey that was often difficult and exhausting but somehow always deeply satisfying. The challenges I faced during the past five years pushed me to grow and the small victories along the way reminded me why I started this whole journey. This dissertation is the fruit of many sleepless nights and the never ending belief in success through hard work. This marks the end of another chapter and I am excited for what comes next. To everyone who was there along the way, this is for you.

I begin by thanking my supervisors, Professor Carlos Ferreira and Professor Rui Camacho, for their guidance, availability and the trust they gave me to carry out this work. Their insights and their ability to push me to consider different approaches/perspectives and strive for higher standards were indispensable throughout this work.

I want to thank Dr.João Mendes and Dr.Francisco Portal from the Unidade de Farmacologia Clínica at Hospital de São João, whose willingness to engage with the project shaped its development, improvements and evaluation. Their time and feedback helped grounding this work in real clinical practice.

To my amazing girlfriend, Inês, the person who has meant the most to me throughout this entire journey. There are no words that can describe what her presence meant during these years. Through every challenge she faces, she never stopped fighting for the future she deserves. Watching her do that, every single day, gave me motivation to keep going on the days I didn't feel like it. She was my calm when things felt overwhelming and my greatest supporter when success felt far away. I have no doubt that she will be the best doctor her future patients could ever ask for. It is a privilege to have been able to share this chapter with her.

To my father Alcides and my mother Rosângela, my role models and the foundation of everything I am. There is no version of this achievement that does not begin with them. They taught me the value of hard work, honesty and perseverance not through words alone but through the example they set every single day. They sacrificed quietly and consistently always believing in a better future even when it was not easy to see. Everything I have worked for, every late night, every difficult moment pushed through, every small achievement celebrated, was built on the ground they prepared for me. I hope this makes them proud, because making them proud has always been one of my greatest motivations. Thank you for being exactly who you are. I love you.

To my brothers Miguel and Joana, thank you for always being there. Through every up and down of these past years, I always knew I could count on you both. Your support meant more than you probably realize. Having you by my side throughout this journey made it all easier and a lot more meaningful.

Finally, to all my friends, thank you for the laughter, the company and the support along the way. You made the difficult days lighter and the good ones even better. I am grateful for each one of you.

João Sequeira

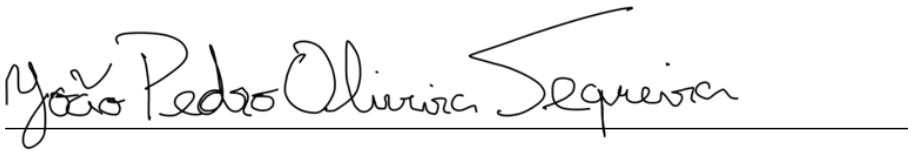
Declaration of honour

I declare that this dissertation is my own work and has not previously been submitted or assessed at this or any other institution. References to other authors (statements, ideas, thoughts), as well as the use of published works, strictly comply with the rules of attribution and are duly identified in the text and in the bibliographic references, in accordance with the applicable referencing standards. I am aware that the practice of plagiarism or self-plagiarism constitutes an academic offence, as established in the U.Porto Code of Ethics.

I further declare that I have included text(s) published in co-authorship, and that I have specified, with transparency and rigor, the contribution of each co-author, as well as stated whether there exists any other dissertation in which the same text(s) may also have been included. The authorization from the journal in which the work is published is also included, together with the express agreement signed by all co-authors authorizing the inclusion of the article.

I also declare that I have used Artificial Intelligence tools to complete part(s) of this dissertation, and that this use and its results are duly noted and have been carefully checked for errors, inaccuracies, unfounded attributions or other forms of bias in content and arguments, as well as determining authorship.

João Pedro Oliveira Sequeira



"Adoramos a perfeição, porque a não podemos ter; repugná-la-íamos se a tivéssemos. O perfeito é o desumano porque o humano é imperfeito."

Fernando Pessoa

Contents

1 Introduction **1**
- 1.1 Context . . . 1
- 1.2 Motivation . . . 2
- 1.3 Problem . . . 3
- 1.4 Research Questions . . . 4
- 1.5 Dissertation Structure . . . 4

2 Literature Review **5**
- 2.1 Methodology . . . 5
  - 2.1.1 Collection Process . . . 6
  - 2.1.2 Management Tools and Collections . . . 8
  - 2.1.3 Inclusion and Exclusion Criteria . . . 9
  - 2.1.4 Screening and Selection Process . . . 9
- 2.2 Resulting Paper Set . . . 10

3 State of the Art **11**
- 3.1 Large Language Models . . . 11
  - 3.1.1 Background and Concept . . . 11
  - 3.1.2 Transformer Architecture . . . 12
  - 3.1.3 Evaluation of LLMs . . . 15
  - 3.1.4 Limitations and Challenges . . . 17
  - 3.1.5 Pre-Training and Tuning Techniques . . . 20
- 3.2 Retrieval-Augmented Generation . . . 21
  - 3.2.1 Concept and Motivation . . . 21
  - 3.2.2 Core Components of a RAG Pipeline and Workflow . . . 22
  - 3.2.3 RAG Architectures . . . 25
- 3.3 LLMs and RAG in Biomedical,Clinical,Pharmacology Contexts . . . 26
  - 3.3.1 Overview of AI Adoption in Healthcare . . . 26
  - 3.3.2 Existing systems and studies . . . 27

4 Methodology **30**
- 4.1 Research Design and Context . . . 30
- 4.2 Business Understanding . . . 31
- 4.3 System Architecture . . . 31
  - 4.3.1 Request Handling . . . 33
  - 4.3.2 Data Layer . . . 33

4.3.3 Retrieval and Context Augmentation Pipeline . . . 35
4.3.4 Report Generation and Post-Processing . . . 36
4.3.5 Knowledge Representation and Databases . . . 38
4.3.6 Human-in-the-Loop Validation . . . 39
4.4 Evaluation Methodology . . . 40

5 Experiments and Results **43**
5.1 Experimental Setup - Automated Metrics . . . 44
5.2 Ragas Evaluation . . . 47
5.3 NLP Evaluation . . . 51
5.4 Human Evaluation . . . 56
5.4.1 Quantitative and Qualitative Findings . . . 57
5.5 Discussion . . . 60
5.5.1 RQ1: Is it possible to automatically generate reliable technical pharmacology reports using data retrieved from public repositories and hospital information systems? . . . 61
5.5.2 RQ2 — What are the main challenges regarding information retrieval, contextual grounding, and data integration? . . . 63
5.5.3 RQ3 — To what level can the integration of structured knowledge sources mitigate hallucinations and improve the accuracy of generated reports? . . . 64

6 Conclusion and Future Work **65**

References **68**

A Annexes **74**
A.1 Human-written UFC report: Belimumab para doente com Lúpus Eritematoso Juvenil . . . 74

List of Figures

3.1 Timeline of major LLM releases (2023–2025). Source: [33]. 13
3.2 Transformer Model Architecture. Source: [44] 14
3.3 Overview of challenges in LLMs and their associated issues (adapted from [22]) 18
3.4 Security Threats associated to LLMs (adapted from [34]) 20
3.5 Overview of RAG pipeline (Source: [67]) 22
3.6 Overview of vector database architectures. Source: Rao (2023) [45]. Licensed under CC BY-NC-SA 4.0. 23

4.1 Proposed System Architecture for Unidade de Farmacologia Clínica (UFC) use cases . . 32
4.2 Web application interface for pharmacologist review and editing of generated reports . . 40

5.1 Ragas Metrics Performance by Report Type 50
5.2 Score per Report Section x Ragas Metric 50
5.3 Average BERTScore performance across distinct sections within Single Drug Evaluation reports. 52
5.4 Average BERTScore performance across distinct sections within Comparative Evaluation reports. 53
5.5 Average BERTScore performance across distinct sections within Initiation and Suspension Criteria reports. 53
5.6 Aggregated BERTScore distributions categorized by comprehensive report type. 54
5.7 Average BERTScore per section aggregated across all report types 55
5.8 Mean Expert Evaluation Scores by Report Type 58
5.9 Overall Mean Evaluation Score by Report Type 58
5.10 Qualitative Feedback from Open Text Comments 59

List of Tables

2.1 QS for Large Language Models (LLMs). 6
2.2 QS for Retrieval-Augmented Generation (RAG). 7
2.3 QS for Combined LLM and RAG in Pharmacological, biomedical and clinical fields. . . 8
2.4 Inclusion and Exclusion Criteria applied during literature retrieval. 9

3.1 Selected LLM evaluation benchmarks across different domains (adapted from [7]) 17

5.1 Experimental Configurations Evaluated 45
5.2 Overall Ragas performance across different experimental setups. 48
5.3 Semantic Similarity evaluated across different massive-context embedding models configuration. 55
5.4 Global NLP document-level evaluation metrics categorized by pharmaceutical report type. 56
5.5 Human evaluation questionnaire (quantitative items) 57

List of Acronyms

AI Artificial Intelligence
ANN Artificial Neural Network
API Application Programming Interface
BLEU Bilingual Evaluation Understudy
BPE Byte-Pair Encoding
CFT Comissão de Farmácia e Terapêutica
CNN Convolutional Neural Networks
EHR Eletronic Health Records
EMA European Medicines Agency
GELU Gaussian Error Linear Unit
Graph-RAG Graph Retrieval-Augmented Generation
HSJ Hospital de São João
INN International Nonproprietary Names
IR Information Retrieval
KG Knowledge Graph
LLM Large Language Model
LoRA Low-Rank Adaptation
LSTM Long Short-Term Memory
MLM Masked LM
NER Named-Entity Recognition
NICE National Institute for Health and Care Excellence
NLP Natural Language Processing
NTP Next Token Prediction
PEFT Parameter-Efficient Fine-Tuning
RAG Retrieval-Augmented Generation
RCM Summary of Product Characteristics
ReLU Rectified Linear Unit
Regex Regular Expressions

RLHF Reinforcement Learning from Human Feedback

RNN Recurrent Neural Network

ROUGE Recall-Oriented Understudy for Gisting Evaluation

SDG Sustainable Development Goal

SoTA State of The Art

UFC Unidade de Farmacologia Clínica

Chapter 1

Introduction

This chapter introduces the context and motivation for the development of an Artificial Intelligence (AI)-based system to support the automatic generation of technical pharmacology reports. It addresses the role of clinical pharmacology in modern healthcare, the challenges associated with managing large volumes of biomedical data and the use of Large Language Models (LLMs) to support information search workflows. Attention is given to the limitations of current AI systems, such as hallucinations and lack of domain-specific reliability, and the use of Retrieval-Augmented Generation (RAG) techniques to mitigate these issues.

Section 1.1 provides the contextual background, introducing key concepts in clinical pharmacology, LLMs, and RAG, as well as challenges surrounding the use of AI in clinical settings. Section 1.2 presents the motivation for this research referring the need for tools that can improve efficiency, consistency, and reliability in pharmacological workflows. Section 1.3 defines the problem addressed by this dissertation and Section 1.4 outlines the research questions that guide the development and evaluation of the proposed system. Lastly, Section 1.5 describes the structure of the remaining dissertation structure.

1.1 Context

Clinical Pharmacology plays a critical role in contemporary healthcare systems. It encompasses all aspects in the relationship between drugs and humans while focusing on the safe, effective and economic use of medicines [6]. One of its main goals is to generate data for optimum use of drugs and the practice of 'evidence-based medicine'. In order to achieve this, clinical pharmacology follows a series of protocols that typically involve collecting, analyzing and synthesizing large volumes of heterogeneous information from scientific literature, clinical studies and regulatory guidelines. This process is currently both labor-intensive and time-consuming and requires timely outputs done by professional experts.

The never-ending growth in data in the biomedical field, combined with the tight guidelines a pharmacologist must comply, further extends the amount of needed time and therefore the workload required

to create technical reports. The rise of AI presents a golden opportunity to aid the pharmacology community in tackling this overwhelming issue, especially with the emergence of LLMs that have shown to be a useful tool in aiding the automation of information-intensive work. LLMs can be defined as a set of machine learning models (computer programs that can find patterns or make decisions) that can process and generate human language text [10]. They work by analyzing massive data sets of language and are one of the hottest topics in a lot of scientific fields. At present, they help solve issues in the most diverse areas of studies as well as everyday activities and even serve as personal assistant technology. These models, however, like humans, are not perfect. One of their biggest flaws is an issue called hallucination. Hallucination is characterized by being a phenomenon where LLMs, often generative AI or a computer vision tool, perceive patterns or objects that are nonexistent or imperceptible to human observers, creating outputs that are nonsensical or altogether inaccurate [3]. Some other challenges include giving out-of-date or generic answers when the user expects a specific response or even creating inaccurate responses due to terminology confusion (the use of the same terminology to talk about different things). To help mitigate these occurrences, there is a technique called RAG that allows the optimization of LLMs output by providing more context in the corresponding field. It references knowledge bases that are outside of the LLMs training data (the foundation of its intelligence) before generating responses. It allows for a more topic-specific context window extending the dynamic capabilities of LLMs without needing additional training (cost-effective approach) [4].

Right now, there is a major lack of confidence in using AI for this field. The current use of AI in clinical settings is limited by lack of clinician confidence and understanding of AI. Concerns around safety and applicability to clinical practice, and a lack of pathways for how these technologies will undergo regulation and evaluation [47].

Nonetheless, the combination of these two tools (RAG systems and LLMs) offers a unique opportunity to accelerate and standardize the creation of pharmacology technical reports while maintaining the required clinical rigor.

1.2 Motivation

The increasing volume of information and its inherent complexity are a strong challenge for the work of clinical pharmacologists and it highlights the need for solutions that can facilitate their methods. As mentioned above, pharmacologists follow a workflow that is time sensitive and requires the development of reliable, up-to-date scientific and regulatory information. Any sort of delay following this process can lead to inconsistencies and even put the patient's well-being at risk [11]. Going beyond possible delays, the current pharmacological workflow relies deeply on manual effort both in the report writing as well as in the retrieval of valuable and reliable information to support their reports, which is a laborious task for professionals and requires substantial personnel costs to sustain this whole process [18].

Developing an intelligent system that can ease their workload might offer an opportunity to enhance their level of productivity and consistency in building their technical reports. The integration of RAG

systems and LLMs generates a prospect of automating the time-consuming aspects such as literature retrieval and information synthesis that would allow professionals to focus more on the technical and judgment-based analysis of results.

This research will also contribute to a growing connection between AI systems and clinical pharmacology. Exploring how RAG-enhanced LLMs behave in the generation of technical pharmacology reports gives the scientific community a way to evaluate the applicability of these models in real-world clinical settings.

By addressing these adversities, this thesis aims to exhibit how AI systems can improve efficiency, standardization and quality in the creation of technical pharmacology reports, supporting better clinical decision-making and patient care.

1.3 Problem

The problem is contextualized within the ongoing workflow of the Unidade de Farmacologia Clínica (UFC) of Hospital de São João (HSJ),Porto. Within this unit, pharmacologists are tasked with the production of technical-scientific reports in response to requests submitted by Comissão de Farmácia e Terapêutica (CFT), such as the evaluation of therapeutic alternatives or gathering and assessing scientific evidence for specific clinical cases. The building of these reports requires systematic consultation of extensive documentation, a process that relies entirely on manual procedures. This workflow is also very time-consuming and is associated with high personnel costs to maintain. With the constant increase in the literature being published (studies indicate that up to 6000 papers are published per day [20]) it becomes a challenge to ensure the finding of correct and relevant information.

To address this problem, this thesis investigates the integration and evaluation of the use of RAG systems in the automatic production of technical pharmacology reports. More notably, the purpose of this research work is to design, develop and assess an AI-based system capable of automatically drafting technical reports through retrieval and generation mechanisms, as a means of addressing a real-world problem associated with the production of pharmacology reports. The goal is to create a modular and scalable system capable of communicating with external Application Programming Interfaces (APIs) and data sources that produce accurate and context-aware reports through the combination of information retrieval and generative mechanisms. To achieve this goal, there will be a study on analyzing the current procedures followed by pharmacologists involved in evaluating and producing technical reports, the identification of requirements and the creation of a functional prototype that retrieves information from reliable and renowned public repositories such as Pubmed and Infarmed. The evaluation will focus on verifying the system's performance in terms of quality of the generated information, factual accuracy and usefulness.

Within this context, the proposed system aims to act as a decision-support tool, assisting pharmacologists in the retrieval and synthesis of evidence, and reducing the burden of repetitive manual tasks while preserving scientific rigor and clinical relevance.

1.4 Research Questions

The following questions are addressed by this dissertation:

RQ1 Is it possible to automatically generate reliable technical pharmacology reports using data retrieved from public repositories and hospital information systems?

RQ2 What are the main challenges regarding information retrieval, contextual grounding, and data integration when using RAG-enhanced LLMs for clinical pharmacology report generation?

RQ3 To what level can the integration of structured knowledge sources, such as knowledge graphs, mitigate the occurrences of hallucination from the LLMs and improve the accuracy of generated reports?

1.5 Dissertation Structure

This thesis proposal report contains five more chapters, each building on the previous one to provide a comprehensive view of the research. Chapter 2 describes the literature review methodology and resulting paper set. Chapter 3 presents the state of the art in LLMs, RAG systems and their applications in biomedical contexts. Chapter 4 presents the design and implementation of the proposed architecture, including the construction of the Knowledge Graph, retrieval pipeline, and report generation framework. Chapter 5 presents the experimental setup, evaluation methodology, results, and discussion of the system's performance. Finally, Chapter 6 summarizes the main contributions of this work, describes its limitations and outlines directions for future research.

Chapter 2

Literature Review

This chapter documents the process followed to identify, gather and analyze literature in the areas of LLMs, RAG and the work done with the concatenation of these two topics with focus within the biomedical and pharmacology fields. The purpose of this review is to lay a solid knowledge foundation, both theoretical and practical, that supports the development of the system proposed in this dissertation as well as to identify any trending implementations in the use of RAG-enhanced LLMs in the biomedical context as well as any research gaps within the biomedical domain regarding this topic.

Section 2.1 and its respective subsections, give insight on the methodology followed for this review that combines some characteristics of both systematic and exploratory approaches to find and collect relevant pieces of literature. Section 2.2 will present some insight on the final group of papers selected through this process.

2.1 Methodology

This literature review followed a hybrid approach meaning that it combines techniques from systematic search methods as well as some exploratory strategies in order to create a process that is reproducible and more importantly a flexible and relevant collection of papers that captures both fundamentals and trending ideas in the areas of LLMs and RAG and their combination in the biomedical field.

The methodology is divided into three main phases: collection process of the literature, application of inclusion and exclusion criteria and finally the screening and selection of the most relevant papers.

Each phase contributed to the retrieval of a comprehensive and robust dataset ensuring that the final collection of papers was both relevant to this research objectives and also representative of the state of knowledge in the fields of LLMs and RAG in addition to their applications in biomedical contexts.

The combination of systematic and exploratory techniques allowed the inclusion of foundational works, trending methodologies and some practical implementations while providing a solid starting point for discussion in the state-of-the-art chapter. Each phase characteristics are thoroughly detailed below in each subsection.

2.1.1 Collection Process

The collection of literature started with a systematic search of academic databases, as of which, IEEE Xplore [1], Scopus [2] and ACM Digital Library [3]. This search focused on the retrieval of publications across three main topics: information referring the foundations of LLMs knowledge as well as its applications, RAG systems studies and architectural analyzes and their applications and lastly a discovery of works that explore the integration of these two technologies in clinical, biomedical or pharmacological contexts. In order to perform this search, three queries were created to target the main topics referred. Regarding the first one, corresponding to the LLMs domain, three variations of essentially the same query were created as seen in Table 2.1 for the different databases.

Table 2.1: QS for Large Language Models (LLMs).

Database	**Search String**
IEEE Xplore	*(("large language model" OR "foundation model") AND (architecture OR "model design" OR transformer OR "decoder-only" OR pretraining OR "fine-tuning" OR training OR alignment OR optimization OR application) AND (survey OR overview OR taxonomy OR "systematic review" OR tutorial) NOT (vision OR image OR video OR speech OR audio OR multimodal))*
Scopus	*TITLE-ABS-KEY (("large language model" OR "foundation model") AND (architecture OR "model design" OR transformer OR "decoder-only" OR pre-training OR "fine-tuning" OR training OR alignment OR optimization) AND (survey OR taxonomy OR "systematic review") AND ("natural language" OR NLP OR "text generation" OR "language modeling") AND NOT (vision OR image OR video OR speech OR audio OR multimodal)) AND PUBYEAR > 2021 AND PUBYEAR < 2027 AND PUBYEAR > 2021 AND PUBYEAR < 2027 AND (LIMIT-TO (SUBJAREA , "COMP") OR LIMIT-TO (SUBJAREA , "ENGI")) AND (LIMIT-TO (DOCTYPE , "cp") OR LIMIT-TO (DOCTYPE , "ar")) AND (LIMIT-TO (LANGUAGE , "English"))*
ACM Digital Library	*("large language model" OR "foundation model" OR "large-scale pretraining") AND (architecture OR "model design" OR "transformer architecture" OR "decoder-only" OR application) AND ("systematic review" OR "literature review" OR taxonomy) AND ("natural language processing" OR NLP OR "text generation" OR "language modeling") AND ("GPT" OR "ChatGPT" OR "BERT" OR "T5" OR "LLaMA") AND NOT vision AND NOT image AND NOT audio AND NOT multimodal*

These variations come mostly from the need to accommodate the search syntax and field constraints specific to each digital library. Some fields were also added in the Scopus and ACM Digital Library search strings in order to reduce noise and remove unnecessary papers that referred to other areas of

[1] https://ieeexplore.ieee.org/
[2] https://www.scopus.com/home.uri
[3] https://dl.acm.org/

science and retrieve mostly relevant papers according to the previously defined search goal. Despite this, they share common keywords aimed at getting mostly systematic reviews, taxonomies and surveys.

As for the second topic, RAG systems, a similar process was followed with different but appropriate keywords as verified in Table 2.2

Table 2.2: QS for Retrieval-Augmented Generation (RAG).

Database	Search String
IEEE Xplore	*("retrieval-augmented generation" OR "retrieval augmented generation" OR "RAG" OR "retrieval-augmented language model" OR "knowledge grounding") AND ("large language model" OR "foundation model" OR transformer OR application) AND (architecture OR framework OR pipeline OR system OR "system design") AND ("natural language processing" OR NLP OR "text generation" OR "document retrieval") AND (evaluation OR benchmark OR performance OR comparison) NOT (vision OR image OR audio OR video OR speech OR multimodal)*
Scopus	*TITLE-ABS-KEY ("retrieval-augmented generation" OR "RAG" OR "retrieval-augmented language model") AND TITLE-ABS-KEY ("large language model" OR "foundation model" OR transformer) AND TITLE-ABS-KEY (architecture OR framework OR pipeline OR system OR "system design" OR "implementation") AND TITLE-ABS-KEY (evaluation OR benchmark OR performance OR "case study") AND TITLE-ABS-KEY ("retrieval module" OR "dense retrieval" OR "vector database" OR "document retriever") AND NOT TITLE-ABS-KEY (vision OR image OR video OR audio OR multimodal OR speech OR demo OR chatbot) AND PUBYEAR > 2021 AND PUBYEAR < 2027 AND (LIMIT-TO (SUBJAREA , "COMP") OR LIMIT-TO (SUBJAREA , "ENGI")) AND (LIMIT-TO (DOCTYPE , "ar") OR LIMIT-TO (DOCTYPE , "cp")) AND (LIMIT-TO (LANGUAGE , "English"))*
ACM Digital Library	*("retrieval-augmented generation" OR "RAG" OR "knowledge grounding" OR "context augmentation" OR "retrieval-enhanced language model" OR "retrieval-augmented language model") AND ("large language model" OR "foundation model" OR "transformer" OR application) AND (architecture OR framework OR pipeline OR system OR design) AND ("natural language processing" OR NLP OR "text generation" OR "document retrieval") AND NOT vision AND NOT image AND NOT audio AND NOT multimodal*

Identically, as on the previous query, the search strings differentiate from database to database due to platform-specific limitations regarding syntax. It's important also to highlight the use of `NOT (vision OR image OR audio OR video OR speech OR multimodal)` in each string to attempt to isolate more text generation oriented results.

Lastly, a query was created in order to assess the state of integrating both LLM and RAG tools in the context of biomedical, clinical or pharmacological domains as displayed in Table 2.3. Here, the main differences, come from adapting the biomedical terminology and filtering mechanisms to each database's capacities.

Table 2.3: QS for Combined LLM and RAG in Pharmacological, biomedical and clinical fields.

Database	**Search String**
IEEE Xplore	*("large language model" OR "foundation model") AND ("retrieval-augmented generation" OR "RAG") AND ("clinical" OR "biomedical" OR "pharmacology") AND (document generation OR technical report OR literature synthesis OR evaluation OR benchmark OR performance OR application)*
Scopus	*TITLE-ABS-KEY (("large language model" OR "foundation model") AND ("retrieval-augmented generation" OR "RAG") AND (clinical OR biomedical OR pharmacology) AND ("document generation" OR "technical report" OR "literature synthesis" OR evaluation OR benchmark OR performance OR application) AND NOT (vision OR image OR video OR audio OR speech OR multimodal)) AND (LIMIT-TO (SUBJAREA , "MEDI") OR LIMIT-TO (SUBJAREA , "COMP") OR LIMIT-TO (SUBJAREA , "ENGI")) AND (LIMIT-TO (LANGUAGE , "English"))*
ACM Digital Library	*("large language model" OR "foundation model") AND ("retrieval-augmented generation" OR "RAG") AND ("clinical" OR "biomedical" OR "pharmacology") AND ("document generation" OR "technical report" OR "literature synthesis" OR evaluation OR benchmark OR performance OR application) AND NOT audio*

Disclaimer: Although not present in each search string, a filter, concerning publication year, was applied in each database from the year 2020 to the present year (2025). These search strings can be copied to each one of the digital libraries mentioned and performed using the advanced search feature on those websites. Small note for the ACM Digital Library search that was made using the advanced search, selecting the "anywhere" search filter and pasting the respective search string.

2.1.2 Management Tools and Collections

To store every result obtained through this systematic search, Zotero [4] was used as the main reference management tool for importing the documents gathered, storing them and organizing them into sections similarly to the tables logic (three groups - LLM, RAG, Combination of both in the biomedical field).

[4] https://www.zotero.org/

2.1.3 Inclusion and Exclusion Criteria

To assure that the literature review included uniquely the most relevant studies that aligned with the proposed search methodology while keeping high standards of quality and reliability, some filters were determined and exhibited in Table 2.4

Table 2.4: Inclusion and Exclusion Criteria applied during literature retrieval.

Inclusion Criteria	**Exclusion Criteria**
Studies focusing on LLMs or RAG systems, including their architecture, optimization, training, evaluation, or applications.	Works primarily addressing multimodal models (vision, audio, or speech) rather than text-based language processing.
Publications exploring the integration of LLMs and RAG systems in biomedical, clinical, or pharmacological domains.	Duplicates or papers with incomplete bibliographic information.
Systematic reviews, surveys, taxonomies, tutorials, or case studies relevant to LLM or RAG development.	Opinion pieces, editorials, or non-peer-reviewed content.
Peer-reviewed journal articles, conference papers, and workshop proceedings.	Papers not written in English.
Publications between 2020 and 2025.	Papers lacking methodological or experimental detail to assess validity.

The creation of filtering criteria established the basis for the subsequent screening and selection phase.

2.1.4 Screening and Selection Process

The screening and selection process starts with the removal of duplicate entries, facilitated by Zotero's functionality to proceed with this method. Given the task of identifying both foundational and upcoming trends in the fields of LLMs and RAG, a semi-structured and iterative approach was adopted. This process involved several stages. For starters, a review of the titles and abstracts of all the retrieved papers was done in order to verify their alignment with this research objectives. In cases where the abstract missed some clarity or was too generic but the title showed that the paper had potential relevance, a quick review was performed in the introduction and conclusion sections. As a result, papers that might have been discarded due to weak abstracts were instead retained.

Additionally, references of specific relevant papers were examined to gather highly cited researches and collect further related works that might not have been captured by the initial iteration, expanding the final dataset through a brief backwards snowballing stage.

2.2 Resulting Paper Set

Starting from over a thousand papers retrieved through all the queries performed, both the inclusion and exclusion criteria and the screening and selection process were concluded. From them, a final dataset was generated containing the most relevant papers. This dataset features key topics and essential foundational material on the study of LLMs and RAG and it supplies diverse techniques exploring the integration of both technologies in biomedical and pharmacology domains.

Overall, the final selection contains about 50 papers in each query topic. Despite this elevated number, not all studies are to be discussed exhaustively in the upcoming State of The Art (SoTA) section, instead, the focus is placed on the most relevant insights, highly cited and foundational studies within each research topic.

This resulting dataset serves as a cornerstone for the SoTA chapter where the selected studies are analyzed and synthesized to highlight their findings, approaches and contributions.

Chapter 3

State of the Art

This chapter provides an in-depth analysis of the theoretical and practical foundations of this dissertation. It starts by looking into the present evolution of Generative Artificial Intelligence, particularly the advances made by LLMs (covering its historical development, architectural principles, tuning methods, evaluation tools and limitations), RAG systems importance (concept clarification, what it attempts to attenuate, the components that build a RAG pipeline and different architecture implementations as well as their strengths and flaws).

The final section covers current applications and studies that combine both AI technologies within the biomedical, clinical and pharmacology fields. It mentions existing tools, their strengths and challenges they face.

3.1 Large Language Models

3.1.1 Background and Concept

AI, at the time of writing this dissertation, stands as one of the most rapidly advancing technological fields. It has quickly transitioned from theoretical concepts and mathematical formulations into a practical set of tools that influences all sorts of areas such as finances, engineering, scientific research and even daily consumer applications [22].

Within this landscape, LLMs refer to a specific category of AI models that were developed to comprehend and produce human language [30]. Although the term "LLM" has become popular in recent years, it is not a new invention. Work regarding the study and creation of LLMs can be traced to as early as the 1940s with the idea of Artificial Neural Networks (ANNs) that later became the foundation to the early first language models that were mostly rule-based or statistical [28]. They relied on manual engineered rules and probabilities that lacked on capturing semantic relationships.

The 2000s presented breakthroughs in the area of Natural Language Processing (NLP) with the introduction of word representations which proved to be an effective manner at storing semantic meaning between words in phrases. Later, this innovation, became an important component of LLMs. The

following decade also showed many advances in this AI category with the creation of new models that featured deep learning techniques to retrieve information and utilized ANNs to understand and produce human readable text marking a transition period between rudimentary rule-based toward scalable neural approaches [44].

The turning point happens between the years 2015 and 2017 with Google's groundbreaking research that introduced their Neural Machine Translation model (NMT) and the introduction of the Transformer architecture. This architecture replaced recurrent structures with a self-attention mechanism allowing parallel sequence processing and a better handling of long-range dependencies [58]. It provided the basis for the emergence of LLMs and is,to this day, the main architecture used in state-of-the-art generative models [56].

Beyond their historic evolution, it is of interest to elucidate the concept of Large Language Models. An LLM, at its core, represents an algorithm that is capable of adapting to several NLP tasks such as sentiment analysis, named entity recognition, text translation and many other cases. They are neural network-based models trained on enormous quantities of textual data (mostly, not exclusively) to learn linguistic patterns and representations, enabling them to generalize across diverse tasks without requiring task-specific programming [44].

This capability is fundamentally enabled by the autoregressive training mechanism inherent to LLMs. Language models construct their output based on probabilities. They actively try to predict which token should appear next based on the preceding tokens allowing the LLM to produce text that is coherent and contextually appropriate depending on the input.

To sum up, LLMs are scalable, general-purpose generative models that leverage correlations gathered from massive textual corpora to perform language-based tasks. Their capacity in handling increasing contextual complexity and being able to adapt across multiple fields while keeping its performance makes them a foundational component of modern AI.

The rapid evolution of this field is exemplified by the numerous prominent models released in recent years. Figure 3.1 illustrates a timeline of major LLM releases between 2023 and 2025, demonstrating the accelerated pace of development in this domain.

3.1.2 Transformer Architecture

Before the introduction of the Transformer architecture, most NLP systems relied on Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) networks. Both these systems posed a significant bottleneck regarding text processing, which processed text sequentially thus leading to a loss of dependencies in long contexts, limited task parallelism and increasing costs as the sequence length grew in size. The Transformer architecture, proposed by Vaswani et al. , presented as a new architecture that would address these constraints. The innovation was the introduction of the self-attention mechanism that replaces recurrence entirely and enabled the parallel processing of all the tokens in a sequence while keeping contextual relationships [58].

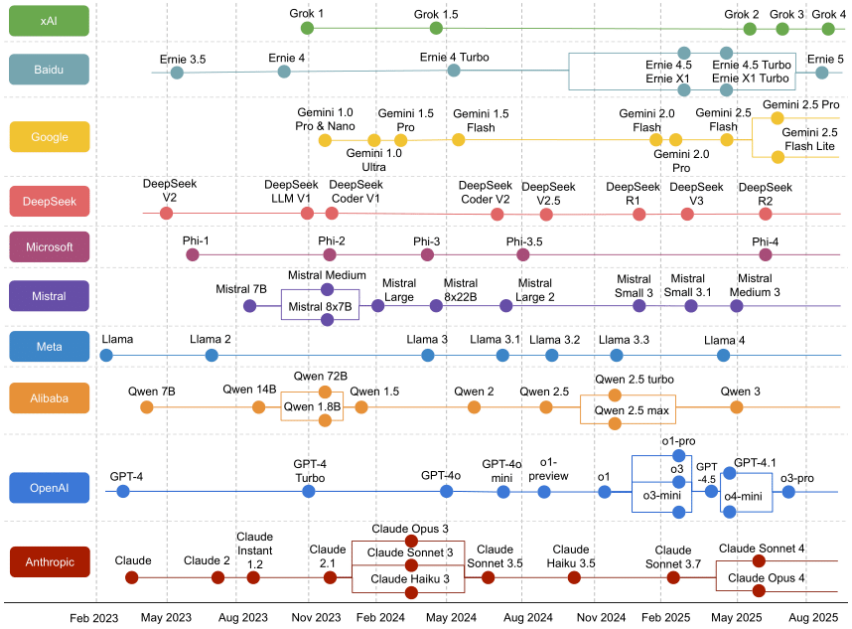


Figure 3.1: Timeline of major LLM releases (2023–2025). Source: [33].

Figure 3.2 illustrates the Transformer architecture and its components. The main components are: tokenization and embedding layers who convert input text into numerical representations, positional encodings that capture sequence order, multi-head attention mechanisms that model relationships between tokens, and feed-forward networks that apply non-linear transformations. The following subsections detail each of these components.

3.1.2.1 Tokenization

The first stage of this architecture starts with the feeding of input. This input has to be converted into what it's called tokens so that they can be processed by the model [12]. These tokens can come from characters, subwords, symbols or words depending on the nature and dimension of the language model. Traditional NLP systems made use of word-level tokenization but modern LLMs operate at a subword level using algorithms such as Byte-Pair Encoding (BPE), WordPiece, SentencePiece. Working with subword tokenizaton brings three main advantages: shorter encoding of frequent tokens, compositionality of subwords, and ability to deal with unknown words.

The output from tokenization consists on a sequence of tokenIDs that match the model's vocabulary. These integer sequences serve as input to the subsequent embedding layer, which transforms them into dense vector representations that the Transformer can process.

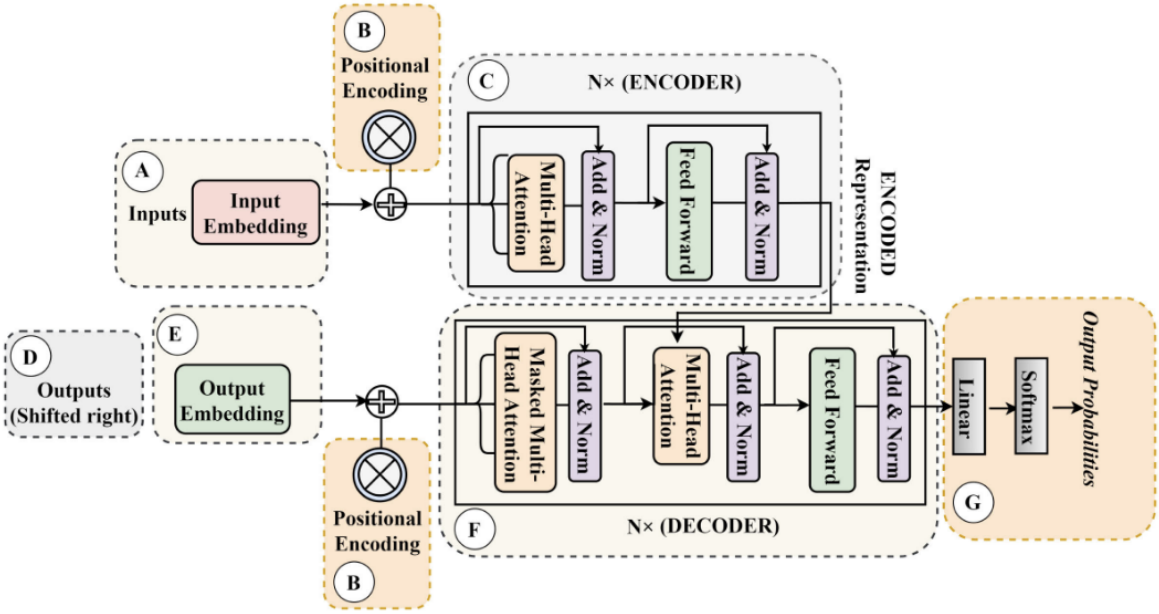


Figure 3.2: Transformer Model Architecture. Source: [44]

3.1.2.2 Embeddings

Machine learning models only understand numerical data which is why tokenization is a crucial first step to this process [44]. After it, the model now needs to recognize the meaning of those tokens and it does so by transforming each tokenID into a continuous vector representation. These embeddings are learned during training and stay within a high-dimensional vector space where tokens with similar meanings are located in close proximity. This feature enables the model to capture and leverage semantic relationships during processing.

3.1.2.3 Positional Encodings

Despite the fact that embeddings capture the semantic meaning between tokens, they do not process the order in which the tokens appear because unlike previous adaptations such as RNN and Convolutional Neural Networks (CNN), the Transformer architecture processes tokens in parallel and therefore lose the word position. This quickly becomes a problem since word order matters for meaning purposes (for instance - "cat ate fish" is different from "fish ate cat", although it is the same set of words).

To address this, positional encodings are added. It is a similar process to the encoding phase, however, here, the position of the words is encoded and transformed into a collection of integers.

The combination of these three stages gives us a vector representation that contains both the semantic content (3.1.2.2) and positional information (3.1.2.3).

3.1.2.4 Multi-head attention and Feed-forward networks

The multi-head attention mechanism is the great innovation brought by this architecture. Thanks to it, developers had now a method that could compute contextual relationships between tokens independently on their distance in the token sequence. The procedure, called attention function, takes query vectors, key vectors and value vectors as inputs and returns a tensor (multi-dimensional array of numbers) of same length as input representing their similarity. Multi-head attention extends this process by parallelizing the inputs into multiple attention heads. Each "head" then learns to grasp different aspects of the relationships between tokens.

Following this mechanism, each token representation is processed in a feed forward network and undergoes linear transformations with an activation function.The activation function is typically either Rectified Linear Unit (ReLU), that outputs the input if positive and zero otherwise, or Gaussian Error Linear Unit (GELU), which provides a smoother approximation by weighing inputs based on their magnitude [44].

As a result of this, non-linearity is introduced to the model allowing it to learn complex patterns beyond simple linear relationships.

3.1.3 Evaluation of LLMs

Evaluating a LLM is a complex task that can even be comparable to developing it [36]. As these models become progressively integrated in real-world applications, the need for proper evaluation has grown in order to enable reliability and ensure their safety of use. As previously mentioned, LLMs are general-purpose models, meaning that they work in many domains. As a result of it, they require assessment across those wide spectrum of tasks.

Evaluation methods can be divided into two categories: Automatic Evaluation, when it can be done automatically and if not then Human Evaluation [7].

The following subsections detail each of these techniques.

3.1.3.1 Automatic Evaluation

Automatic Evaluation is the most popular evaluation method. It utilizes quantitative measures and evaluation tools in order to assess model performance. This method removes human subjectivity and offer scalability, reproducibility and benchmarks between different models.

One metric commonly used by this method is perplexity, that measures how well a model predicts the next token in a sequence [7]. It is defined by:

$$\text{Perplexity}(W) = \exp\left(-\frac{1}{N}\sum_{i=1}^{N}\log P(w_i|w_1,\ldots,w_{i-1})\right) \tag{3.1}$$

where $W = (w_1, w_2, \ldots, w_N)$ represents the sequence of tokens, and $P(w_i|w_1, \ldots, w_{i-1})$ is the probability of token w_i given the preceding context.

Lower perplexity translates to a good performing model. Other metrics are frequently used in text generation tasks, such as, Bilingual Evaluation Understudy (BLEU) and Recall-Oriented Understudy for Gisting Evaluation (ROUGE). BLEU score is calculated as:

$$\text{BLEU} = \text{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \tag{3.2}$$

where p_n is the precision of n-grams, w_n are positive weights summing to one, and BP is the brevity penalty defined as:

$$\text{BP} = \begin{cases} 1 & \text{if } c > r \\ e^{(1-r/c)} & \text{if } c \leq r \end{cases} \tag{3.3}$$

where c is the length of the candidate translation and r is the effective reference corpus length.

ROUGE-N works by measuring the overlap of n-grams between the generated text and the reference text. It is defined by:

$$\text{ROUGE-N} = \frac{\sum_{S \in \text{Reference}} \sum_{\text{gram}_n \in S} \text{Count}_{\text{match}}(\text{gram}_n)}{\sum_{S \in \text{Reference}} \sum_{\text{gram}_n \in S} \text{Count}(\text{gram}_n)} \tag{3.4}$$

where $\text{Count}_{\text{match}}(\text{gram}_n)$ is the maximum number of n-grams co-occurring in the candidate and reference texts, and $\text{Count}(\text{gram}_n)$ is the number of n-grams in the reference text.

To evaluate LLM performance across diverse tasks, large benchmark suites have been developed to assess their performance across multiple task categories. Table 3.1 presents a selection of prominent benchmarks categorized by their focus domain.

3.1.3.2 Human Evaluation

As the name suggests, human evaluation consists of methods that evaluate models through human participation. It is pivotal in cases where automatic evaluation fails to grasp attributes related to coherence, factual accuracy, relevance and especially in open-ended tasks where errors might have significant harmful outcomes (such as healthcare, for instance, where human evaluation is referred as the gold standard for the near future).

This method presents some challenges. It is resource-intensive, requiring substantial time and cost to obtain consistent and high-quality assessments. It is difficult and exhaustive to scale since evaluators might have to look at dozens of outputs and prompts and also there is a need to make sure that these evaluators are experts of the subject. Even with this requirements, human evaluation still injects subjectivity to the model, where, different evaluators due to their backgrounds might have diverging opinions regarding the same topic [7].

Table 3.1: Selected LLM evaluation benchmarks across different domains (adapted from [7])

Benchmark	**Focus Domain**	**Domain**	**Evaluation Criteria**
MMLU	Text models	General language	Multitask accuracy
HELM	Holistic evaluation	General language	Multi-metric evaluation
BIG-bench	Capabilities and limitations	General language	Model performance and calibration
MATH	Mathematical problem	Specific downstream	Mathematical ability
APPS	Coding challenge	Specific downstream	Code generation ability
CUAD	Legal contract review	Specific downstream	Legal contract understanding
MultiMedQA	Medical QA	Specific downstream	Accuracy and human evaluation
TRUSTGPT	Ethics	Specific downstream	Toxicity, bias, and value-alignment
MME	Multimodal LLMs	Multi-modal	Perception and cognition ability
SEED-Bench	Multimodal LLMs	Multi-modal	Generative understanding of MLLMs
MM-Vet	Complicated multi-modal	Multi-modal	Integrated vision-language capabilities

Recent studies have explored hybrid approaches to mitigate some of the challenges that human evaluation faces [55] [9]. These systems first generate preliminary evaluations that are then assessed by human evaluators who validate or refine the outputs. When a certain level of alignment between the evaluators and model outputs is reached, the model can be used to scale the process more efficiently, not discarding human oversight.

3.1.4 Limitations and Challenges

LLM have been attracting an increasingly amount of people which has led to this massive and sudden rise of technological dependency. Although they've made significant contributions to various fields, they have noteworthy limitations and challenges that should be taken into account by any end-users.

Figure 3.3 provides a comprehensive overview of the challenges discussed in this section.

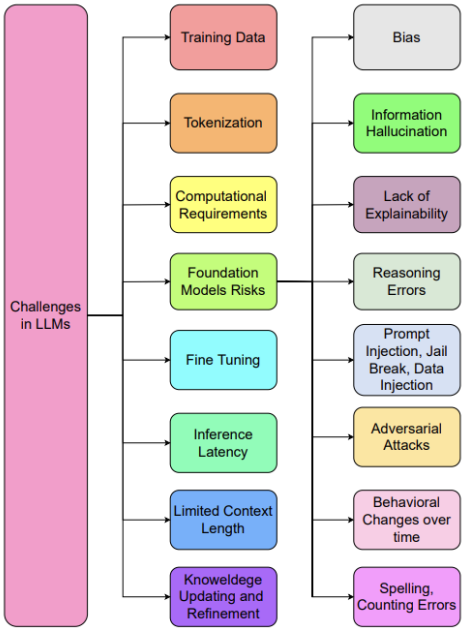


Figure 3.3: Overview of challenges in LLMs and their associated issues (adapted from [22])

The following subsections describe some of these challenges with emphasis on technical and contextual constraints, reliability and safety.

3.1.4.1 Technical and Contextual constraints

Modern LLMs can reach up to 300 billion parameters or more [51]. In order to train a LLM a humongous amount of energy and hardware is used. This becomes unfathomable for resource-constrained environments who might want to deploy a LLM since they won't have access to the required computational resources. Currently a new concept, Computer Optimal Training, was introduced to address this problem for more efficient training taking into consideration the corpus and model size [27].

Another issue can be highlighted regarding the context length of these models. LLMs have a limited number of tokens that they "keep in memory" due to their limited context window. This is especially an issue when working with large documents since without being able to keep the contextual information the model leading to degradation in performance. Related to the context window, LLMs, during the generation process, are strongly influenced by the training data that they possess. The scale and diversity required for effective training make it challenging to assess data quality comprehensively, potentially introducing biases or knowledge gaps that affect model behavior [21].

3.1.4.2 Reliability

One of the most critical problems LLMs face is information hallucination. These models have the goal of, as described before, predicting the next token based on statistical patterns learned from large text corpora. When someone prompts the model with something it does not know how to respond to, the LLM will choose to make up some information with assumptions based on the patterns learned during training phase. These models tend to prefer to hallucinate rather than saying "I don't know" due to the way they are built. They don't possess a notion of uncertainty for declining to answer so in the majority of times, they continue to generate information even if it might be factually incorrect. Recent studies suggest that it's a complex problem related to the model's training process, dataset, and architectural design [21]. Currently this is an area of active research. Some proposals have been made to reduce this phenomenon such as providing the model a larger and diverse dataset during training, the use of "verifiable" and "fact-checkable" data so that the model relies on actual facts instead of assumptions and the insertion of RAG (which will be covered in the following section in detail as it will be a key component in the design of architecture of this dissertation system). Inside RAG approaches, recent studies point to an emergent solution that further improves reliability and shows promising results in hallucination reduction. Graph Retrieval-Augmented Generation (Graph-RAG) is the idea. It leverages structured knowledge representations to establish relationships between entities during the retrieval process providing preciser information with strong context grounding for generation [35].

Other factor that affects reliability is reasoning errors meaning that LLMs can make mistakes in logical reasoning and mathematical computation or inferences due to ambiguities from the prompt or by internal limitations intrinsic to the model [66].

3.1.4.3 Safety concerns

As LLMs gain more popularity, the potential for malicious misuse grows aswell posing risks for both the system and the users. Figure 3.4 demonstrates the landscape of security threats that LLMs have been inheriting throughout their rise.

Figure 3.4: Security Threats associated to LLMs (adapted from [34])

Prompt Manipulation and Data Poisoning constitute some of the higher risk security threats. Malicious actors might craft specific inputs to exploit and gain control or bypass restrictions of the model's behavior.

Another issue, not represented in the figure, is bias in training data. Due to learning from large documents, LLMs might inherit some biases regarding gender. race or other sensitive topics that later manifest in outputs. These can be amplified by malicious actors who attempt to target certain demographic groups [16].

Mitigating bias is also an active research area with works being done targeting LLMs stages of pre-processing, in-training, intra-processing and post-processing [39].

3.1.5 Pre-Training and Tuning Techniques

The development of a LLM is strongly influenced by a paradigm described as "pre-train-then-align". This means that models pass through a pre-training phase and later are submitted to a tuning phase to align the model with the tasks it's being built to solve [54].

Pre-training consists in a stage where the model is learning patterns and world-knowledge from large-scale, unlabeled text corpora. This process typically happens with an objective in place. Some of these objectives are: making the model to predict future tokens given the previous tokens, described as Next Token Prediction (NTP); tokens are masked randomly and the model is asked to predict masked tokens given the past and future context, Masked LM (MLM) example.

After the completion of this process, fine-tuning occurs. Fine-tuning can be described as the process to improve the model architecture by the enchantment of parameters and learning strategies with a goal of boosting model performance.

During the past years, several fine-tuning approaches have emerged such as Parameter-Efficient Fine-Tuning (PEFT) with Low-Rank Adaptation (LoRA) that freezes the majority of the model and trains only a small subset of parameters allowing for reduction of computational costs and memory requirements and showing similar performance to fine-tuned models. Reinforcement Learning from Human Feedback (RLHF), as the name suggests, utilizes human feedback in order to fine-tune the models by aligning them with human preferences. Instruction Tuning, a method used to generalize the model to unseen tasks. The model is given a collection of instruction-response pairs covering various capabilities (e.g., "Translate this text to language X" or "Answer this question"). This type of fine-tuning improves zero-shot generalization and downstream task performance [60].

Lastly, but not least important, Supervised Fine-Tuning is a technique, used as a step for RLHF, where the model receives labeled and task-specific datasets and its parameters are optimized for a given task. It is particularly valuable in healthcare settings because it allows to train the model with specific clinical examples [35].

3.2 Retrieval-Augmented Generation

3.2.1 Concept and Motivation

As previously referred during section 3.1, LLMs, whose ability to store large amounts of factual knowledge has been widely verified, still have some limitations regarding knowledge-intensive tasks and the production of "hallucinations" when handling queries that surpass their training knowledge capacities [43]. As discussed in Section 3.1.5, LLMs encode knowledge in their parameters during pre-training phase. This makes the model static, difficult to update and locked within the corpus quality. In a need to fight these constraints, RAG has been presented as the most regarded approach to follow[64].

In its core, RAG is a stellar innovation that combines text generation (from the LLMs) with Information Retrieval (IR). Rather than resorting only on parameter knowledge gained through pre-training, RAG systems, incorporate into generative models, external knowledge sources such as vector databases or knowledge graphs. This way, LLMs have a way of updating and grasp new knowledge in real-time ensuring that the generations are up-to-date and accurate [69].

Another advantage this approach allows is the ease with which knowledge can be updated. Instead of relying of retraining the model, RAG systems require only modifications to their external knowledge corpus, making them adaptable for any information need or imposed domain task. It presents itself as extremely valuable to avoid purely generative behavior and shift towards a evidence-based generation where the model provides relevant citations and experiments towards the generated output.

Projecting to healthcare, RAG systems facilitate traceability and provide explainability since they link answers to source documents allowing experts to verify, validate and assess AI generated recommendations. These properties are essential for mitigating hallucinations, boost trust, and supporting safe deployment of language model-based decision support systems in clinical environments [8].

3.2.2 Core Components of a RAG Pipeline and Workflow

Although there are several possible implementations of RAG, most systems follow a similar workflow that can be divided into four main components. These components work in a sequential way to transform the input into a factually correct answer. The following subsections describe these components and their respective interactions. Figure 3.5 demonstrates a simplified RAG pipeline highlighting the workflow between knowledge indexing, retrieval and answer generation.

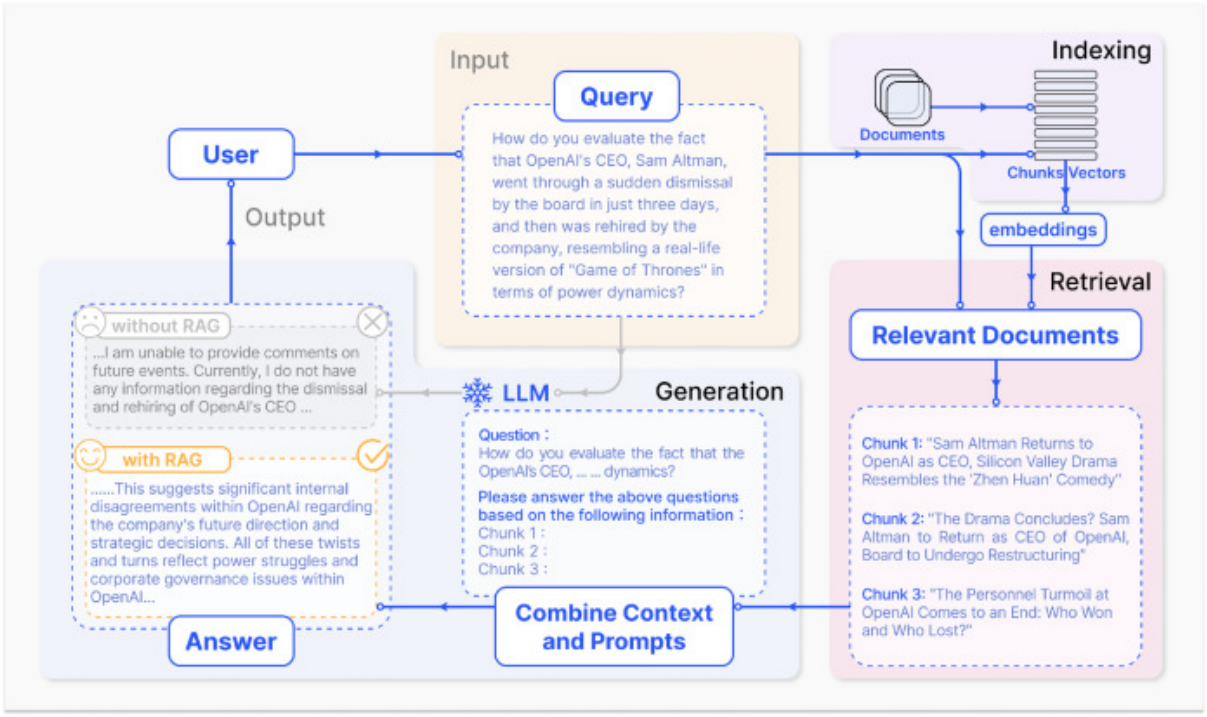


Figure 3.5: Overview of RAG pipeline (Source: [67])

3.2.2.1 Knowledge Indexing

Indexing refers to the cleaning and extraction of raw data in diverse formats. It can be considered as a processing stage that prepares external knowledge sources for the retrieval phase. Here, documents are ingested, segmented into smaller units called chunks. The segmentation part of this process is crucial as the quality of them determine the correct context that can be obtained.

Studies reveal that the most common methodology followed for segmentation is to split the documents into chunks with a fixed number of tokens depending on our needs and computational power. Larger chunk sizes give more context but also generate more noise leading to higher costs and processing time. Small chunks can miss out on context but don't have as much noise. Alternative methods as sentence-based segmentation or semantic chunking are also valid approaches that preserve topical coherence and are as relevant [62].

These chunks are then converted into dense vector representations (encoded) and stored in a vector database. Similarly to the embeddings created in the Transformer architecture (Section 3.1.2.2), they contain the semantic meaning of the text and enable nearest-neighbor search. Figure 3.6 shows some existing vector databases.

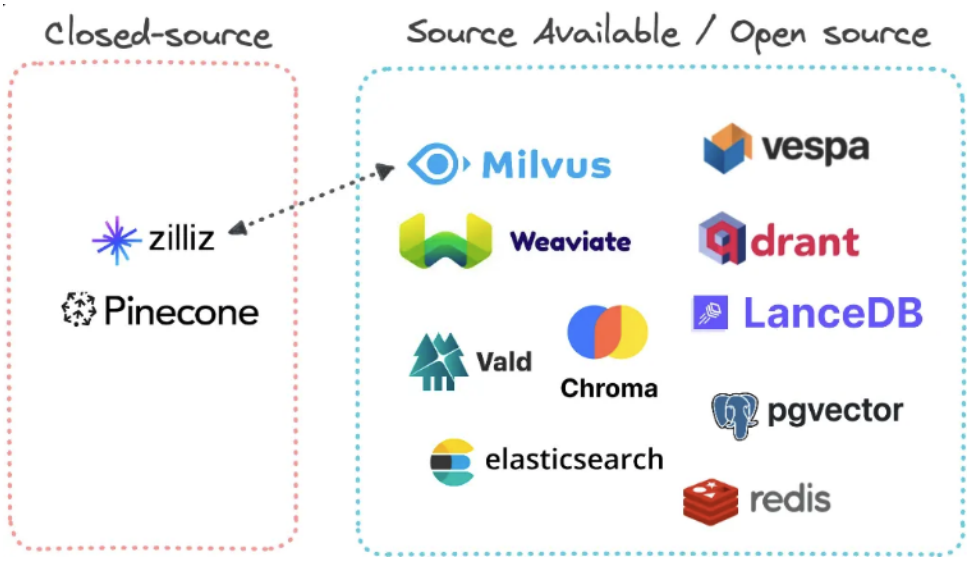


Figure 3.6: Overview of vector database architectures. Source: Rao (2023) [45]. Licensed under CC BY-NC-SA 4.0.

In order to perform searches efficiently in these databases, frameworks exists to promote efficiency. Most common are FAISS and HNSW (a clustering and similarity algorithm and a nearest neighbour search algorithm, respectively).

A complementary approach to vector databases is the use of knowledge graphs. Knowledge Graphs (KGs) consist of a structure that allows the representation of entities and the relationships associated to them. They are able to represent factual knowledge in a structured form and allows the model to understand clearly the semantic meaning of input queries done to it and have efficient relationship inference based on the built graph. A research done on this matter shows the possibility to leverage both vector databases and KGs to improve the quality and accuracy of the responses given [67].

3.2.2.2 Retrieval Process

The main goal of retrieving is identify which pieces of text (in the large collection of documents) are relevant for answering the user input. After receiving a user query, the system utilizes the same strategy as in indexing (using an encoding model) and proceeds to transform that query into a vector representation. After, it computes the similarity scores between the query vector and the chunks stored during indexing. With this process, the system retrieves the top-K documents that show greatest similarity to the query [43].

This process struggles with precision and recall. Precision issues happen when retrieved documents are semantically similar to the query but not quite factually relevant and recall problems occur due to the system missing information related to semantic differences. To mitigate these occurrences, frequently, sparse methods are used such as BM25 and TF-IDF, which analyze word statistics from texts and construct inverted indices for efficient searching [69] [8]. Another used method in some more complex systems is Re-ranking. This technique refers to a reordering step on the retrieved content to achieve better results (it is a computationally expensive approach although producing noticeable performance upgrade [50].

3.2.2.3 Context Augmentation

Augmentation or knowledge integration, is the process that connects retrieval to the generation phase [15]. It's goal is to build a query that combines the retrieved documents with the user query to feed the generation model so that we use both external evidence and the models internal parameters. An example of this is to add information such as "Use the context [retrieval documents]" and then prompting "Given the context answer [initial user query]. A thing to keep in mind in this step is also the trade-off in the amount of context that is given to the model. Too little may lead to information being lost due to the omission of critical parts but too much can make the model overwhelmed with the introduction of noise or irrelevant information [31] [53].

3.2.2.4 Generation

The generator is responsible for the production of the final answer given the augmented query from previous phase, meaning that, unlike LLM only using its internal training information, it is now using RAG-based generation incorporating evidence retrieved at inference time, enabling more contextually informed outputs [43]. An important detail to keep in mind is that it is not a good practice to input all the retrieved information into the LLM. Because of the limited context window LLMs have, redundant information interfere with the final information and often lead to a "Lost in the middle" effect where due to the enormous amount of information in the context, the LLM struggles to retain knowledge buried in the middle of the data [38]. These issues are resolved with better augmentation techniques such as the described re-ranking [50] and also through context selection/compression [69].

3.2.3 RAG Architectures

While most of RAG systems follow the pipeline described on the previous section, the way the components interact and act can differ depending on the objective defined or system requirements. As a result of this, RAG has many possible implementations that trade off performance, computational costs and controllability.

A recent survey distinguished some of these implementations by categorizing them from the simplest to the more complex form calling them naive, advanced and modular RAG systems, respectively [17]. Another trending subject regarding domain-specific implementations is the KG-enhanced RAG particularly for domains such as healthcare [24]. A rundown on these approaches is done in the following subsections.

3.2.3.1 Naive RAG

Naive RAG represents the most simple instantiation of a RAG system. It follows a very straightforward approach, similar to what its described in Section 3.2.2, where the user query is embedded and used to retrieve the top-K similar documents. The representations of these documents are then concatenated with the user query and passed to the language model which generates the answer. Figure 3.5 is a good representation of this implementation.

3.2.3.2 Advanced RAG

Built from Naive RAG, Advanced RAG proposes some improvements to address Naive RAG limitations in the indexing and retrieval stages. It is introduced here the concepts of pre-retrieval and post-retrieval. These processes occur, as the name indicates, before and after retrieval. Pre-retrieval consists in optimizing the user input with query rewriting, transformation and expansion techniques so that they become clearer for the retrieval task. Indexing might also be improved with the addition of metadata that later can reduce the scope of retrieval. Post-retrieval is the application of techniques already described in the above section such as Re-ranking, context compression and summarization techniques [17].

This is a more complex system with a higher computational cost that offers more robustness and trust for knowledge-intensive applications.

3.2.3.3 Modular RAG

This is an innovative approach that leaves the common "Retrieve-then-generate" workflow that both Naive and Advanced approaches follow. In Modular RAG, the pipeline is decomposed into independent and reusable components to enhance the entire RAG ecosystem. Instead of following the common executing flow, specialized components such searchers, retrievers, reasoning units, verifiers are added to the workflow. Each module appended is configured for a specific task and can be changed or removed without disrupting the remaining pipeline [17]. One of the major benefits of this architecture

is that it enables integration with other technologies in a simpler manner such as fine-tuning tools or structured knowledge sources. Modular RAG, however, has the highest computational requirements and costs, introduces latency due to all the associated components and requires careful coordination of the relationships between components to avoid unintended results.

3.2.3.4 Knowledge Graph Enhanced RAG

For domains where structured knowledge and many relationships between entities exist, the integration of knowledge graphs in RAG systems is becoming an important complementary tool to ensure that the retrieval phase generates results that match factual knowledge. As previously described in this dissertation, knowledge graphs have plenty of advantages compared to simple RAG systems as these can process entities relationships, keep the semantic meaning of input queries, easily update knowledge and due to the connections and information they keep, provide a clear path of knowledge.

A paper by Xu et al. [62] proposed some strategies for KG-RAG implementation. Here, four methods are explored regarding the way retrieval mechanism is instantiated. Vector-based KG is the first one, described as the simplest to use, but faces problems regarding context windows since only local information from the knowledge graph is used thus may not fully take advantage of relational structure. Then, Keyword-based KG are proposed as an upgrade from vector implementation since it grasps more context due to using keywords to find matches between entities and their connections, however, it requires manual configuration and fails to process unstructured text.

Hybrid KG method is also described as it combines the strengths of vector and keyword-based retrieval so it tackles both the unstructured text and structured graph reasoning. This is a more expensive alternative since it needs parameter tuning and has a higher computational complexity.

Finally, the authors proposed an approach that leverages the complementarity between KGs and vector indexing by integrating a KG and vector query engines. This Dual-Path approach, as it's called, use the vector path to capture textual semantic similarities and KG path to retrieve more relevant information from the structured semantic relationships. This requires more development work and might be an "overkill" approach for simple queries.

3.3 LLMs and RAG in Biomedical,Clinical,Pharmacology Contexts

3.3.1 Overview of AI Adoption in Healthcare

AI adoption in healthcare has been an emergent challenge with constant evolution throughout the years moving from rule-based systems (that are still used) to LLMs [42]. The biomedical field has been passing through a transition phase where the early applications were using semantic web approaches such as building ontologies to grasp domain-specific knowledge. As the increasing size of datasets in this area and the remarkable advancements that LLMs have had over the last years, they started to

being implemented to replace the methodology used and facilitate the work of practitioners [59]. Early implementations required lots of manual fine-tuning and engineering [2].

At the moment of writing this dissertation, AI systems have been integrated in some specific areas of healthcare particularly on diagnostic support, medical imaging, clinical decision support, patient monitoring. Along with these implementations, a rise of concerns has also been forming together. Experts, clinicians and even the providers of these applications highlight that there is the possibility of these systems to fail and that they should not be used alone, advising for human verifications and reliability and transparency issues [29]. Since this is an area where mistakes can be fatal for human beings, there is on going research on how should providers mitigate certain faults such as hallucinations, bias and even problems associated with context windows that may lead to misinformation.

These applications are to be decision support tools and not autonomous decision makers. They aid practitioners by acquiring and synthesizing knowledge from large text corpora that should be supported by evidence and have output to grounded information [49].

3.3.2 Existing systems and studies

Healthcare professionals already have at their disposal a variety of tools to support decision-making. Most traditional solutions include PubMed [1], UpToDate [2], DynaMed [3] and other similar platforms that function as structured databases and contain search mechanisms (keyword-based or semantic search) for clinicians to manually explore and retrieve the needed information.

As mentioned, recently, there has been a growing development of AI-driven tools for healthcare that aims to support professionals during their literature exploration processes, clinical reasoning or even treatment related questions. Examples of these tools include OpenEvidence [61] [4], ClinicalKey AI [5], Medwise AI [6] and Glass Health [7]. There have also been produced a substantial amount of research at approaches of integrating LLMs with RAG techniques to create biomedical tools. Most of these serve to introduce potential techniques to enhance factual accuracy, computational efficiency, retrieval setups and even validation layers to improve the output generated.

These papers collected during the Literature Review phase, showed plenty of implementations that allowed to gather an idea on how to approach the next section of this dissertation by showing what is possible and how to optimize the things. Starting with MedRAG [70], for example. MedRAG proposes a RAG framework combined with a hierarchical knowledge graph designed to differentiate between conditions with similar clinical manifestations. The system constructs a four-tier hierarchical diagnostic KG over existing Electronic Health Records, integrating diagnostic relations with vector similarity

[1] https://pubmed.ncbi.nlm.nih.gov/
[2] https://www.uptodate.com/
[3] https://www.dynamed.com/
[4] https://www.openevidence.com/
[5] https://ai.clinicalkey.com/
[6] https://medwise.ai/
[7] https://glass.health/

search. MedRAG achieves a diagnostic accuracy result of 79.25% on a diagnostic dataset, collected from Tan Tock Seng Hospital and 88.65% in a public DDXPlus (a large-scale synthesized EHR dataset, recognized for its complex and diverse medical cases) benchmark. Evaluation in this work was done using accuracy as the main metric and compared with standard RAG and LLM baselines. The system has been published and its code is publicly released but no production deployment has yet been reported.

KG-RAG [52], a similar study, shows the combination of large-scale biomedical KGs with pre-trained LLMs, leveraging the SPOKE (Scalable Precision Medicine Open Knowledge Engine) knowledge engine. It studies a pipeline implementation that starts with entity recognition of diseases, mapping concepts from input to nodes in the knowledge graph and passes through a "selection" stage where only the most relevant neighboring nodes are extracted so that the token usage is reduced, making the system more efficient, but preserving the context window. Its benchmarks, done in human-curated biomedical questions (multiple-choice and true/false) datasets, showed a performance improvement up to 71% (for *Llama-2-13b* model). It also reports a reduction of over 50% in token consumption without compromising accuracy. The authors note that the system is not ready for clinical use because the KG is missing formal clinical validation.

On the same note of efficiency and domain relevance, CLEAR (Clinical Entity Augmented Retrieval) [41] proposes a pipeline based on RAG that attempts to extract clinical information from Eletronic Health Records (EHR) notes. It drifts from chunk-based embedding retrieval and utilizes a Named-Entity Recognition (NER) model to identify clinical meaningful entities that are later filtered accordingly to match the input case and expanded with the help of medical ontologies for related concepts. Evaluated on 20,000 clinical notes and using six LLMs and targeting 18 extraction variables, CLEAR achives a mean F1 score of 0.90 compared to 0.86 for embedding based RAG and 0.79 for full note approaches. What really stands out is the inference time of 4.95 seconds per note against the 17.41 and 20.08 seconds for the other methods being compared. This represents a reduction of over 70% in both token consumption and inference time, making this work particularly good to understand the use of NER against embedding retrieval on a large clinical corpus.

To fight hallucinations and bias issues, SelfRewardRAG [23] introduces a multi agent RAG framework designed for medical reasoning. The system combines *GPT-3.5* model with live retrieval from Pubmed and structures its pipeline into four stages (those being, query formulation, answer generation with the LLM agents, self-evaluation layer that identifies inconsistencies and medical inaccuracies and a final agent tasked with selecting and refining the final output. This system was benchmarked on three medical question answering datasets, PubMedQA, MedQA-USMLE and BioASQ, having achieved scores of 81.1%, 50% and 95%, respectively. This performance surpassed the base *GPT-4* model across all tasks. The system however has not been validated in real clinical workflows and authors note for the computational demands that can be a challenge for its deployment.

To mitigate another issue regarding LLMs, "Lost-in-the-middle" effect, RAG-Chain [13] addresses this limitation through the introduction of prompt and reasoning optimization. It integrates a multi-stage pipeline that uses sentence-level chunking and re-ranking mechanisms, uses chain-of-thought techniques

and validation steps through multiple possible outputs. It was also evaluated on medical QA datasets such as, MedQA and MedMCQA. In them, along with the *Qwen2-7B-Instruct* model achieved 60.6% and 63.7% accuracy respectively. These results surpass domain specific pre-trained models without requiring any doamin-specific fine-tuning or pre-training. Still no real use of this system is documented.

Finally, an idea focusing on producing output and reasoning from up-to-date medical repositories, Almanac [65]. The idea behind it, is the connection of LLMs to curated medical repositories, more specifically, to Pubmed and UpToDate, at inference time, using semantic search and embedding based retrieval to generate recommendations with both verifiable citations and clinical guidelines. The work was evaluated with a dataset (ClinicalQA) featuring 130 clinical scenarios by a panel of five certified physicians. Almanac shows that its pipeline demonstrates an 18% improvement in factuality when compared to normal LLMs with also improvements in the completeness and safety side. Despite this, the system is not yet publicly available and has not been reported as entering clinical production.

Across these works, it is identifiable a set of limitations. Most of the systems are evaluated on controlled benchmarks or curated datasets rather than real clinical workflows. None of them has been validated and deployed in a production hospital environment and evaluation protocols remain heterogeneous meaning that a study comparison of the results is difficult.

Chapter 4

Methodology

This chapter gives an overview on the system architecture, experimental design and evaluation methodology adopted in this dissertation. Based on the theoretical foundations established by the state-of-the-art chapter, the objective of this chapter is to describe how the project's solution is designed, evaluated and how its use can enhance the process of automating the draft of technical pharmacology reports.

4.1 Research Design and Context

This project follows an experimental design conducted in collaboration with UFC at HSJ focusing on automating technical pharmacology reports generation in response to requests from CFT. These reports are produced whenever the CFT requires evaluation of a medication, therapeutic strategy or a clinical question for which a review of the available scientific evidence is required. Once they are finalized, they serve as technical decision-support documents, synthesizing the available evidence and providing recommendations that assist the committee in making informed decisions.

This research will follow an artifact-oriented approach where the primary goal is the design of a RAG system and combining it with existent pharmacologist produced reports, systematic error analysis and human-in-the-loop feedback.

UFC follows a pattern defined workflow regarding the creation and analysis of technical reports going through six main stages [46] (Background and scope, Clinical Case Details (when available), Scientific Foundation (based on existing medical studies), Synthesis and Considerations, Evidence-based recommendation and finally the necessary citations). The CFT requests also vary according to six different types: Medication efficacy/safety, scientific evidence for clinical contexts, therapeutic options for diseases, efficacy/safety versus standard of care, monitoring criteria and initiation/suspension of a specific drug criteria.

To summarize, the system must be able to generate reports that follow the described workflow by UFC retrieving information from trustworthy and regulated medical sources and adapt to the request

category while providing, in the output, verifiable citations that ground the generation process. Final validation and approval still remains with pharmacologists or medical experts.

4.2 Business Understanding

Currently, the workflow carried out by UFC is as follows: the CFT submits a request requiring the evaluation of a given drug(s), therapeutic strategy or a specific clinical question. These requests arrive at the UFC where pharmacologists are to perform a detailed review of scientific literature, documentation and other relevant sources of information. Once this is done, a technical report is produced following a template structure. This process currently takes between two weeks to one month for a report. Later the report is sent to the CFT to support a decision on the given subject.

Based on this workflow, the goal of the business understanding phase is to assess and define the challenges that can be addressed through the proposed system. Meetings with the stakeholders allowed the identification of the HSJ organization current challenges, needs and expectations to be met with the development of this clinical support tool.

In these meetings, HSJ organization alerted for current issues regarding time management and the need to reduce their workload in what is a demanding and resource-intensive process. The adoption of AI-based systems was recognized as a promising solution to alleviate these constraints while keeping the necessary scientific rigor, reliability and accuracy required in healthcare contexts. Stakeholders also expressed the need of a natural language interface, to be able to communicate with the tool, as well as highlighting the need of proper context grounding throughout the documents retrieved in generation.

The business objectives guiding this study are as follows:

1. Implement an AI system capable of generating clinical reports (in Portuguese: "Pareceres") in a fraction of the current 2 to 4 week timeline
2. Ensure traceability and transparency of generated outputs through explicit source attribution;
3. Enable reliable retrieval and synthesis of relevant scientific and regulatory information;
4. Preserve expert supervision as the main element of the final decision-making process.

4.3 System Architecture

The proposed system is designed to automate the real-world workflow of UFC at HSJ, where technical reports are produced by clinical pharmacology professionals given a formal solicitation by CFT, as mentioned on Section 1.3. The architecture reflects the constraints of a hospital environment ensuring that there is the integration of information sources used by UFC as well as relevant data provided from HSJ and keeping human-in-the-loop for the required expert validation.

The system architecture follows a layered, component-based design comprising external API integrations, a core RAG pipeline (retrieval, context augmentation, report generation, post-processing), and dual storage backends (vector database and knowledge graph). Figure 4.1 provides a high-level block diagram of these components and their interactions.

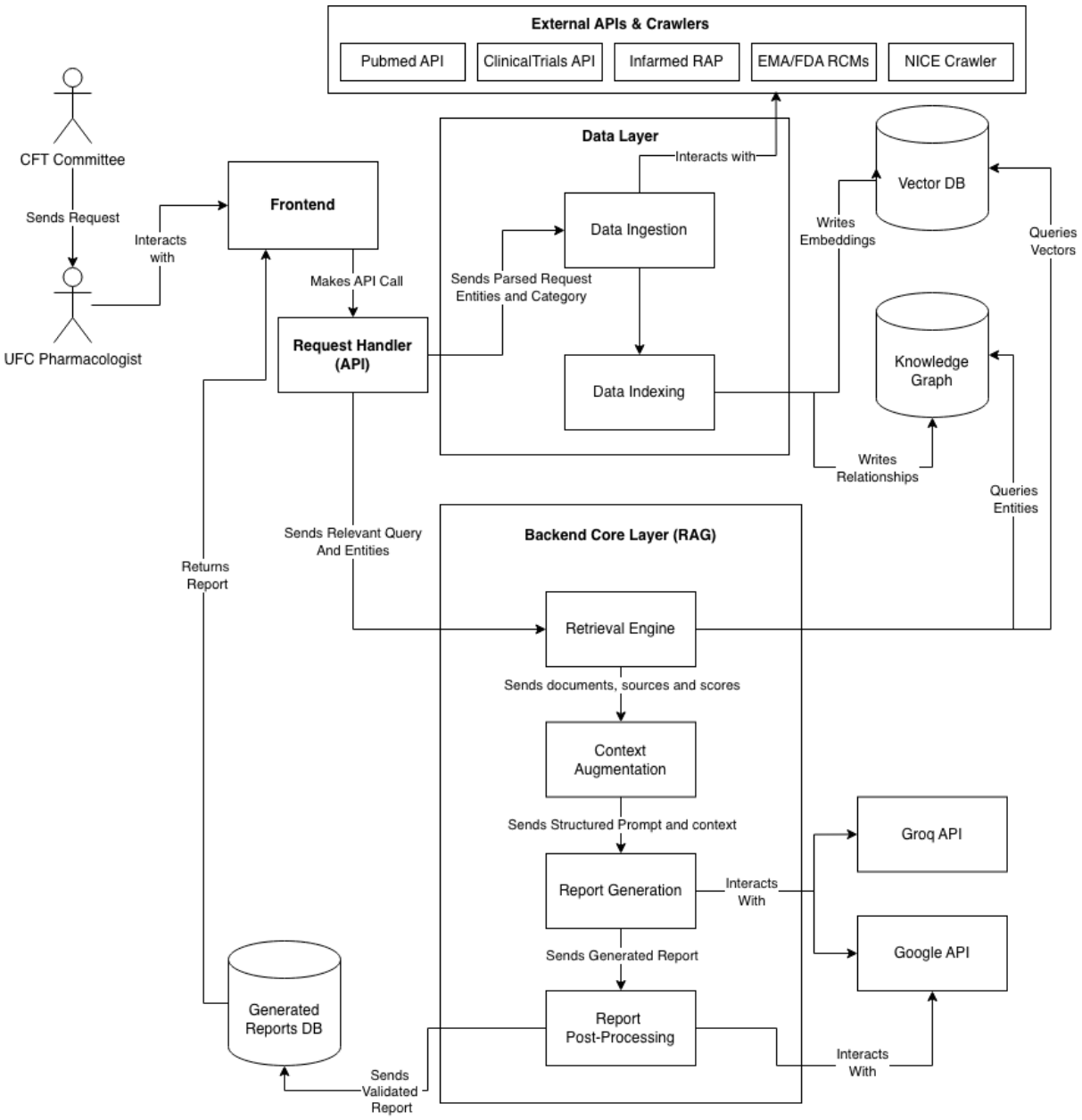


Figure 4.1: Proposed System Architecture for UFC use cases

4.3.1 Request Handling

The Request Handling layer acts as the primary entry point of the system serving as the bridge between the Frontend interface and Backend processing components. This component is responsible for receiving the necessary input files coming from the user (such as the CFT request) containing unstructured information regarding the drugs to be evaluated, the medical condition to be under study, details about the patient's clinical case (optionally) and the categorization of the type of report to be produced.

To manage the input of unstructured data, this layer utilizes Regular Expressions (Regex) and NER. Regex is applied for data cleaning purposes and also for structural parsing to ensure that only the needed segments of the request are parsed removing the unnecessary text parts. NER is implemented using *medialbertina_pt-pt_900m_NER* to correctly retrieve only entities that are classified in the biomedical domain according to what is being extracted (either drugs or conditions). Once the text is cleaned both Regex rules and NER models work together in the extraction of the relevant clinical entities.

This extraction is crucial to the correct functioning of the pipeline as it defines the drugs and conditions to be evaluated. The collected information also serves subsequently for the formulation of search queries for the retrieval phase as well as the ingestion phase in both the API calls and the crawlers queries formulation and finally are also injected into the report templates for LLM instructions and vector searching. This parametrized method ensures that generation is grounded in the specific clinical context of each request, rather than relying on generic prompting, keeping the patient context relevant.

Finally, the categorization of the request is manually defined by the user. The user can choose from three types of requests that each connect to a different report template that is selected in the generation phase. The request handler component makes sure to redirect this information along the necessary extracted parameters. This manual decision-making approach was adopted due to limitations encountered during automated classification from the combination of Regex and NER that showed inconsistencies and struggled determine the correct category of a given request consistently.

4.3.2 Data Layer

The Data Layer (representing both the ingestion and indexing phases in the architecture figure) acts as the core knowledge retrieval for the RAG pipeline. It's responsible for extracting, transforming and loading the necessary external medical knowledge to ground the LLMs generations. This layer agglomerates the processes of acquisition, standardization, processing and vectorization of data meaning that it is designed to deal with the heterogeneous set of interfaces from different scientific sources transforming their unstructured literature into machine-readable embeddings.

This engine was designed in a way that avoids ingesting huge clinical databases with all sorts of information. Instead, the data ingestion process is triggered based on the entities retrieved from the request handling via NER and Regex (such as drug names, clinical history and target medical conditions) ensuring that for each CFT request there are precise corresponding search queries being done. Once the module receives these entities, several distinct fetching systems begin working to retrieve relevant data

for the specific case in hands. The reason for the existence of multiple fetching systems was the need to satisfy all the technical pharmacology reports sections each requesting information from different data sources and to work around some formatting constraints from each data endpoint. The data sources are as followed:

- **Public funding evaluation reports (Infarmed) and regulatory documents (European Medicines Agency (EMA)):** The system connects to an EMA database in an excel file to obtain the Summary of Product Characteristics (RCM) through string matching the drug name with its International Nonproprietary Names (INN) and locate the updated drug URL, and parses a user input funding report from Infarmed referring to the drugs being evaluated. Initially, the funding report coming from Infarmed was fetched with a crawler that mimicked a human search for the drug(s) and extracted the file to the environment. Unfortunately, due to Infarmed website protection against crawlers, the use of this idea was discarded since it led to the required webpage being taken down. Infarmed has an API but its use is not free so the use of it was discarded. Due to this, the solution was to let the user introduce the document (optionally) to the LLM.

  These documents provide information regarding the drug(s) therapeutic indications, safety profiles, posology and pharmaceutical guidelines. It serves purposes for both the pharmacology and pharmacoeconomics evaluation sections draft.

- **Scientific literature (PubMed and ClinicalTrials.gov):** To have access to the latest and updated peer-reviewed evidence and trial results, fetchers were built to connect directly with the PubMed and ClinicalTrials APIs. These fetchers retrieve mainly abstracts, study methodologies and conclusions regarding the safety of the drug(s) in question. It's required to build the Main Molecules Studies section.

- **Clinical Guidelines (National Institute for Health and Care Excellence (NICE)):** Information required to fill the pharmacoeconomic and related studies sections come from NICE. It possesses internationally recognized clinical guidelines and care recommendations. It also provides a framework to compare proposed treatments against established protocols. NICE does not allow for the use of its API for countries outside of the UK, for that reason, a crawler was built. This crawler takes the main entities from the request, formulates a query and fetches the top-K documents.

- **Terminological Normalization (FDA/NIH):** To ensure consistency across the different data sources, the system uses APIs to map raw extracted drug names (which may include brand names, abbreviations, or spelling variations) into standardized medical concepts, specifically their International Nonproprietary Names (INN) or active substances. This step is essential for interacting with the Knowledge Graph. The nodes within the graph do not represent commercial drugs; rather, they represent these standardized active substances. By normalizing the terminology before querying or building the graph, the system removes ambiguity and ensures that the entities extracted

from the CFT request map exactly to the correct graph nodes, enabling accurate retrieval of drug interactions and contraindications.

One of the main challenges in building this layer is extracting crucial information from unstructured data, particularly from PDF documents. To address this, document conversion utilities were employed, such as Docling [1] [40] for more complex PDFs (EMA and Infarmed reports), due to its ability to preserve document structure, accurately extract textual content, parse embedded images and tables, and convert documents into well-structured Markdown format. Additionally, PyMuPDF [2], a Python library for fast PDF parsing, is used for processing PubMed and NICE studies due to their simpler text formatting.

After parsing RCM and Infarmed reports, the document layout is converted into JSON format. This structured representation enabled the use of Regex pattern matching to automatically identify and segment relevant sections of the document. Numerical patterns present in section headers are used to detect boundaries and slice the text accordingly. By restricting the document to the most relevant sections, this step reduces the search space and helps mitigate potential hallucination risks with the overloading of the context window.

Keeping the context window in mind, it was essential to have a chunking step where raw text, after being parsed and cleaned, is divided into smaller segments. The chunk size was defined based on some studies [19] recommending an approach that use both their size and an overlap (sliding window of token extension). This overlap is essential in preserving text relationships by ensuring that a sentence that is split by the chunk does not lose its semantic meaning.

Finally, the processed text is prepared for the retrieval phase. To do this, a medical specialized embedding model was utilized (more specifically, *MedEmbed-large-v0.1* [5], due to it's performance on biomedical text retrieval tasks). The reason to go with a specialized model instead of a general purpose one is justified by the need to have a model that could capture complex clinical semantics within the created chunks, a need that, general embedding models are more predisposed to fail in recognizing. Once embedded, data is routed to a vector database hosted in PostgreSQL (supported by the pgvector extension to allow vector search mechanisms).

4.3.3 Retrieval and Context Augmentation Pipeline

After the processing and storing of all the needed medical literature and structured pharmacology data regarding the requested drugs, the system must now retrieve the most relevant information to answer the CFT request. To execute this, the retrieval and augmentation pipeline ensures the connection between the stored data present in both the vector database and the knowledge graph, and the generation of text with a LLM following a Hybrid-RAG strategy.

Following the entities left by the request handler, such as, the primary drug, the drugs to be compared (if its the case) and the clinical history, the retrieval process begins by translating them into english

[1] https://www.docling.ai/
[2] https://pymupdf.readthedocs.io/en/latest/

language. This is essential before proceeding as it ensures that the mismatch errors or even conflicting information with the databases is avoided as these entities come in portuguese text and wouldn't be usable as the embeddings are in english. Also, because the final report is divided into specific sections, querying the entire database for every section will lead to the introduction of noisy data and botch the LLM's context. To avoid this, the system was created with the option to choose which specific database tables are to be used depending on the section to be generated. For instance, if the "Pharmacology" section is being generated, there is no need to query the PubMed or ClinicalTrials tables as the information mostly needed comes from only the EMA table and the KG.

To gather the complete knowledge necessary for generation, the pipeline merges all pharmacological targets from the request and loops through them. For each drug, a specific text query is put together by concatenating the translated drug name and the clinical condition (for some data sources only the drug name is used though - this is the case of NICE and ClinicalTrials.gov. This happens because these sources have mostly guidelines and general experiments not associating the condition in the title to which is best to gather more documents and then let the vector search do its work).

In order to perform the vector search, with the help of the pgvector extension in PostgreSQL, the search queries are also converted into vector embeddings using the same embedding model used in the ingestion phase. Then, the system performs a cosine similarity search (K-Nearest neighbors), between both query and vector database. Because standard cosine similarity can sometimes miss deep clinical semantic nuances, a second phase is introduced using a specialized medical Cross-Encoder (*MedCPT-Cross-Encoder*. This reranker evaluates the original query and each candidate chunk simultaneously, assigning a highly accurate relevance score. The result is a top-K highest scored chunks that contain the necessary context to feed into the LLM to answer the original query for the section. Concurrently, the system queries the Knowledge Graph by matching the drug entities that come from the request with the drug nodes present on the graph. Once these are identified, all the information present in that node is sent to the LLM. The use of the KG allows for consistent retrieval of pharmacological regulations, bypassing the probabilistic nature of vector similarity that could introduce a point of failure in context gathering.

Once the data is retrieved, the information now must be redirected to the generation layer. Before this, it needs to be formatted into a prompt that the LLM can comprehend. This is where the Augmentation component acts. It takes the outputs from both the vector and graph databases and structures it into a payload string. To not lose the ability to ground the information, as the chunks and graph relationships are appended to the context window, their origin variables are also injected into the text so that they can be later formatted into the bibliography section.

4.3.4 Report Generation and Post-Processing

Moving into the final part of the pipeline, since now the clinical context is fully structured, the system proceeds to draft the requested pharmacology report. Considering that the final document requires distinct ways of phrasing and thought depending on the section being written, generating the entire report

in a single pass is tendentious to make the LLM forget the instructions that are given to it and exhaust the context window. Another problem that emerges from attempting to generate in a single pass is the high token consumption. As only free LLM APIs are used, token limits and request numbers must be respected in order to keep using the system. To mitigate these issues, the generation process operates in a iterative way, generating the document section-by-section being guided by a prebuilt structural template.

These templates appear as configuration files (written in YAML format). In them, the report is structured section by section with specific prompt instructions, optimal token length to output, the amount of chunks necessary to fetch and from which data sources should the system retrieve from. The system iterates through this file, extracting the necessary context from the retrieval pipeline and appending it to the prompt schema.

Although everything is concatenated, the created prompt structure explicitly isolates the clinical entities, retrieval results and the generation task. By separating the distinct sections of the report, the LLM is allowed to be focused on a much smaller task at each time with the goal of minimizing any type of hallucinations and to enhance retrieval results since more specific queries are being made for each section.

To handle the generation, some adjustments are also made to ensure the correct functioning of the system. Firstly, this architecture chooses a dual-language strategy where the initial generation of each section is performed entirely in English. The reasoning behind this method is the fact that open source LLMs work significantly better when working with English based instructions [1] and also since the retrieved context is fully in English, the LLM reasons better with English text instead of Portuguese. So what ends up happening is that the sections are initially generated in English resorting to the use of APIs such as Google (particurarly the Gemma family - *gemma-4-26b-a4b-it* and *gemma-4-31b-it*) and once the section is generated, a secondary translation step takes place using a different API (can be the same but a different one (Groq API with the *llama-3.1-8b-instant* model) was used to comply with the free tier restrictions of the Google API) that converts the text into Portuguese.

Furthermore, because of the relying on external LLM APIs that can give sudden timeouts or server errors during long tasks such as this one, a retry mechanism is present in the system. It consists in a model fallback where, after attempting to generate using the primary model, if the system encounters some sort of failure, it identifies it and attempts generation with a secondary model. This ensures that the report does not end up with missing sections and the workflow is not disrupted because a one section failure implies the restart of the whole generation process. This is the biggest point of failure of the system and might require some user attention to be aware of possible errors in this stage. The solution to avoid this happening in a more permanent way is to pass from the free API tiers to a paid one to ensure that no errors or limitations are met.

After completing the iteration through all the sections, the database stores both the English and Portuguese reports. The choice to store both is inherent to the possible translation mistakes that can happen and make the Portuguese report have confusing sentences. One can verify the original context of those sentences in the English version to remove any ambiguities or doubts.

Given that the report was created section by section, although each one may be factually accurate, it misses formatting in a stylistic and professional way. To transform this draft into a more formal report, the entire text is passed to the final part of the pipeline, the Post-Processing step.

This serves mostly as a correctness mechanism to ensure that the core business requirements are fulfilled regarding transparency and source citations attribution. This process is also carried out by an external LLM API that is configured with system instructions and low temperature settings (temperature is a LLM hyperparameter that controls the randomness of text that is generated by the LLM [25]) to prevent drifting from only a grammar and text cohesion check to a more creative task.

The system instructions are as followed:

- **Language Standardization:** The LLM must enforce strict clinical European Portuguese grammar and terminology ensuring that it matches expert medical writing while also verifies and removes the existence of potential AI artifacts (for instance, "If you want i could..." or "Here is the requested...") that lead to a downgrade on the report's professionalism.
- **Citation Mapping and Formatting:** In the generation phase, the LLM writes raw metadata tags in the text to ensure traceability. During the post-processing phase, the LLM must rewrite these tags by formatting them using sequential numeric format and mapping them along the document.
- **Bibliography Construction:** The LLM fetches the real-world citation data for the corresponding paper IDs and documents, creating a new section called Bibliography at the end of the report containing all citations used for the report.

Once finalized, the report is stored in a database that is connected to a web interface from where the user can interact with the generated report (allowing for edits and exporting into PDF to deliver).

4.3.5 Knowledge Representation and Databases

As described in Section 3, standard RAG systems rely exclusive on semantic vector similarity for the retrieval of relevant information. This comes with certain limitations regarding the loss of information via the chunking process or even bring documents that although have high similarity also come with low accuracy, meaning that they share the same terminologies but do not quite answer to the needed requests. In a way to mitigate the issues that come from standard RAG, as stated in 4.3.3, a hybrid approach was chosen to also introduce a Knowledge Graph to the system to handle more complex relationships such as drug to drug interactions, pharmacological characteristics, or more rigid parameters like contraindications and warnings with the goal of preventing misinformation.

Since that there are not available drug graphs that are free to use, the Knowledge Graph introduced to this system was created from scratch and hosted on Neo4j. It is based on the ingestion of information coming from regulatory APIs (such as FDA and NIH) and also from the scraping of EMA summary of product characteristics documents for the respective drugs.

The idea is to create a solid foundation of the drugs pharmacological details while also introducing how they react with other drugs and the conditions that they treat so that structured pharmacological facts are stored. These facts serve as a complement to the probabilistic nature of vector retrieval.

The graph is modeled as:

- **Nodes:**

  Drug entities are created as nodes. These drug nodes contain information regarding their INN, active substance behind it, drug indications, contraindications, warnings and interactions. All these parameters are supported with the source from where they were taken to ensure that the data is grounded. All the drugs present on the current EMA dataset are present on the KG with the exclusion of drugs whose domain is not Humans (meaning drugs that are targetted to the veterinary domain) and also those whose status is "Not Allowed" (drugs that are no longer allowed to be used).

- **Edges(Relationships):**

  Drug nodes are linked to the condition they treat via a "Indicated_For" relationship and connected to contraindicated other contraindicated drugs with "Interacts_With" relationship.

The justification of this modulation comes from the need to fulfill the expected Pharmacology section to be created in the final report. This graph is capable and possesses the sufficient information to handle the creation of that section and tackles the issue of the inconsistencies that are expected to happen on the reliance of only vector semantics.

On the other hand, also important, the vector database used for unstructured data hosted in PostgreSQL with the pgvector extension. All the medical literature that comes from the defined data sources including, PubMed studies, clinical trials informations and the chunked data coming from the EMA, NICE and Infarmed reports are stored in this database (each with a specific table referencing the source from where they come).

PostgreSQL was chosen due to its proven reliability as a production-grade database system and its suitability for the scale of this project. The pgvector extension provides efficient approximate nearest-neighbour search capabilities that are sufficient for the number of documents handled by this pipeline, eliminating the need for a dedicated vector database instance reducing the complexity of the storage infrastructure while keeping its querying capabilities.

4.3.6 Human-in-the-Loop Validation

It is not feasible to create a tool that supports itself with AI in this field without keeping a human in touch with it. The use of AI brings risks due to hallucinations and the accountability for possible errors. With this in mind, the architecture is designed to only serve as a supporting tool rather than an autonomous agent. It gives an option to reduce the time and ease the work of the clinical pharmacologists when drafting these reports ensuring that the responsibility remains entirely with them.

The system keeps the human active in the pipeline requiring its assistance defining the categorization of the final report and injecting the CFT request and the Infarmed reports for the given drug(s) in the request. As outlined, there is also constant citation mapping and the creation of a formal bibliography within the generation phase so the pharmacologist can easily verify the veracity of the sources retrieved and validate the generated claims. Additionally, an interface (shown in Figure 4.2) was created to allow the expert to manually review, edit, expand or delete any generated content before officially exporting the document.

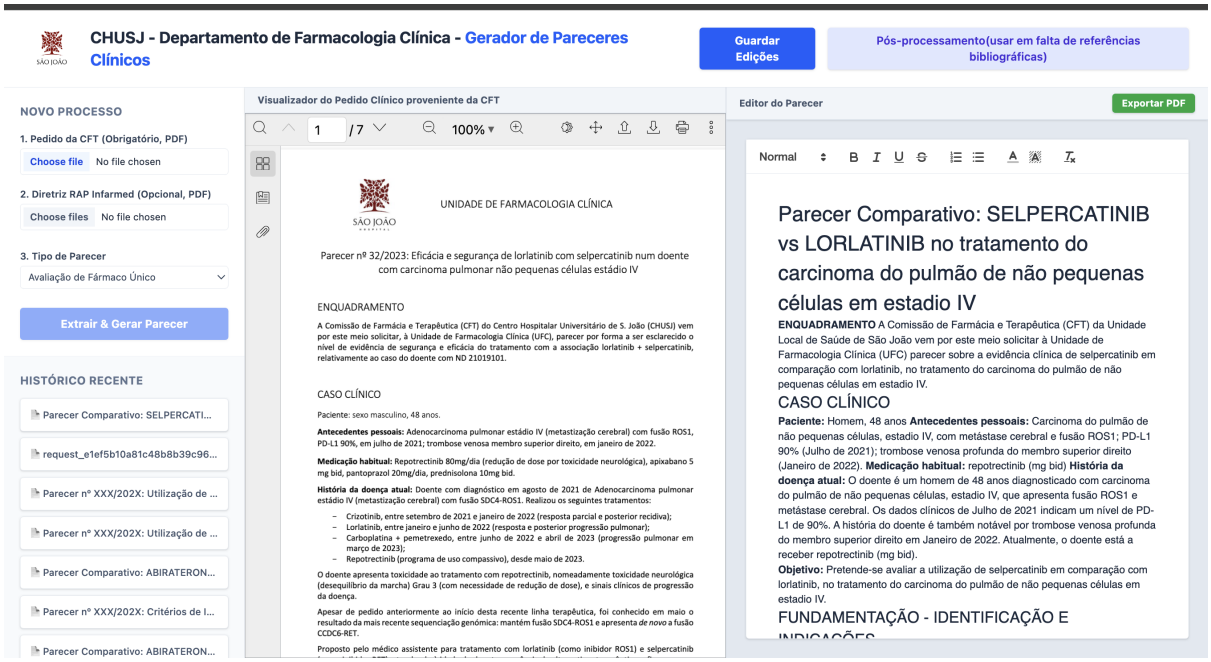


Figure 4.2: Web application interface for pharmacologist review and editing of generated reports

4.4 Evaluation Methodology

The evaluation methodology for the prototype built during this work is designed to assess the performance of the proposed RAG architecture by the comparison of the generated reports with the human-written reports previously produced by UFC pharmacologists (serving as the ground-truth dataset).

In order to judge the efficiency and effectiveness of this tool, the evaluation framework is divided into a mix of both human and machine evaluation meaning that both automated quantitative metrics and qualitative expert opinions are gathered.

Starting with the automated evaluation, textual and semantic similarity assessments are done to verify how close does the generated report comes to the original in terms of structure, terminology and content. To achieve this, metrics such as ROUGE and BERTScore are used. ROUGE, as mentioned on Section 3.1.3.1, measures the n-gram overlap, meaning that it evaluates the overlap of words and phrases

between the generated and original reports, analyzing if the generated report captures the vocabulary and formatting existing in the medical written reports.

Additionally, BERTScore is used to go further from only comparing the existence of words from which consists ROUGE. Clinical/Medical writing can express same contents using different phrasings. This means that ROUGE might produce lower results given that the same idea can be expressed through different words or even entire document structures. BERTScore evaluates semantic similarity by comparing contextual embeddings generated by transformer-based language models, allowing it to capture similarities in meaning and clinical context even when different words or sentence structures are used [68]. This way, it presents itself as a good complementary metric to ROUGE providing more robustness in the assessment of the final report quality.

Further than using the traditional metrics and evaluating only the final output, the pipeline itself needs to be tested. To execute this, a RAG specific evaluation framework, Ragas [14], is utilized to carry out the assessment of the pipeline's internal processes. Ragas consists of an open-source framework used to evaluate RAG pipelines by assessing the relationship between the question, retrieved contexts and the generated answer. Several Ragas metrics were utilized such as:

- **Faithfulness**

  Measures the consistency between the generated response and the retrieved information that allowed to generate that piece of content. The response is faithful if it can be supported by the retrieved context. Goes from 0 to 1.

- **Context Precision**

  A metric that measures the ability of the retriever to rank relevant chunks higher than irrelevant ones for a given query in the retrieved context.

- **Context Recall**

  Measures the number of relevant chunks that were retrieved. A higher value of context recall means fewer relevant chunks were left out.

- **Answer Relevance**

  Rates the generated answer given the prompt that was introduced. A lower score implies that the generated content is either incomplete or based on redundant information.

While these automated metrics provide objective benchmarks, they are not enough to conclude or guarantee the correct functioning of the tool. Therefore, human evaluation is introduced with a qualitative assessment in collaboration with the pharmacologists from UFC.

These pharmacologists are handed with their original written reports and the generated reports side by side in the same .pdf file to allow an easier comparison between them. An example from each of the three existing categories is provided. They are also supplied with a formulary that must be completed

and from where conclusions regarding the benefits of using this tool are derived and if the proposed goals are accomplished.

In this formulary, UFC experts, are tasked with giving their ratings regarding the generated content in terms of Clinical precision (Whether or not the pharmacology, posologic and results from studies fetched are aligned with correct guidelines and scientifically correct), Completeness (If the generated report includes every necessary information to allow a substantiated decision when presented to a commission like CFT), Hallucination detection, Structure and Presentation and finally an opinion is asked to evaluate the full report with questions such as what limitations are found and where could the report improve.

Chapter 5

Experiments and Results

This chapter details the experiments conducted to evaluate the built system and presents the results obtained. The goal is to assess the performance, accuracy and the possible clinical adoption of the developed pipeline. The chapter concludes with a discussion regarding the research questions guiding this dissertation.

The evaluation is structured across three phases, each targeting a distinct aspect of system quality:

1. **Ragas Framework Pipeline Evaluation (Section 5.2):** The internal behavior of the RAG pipeline is assessed using the Ragas framework, covering Faithfulness, Context Precision, Context Recall and Answer Relevancy. This phase operates at section level and requires a structured golden dataset, composed of 6 manually selected reports aligned with the generation template. BERTScore is also applied at this stage, taking advantage of the section-level evaluation to avoid its 512-token truncation constraint.

2. **Document-Level NLP Evaluation (Section 5.3):** The quality of the generated full documents is assessed against all 23 human-written UFC reports using ROUGE-L, Med-F1 and Semantic Similarity. Unlike BERTScore, Semantic Similarity is computed using large-context embedding models, making it suitable for full-document comparison.

3. **Human Evaluation (Section 5.4):** Two clinical pharmacologists from UFC assess the generated reports across four dimensions, with those being, clinical precision, completeness, hallucination detection and document structure using a Likert-scale questionnaire, complemented by open-ended qualitative feedback.

In all automated phases, generated reports are compared against the corresponding human-written reports produced by UFC pharmacologists, which serve as the ground truth. Section 5.1 begins by describing the experimental configurations and corpora used across these phases.

5.1 Experimental Setup - Automated Metrics

Before proceeding to the utilization of the available framework (Ragas) and the semantic metrics, it is vital to construct the ground truth for comparison. The ground truth employed in the evaluation consists of a sample of UFC written reports that were drafted and approved by the CFT.

Disclaimer: Each given report comes anonymized, meaning that no patient data is accessible for both the evaluation and the pipeline normal functioning. This is important to make sure that the system is in compliance with GDPR (General Data Protection Regulation). An example report can be seen in A.1.

A total of 23 reports are leveraged to assess this pipeline. Each one contributes to the validation of one of the three available categories, those being:

1. **Single Drug Evaluation**

   Request to verify the efficacy and safety profile of a specific medication for a specific condition

2. **Comparative Evaluation**

   Request to compare the efficacy and safety of two or more drugs for a specific condition

3. **Initiation, Continuation or Suspension Criteria**

   Request to gather patient-specific guidelines for starting or stopping the use of a given drug for a specific condition.

The distribution of the reports is as follows: 13 Single Drug reports, 8 Comparative Evaluation reports and 2 Initiation, Continuation or Suspension Criteria reports. This distribution reflects the frequency of reports that arrive to UFC, where both single and comparative evaluation reports account for the majority of requests that come from CFT

Although the goal is to validate the generated reports, it's important to keep the categories in mind due to the way they are built. As previously mentioned in Section 4.3.4, each one of these categories is associated to a template from where the generation process begins. It is vital to see how each one performs to see which ones need more work in terms of prompt engineering or even entire section rebuilding.

With the ground truth defined, an experimental study is designed to test robustness and validate the architecture under different settings. The experiments varied along three primary variables:

1. **Large Language Models:**

   Three model families were evaluated, Mistral [26] [1], Qwen [63] [2] and LLaMA [57] [3]. These models were selected because they are open-source, available for research use and can be hosted

[1] https://mistral.ai/
[2] https://qwen.ai/
[3] https://www.llama.com/

locally via the Ollama platform, eliminating the dependency on API providers and the restrictions that come with them. Initially, only LLaMA was used through the Groq API, however, due to the token limitations that come with the use of the free API, hosting the LLM locally was needed to ensure that evaluation ran smoothly. In a collaboration with INESC TEC (Instituto de Engenharia de Sistemas e Computadores, Tecnologia e Ciência), access to a GPU machine, featuring a *NVIDIA L40S gpu*, was gained, which allowed to host more computationally expensive models as Mistral and Qwen and perform the needed evaluations.

2. **Chunk Size:**

   Three chunk sizes were tested, being those, 800, 1000 and 1500 characters. As discussed in Section 3.2.2.1, larger chunks preserve more context but introduce more noise into the context window, while smaller chunks are more precise but can lead to splitting clinically relevant sentences.

3. **Retrieval Top-K:**

   The number of retrieved chunks injected into the context window was varied between 10, 20, 25 and 35. Higher Top-K values provide more context but risk passing the model's context window (the maximum number of tokens the model can process at once) and falling into the lost-in-the-middle effect described in Section 3.2.2.4. A lower number of chunks may omit relevant evidence.

Table 5.1 summarizes the configurations evaluated in this study. A total of ten experimental configurations were tested, varying across model family, model size, chunk size and retrieval top-K. To run these experiments, an embedding model is required to perform the different comparisons. In this case, the same embedding model was used in every experience to isolate the other variables. The model used is *qwen3-embedding:8b*. This embedding model was selected based on its performance in the NLP evaluation performed, where it achieved the highest and most consistent scores across all three report categories (see Table 5.3).

Table 5.1: Experimental Configurations Evaluated

Experiment	**Model**	**Chunks**	**Chunk Size**
llama-8b-10chunks	LLaMA 3 8b	10	800 chars
mistral_nemo-10chunks-800chars	Mistral-Nemo	10	800 chars
mistral_nemo-10chunks-1500chars	Mistral-Nemo	10	1500 chars
mistral_nemo-25chunks-1500	Mistral-Nemo	25	1500 chars
mistral_nemo-allchunks	Mistral-Nemo	All	1500 chars
qwen14b-chunks10-chars800	Qwen2.5 14B	10	800 chars
qwen14b-chunks10-chars1000	Qwen2.5 14B	10	1000 chars
qwen14b-chunks25-chars1500	Qwen2.5 14B	25	1500 chars
qwen14b-chunks35-chars1500	Qwen2.5 14B	35	1500 chars
qwen14b-allchunks	Qwen2.5 14B	All	1500 chars

By testing these variables across the experimental setups, the primary goal is to validate the overall architecture of the built system.

To thoroughly assess the system, the automated evaluation follows a two-phase development as described in Section 4.4. First, the pipeline is evaluated using the Ragas framework, featuring the described metrics in the methodology section (Faithfulness, Context Precision, Context Recall and Answer Relevance). With these metrics, Ragas allows to gather information regarding the quality of the retrieval and generation engines from the pipeline. In order to make Ragas work with our setup, a golden dataset was created. The reason why this is needed is because of the way Ragas performs the evaluations. It requires certain parameters to allow the calculations to be done and to be able to assess correctly the entire document. Since the generation is made section-by-section, Ragas will also evaluate section-by-section instead of the entire document at once. The golden dataset is composed by a subset of 6 reports from the original 23 human written reports. From these 6 reports, one is from type Initiation, Continuation, Suspension Criteria, two are from Comparison Evaluation and the last three from Single Drug Evaluations. Each report is structured into a .json file where each section is divided and composed by 4 attributes, those being:

- **Question:** The query or input provided by the user to build the section
- **Contexts:** The relevant text chunks retrieved in the retrieval phase of the pipeline.
- **Generated Answer:** The response generated by the generation phase of the pipeline.
- **Ground Truth:** The correct text for the section that is inputted. Comes from the original human written report.

The first three parameters come inherently from the end of the generation phase where for each section the created code is storing these parameters and creating a .json file to output at the end. The ground truth parameter is trickier and also the reason on why a subset of the reports was created. The generated reports are following a template that was given by the UFC and serves as the reference on what the reports should have, however, many of the human written reports given, from the past years, didn't follow the template making it hard to match each piece of text to the corresponding ground truth and section. Because of this, only reports that followed the template or were close enough to it are used to be evaluated with the Ragas framework. In this case, six reports are selected. This also comes as a limitation since this parameter is manually done and relies on the judgement of the person creating it while having major influence on the final result.

During this Ragas evaluation running section-by-section, BERTScore is also used to evaluate the semantic similarity between the generated answer and ground truth parameters. Originally the idea was to run BERTScore as the primary metric of full document evaluation, however, that isn't possible because BERTScore has a token input limit that would simply take into account the first sections and then would truncate and not provide valid results for comparisons. This way, since sections are smaller bits of text,

BERTScore is applied here to ensure its correct performance. A cosine similarity based metrics, which is identified by Semantic Similarity, replaced BERTScore in the full document assessment.

This process makes up for the first stage of the automatic evaluation. The second stage is composed by the use of generic NLP metrics that evaluate the entire textual content of both the generated and original reports, featuring:

- **ROUGE-L [37]:** Measures the longest common subsequence, evaluating the structural and syntactical overlap between the generated report and the UFC ground truth.
- **Med-F1 [32]:** A domain-specific F1-score calculating the overlap of medical vocabulary and clinical terminology.
- **Semantic Similarity:** Computes the cosine similarity of the document embeddings generated using the *qwen3-embedding:8b* model to assess whether the overall meaning remains consistent with the original report. Let u and v denote the embedding vectors of the generated report and the human-written report, respectively. The cosine similarity is defined as:

$$\text{Cosine Similarity}(u,v) = \frac{u \cdot v}{\|u\| \, \|v\|}$$

Together, these two phases act as complementary to each other. The Ragas framework will evaluate the pipeline performance considering whether or not the system can retrieve and ground information correctly and the NLP stage evaluates if the output is clinically accurate and comparable to the pharmacologists writing. After defining the evaluation protocol, the next section presents Ragas retrieval results across the ten configurations, followed by the generated document section-level analysis and interpretation.

5.2 Ragas Evaluation

While the NLP evaluation to be presented in the following section assesses the quality of the generated output by comparing it with the original expert-written reports, they do not provide information regarding the internal functioning of the pipeline that produces them. A generated report may be semantically close to an expert report but the contents that allowed the LLM to produce it can be poorly retrieved or have a weak grounding of context. This is what it is proposed to be evaluated with the Ragas framework, providing a perspective on the processes that lead to the output instead of the end product.

As described in the previous section, in order to use the Ragas framework, a golden dataset must be created. In this case only six reports are being evaluated (3 from Single Drug Evaluation, 2 representing Comparative Evaluation and lastly 1 referring to Initiation,Continuation or Suspension Criteria). It should be also highlighted, one more time, that the manual nature of the ground truth assignment

introduces a degree of subjectivity onto this evaluation stage as the selection of the text to be compared with relies on the golden dataset creator. All of the experiences presented next are aided by the *qwen3-embedding-8b* embedding model.

Table 5.2: Overall Ragas performance across different experimental setups.

Experiment	**Faithfulness**	**Context Recall**	**Context Precision**	**Answer Relevancy**
llama3.1:8b_chunks10_chars800	0.5865	0.8085	0.8640	0.6111
mistral_nemo-10chunks-1500chars	0.4885	0.4255	0.3046	0.6504
mistral_nemo-10chunks-800chars	0.3879	0.3967	0.2530	0.6700
mistral_nemo-25chunks-1500chars	0.5989	0.5687	0.2607	0.6675
mistral_nemo-allchunks	0.5003	0.6273	0.2333	0.6641
qwen2.5_14b-allchunks	0.7788	0.6168	0.3398	0.6538
qwen2.5_14b-chunks10-chars1000	0.4271	0.4065	0.3492	0.6368
qwen2.5_14b-chunks10-chars800	0.3929	0.3829	0.3085	0.6432
qwen2.5_14b-chunks25-chars1500	0.6718	0.5526	0.3393	0.6401
qwen2.5_14b-chunks35-chars1500	0.7046	0.5547	0.3109	0.6388

Table 5.2 presents the overall Ragas scores across all ten experimental configurations, covering four metrics (as described in Section 4.4). An important note before analyzing the results, all the generated reports that are being evaluated in this framework are generated with *LLaMA*, meaning that it's results may contain a little bit of bias that should be considered for the interpretations to follow.

Starting with the *LLaMA* configuration results, the *LLaMA* model stands out as the strongest performing experiment on retrieval metrics, achieving both the highest Context Recall and Precision scores of all tested configurations. Remember that for all Ragas metrics, scores range from 0 to 1, where higher values indicate a better performance. The result suggests that a smaller chunk size combined with a top-K retrieval of 10 chunks produces a more precise and relevant retrieval set. From a retrieval perspective, this means that the majority of retrieved chunks are relevant to the input query, with minimal noise being introduced to the context window of the LLM. The high Context Recall score also shows that the chunk size is also enough to capture most of the relevant information. It achieved a decent Faithfulness score showing that the retrieved content, although useful, may not being utilized correctly in the generated output. Answer Relevancy, on the other hand, underperformed, being the lowest of all experiments done meaning that the generated responses may not fully align with the respective section query.

The four *Mistral-Nemo* experiments show variable results in the retrieval quality across the different chunks and sizes configurations. A pattern can be seen in the Context Precision metric that decreases as chunk number increases.

Configurations with a smaller chunk number achieve relatively higher precision compared with the others. This shows a trade-off between the introduction of noise with more chunks that although increase the information available for the LLM, also increases the proportion of irrelevant or weak content leading to a reduction in Context Precision. This same trend can be verifiable in the *Qwen* models. The increase

in the number of chunks though and also chunk size led to a huge increase in Faithfulness indicating that providing more context does help the model to produce more grounded outputs, still coming with the cost of precision in the retrieval stage, showing a sort of lost-in-the-middle effect. Answer Relevancy stays relatively stable across all experiments showing that the output quality is consistent to the prompt regardless of the retrieval quality, implying that the model might be getting information from its internal knowledge rather than the retrieved context.

The *Qwen* model provides a more extensive coverage for chunk and chunk size variation, going from 800 to 1500 characters and top-K from 10 to 35. The results show that as we have higher chunks the Faithfulness metric skyrockets. This shows that the Qwen model benefits from receiving broader context, producing outputs that are consistent with expert written reports when given to more information. This Faithfulness score, however, does not translate into a higher Context Precision or Recall. Precision has similar values for each experiment moving in a range of 0.3095-0.3492, suggesting that some relevant information is present in the retrieved context but there might also be some noise induced. Recall, as Faithfulness, follows the pattern of increasing as chunk number increases. This indicates that the model, once again, tolerates well large contexts being passed to it, even at the cost of reducing precision score.

Answer Relevancy, similarly to *Mistral-Nemo* models, also is stable across all experiments, showing that even though the quality in retrieval and generation quality varies a little, the model produces consistent outputs for the different section queries. The lack of variation might also indicate that this metric is slightly more independent of the retrieval quality than Context Precision or Faithfulness.

To better assess the overall results of each model evaluation, mean values for each metric was calculated on all report types. Figure 5.1 presents the aggregated Ragas metric results by report type.

From this plot, it's possible to affirm similar performance across all three report categories. Answer Relevancy achieves the highest scores of every metric, indicating that for each report type, the generation phase produced decent output for each of the sections queries the model had to face. The most notable variation occurs in the retrieval based metrics. Context Precision decreases by almost half on the more complex reports. Context Recall also presents a less intuitive pattern. It would be expected for the values to keep decreasing on both Comparative and Initiation, Continuation or Suspension Criteria but strangely Comparative Evaluation achieves the highest of the three scores. A possible explanation for this is that for the comparative queries, since various pharmacological entities are being search simultaneously, a larger proportion of relevant chunks is found. Since Initiation/Suspension Criteria requires a level of contextualization that current retrieval strategy does not support, the lowest recall of the three is expected. Faithfulness remains stable across all three report types, showing once again, acceptable grounding of the answers in the retrieved context.

To understand clearly what happens in the buildup of each report evaluation, section assessment was also done. Figure 5.2 shows a view on the pipeline's behavior across all individual report sections with Ragas scores.

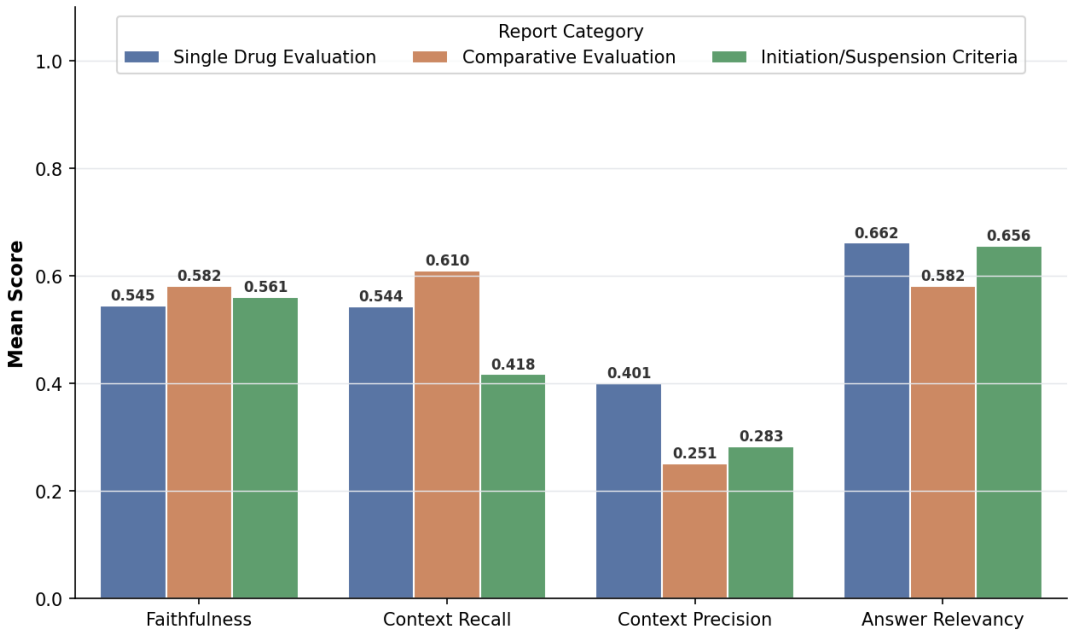


Figure 5.1: Ragas Metrics Performance by Report Type

Figure 5.2: Score per Report Section x Ragas Metric

Some patterns emerge from this analysis. As expected, sections that are grounded by the Knowledge Graph, such as Pharmacology, Posology and Indications, present higher Context Recall, Precision and Answer Relevancy since the LLM receives factual evidence reducing its interpretation claims. It is also expected for Conclusion and Final Considerations sections achieve lower scores because they are more LLM creative built sections, since they do not receive chunks relevant to answer the section query but else receive the entire document made until that moment. The Answer Relevancy in this sections show however that appropriate outputs are built. An interesting section to pay attention to is the Evidence for Initiation and Suspension that shows the highest Faithfulness score of all the metrics while also scoring really low in the retrieval metrics. This means although the retrieval engine struggled with the identification and ranking of chunks, the model stayed loyal to the context retrieved and did not resort to hallucinated information. While this shows a level of robustness in the generation phase, it also can be considerate a risk for clinical use since the content generated is faithful to an incomplete retrieved context not allowing for clear decision-making. A look at Answer Relevancy across all sections demonstrates that the pipeline is answering mostly correctly to the vast majority of section queries.

5.3 NLP Evaluation

Following the Ragas assessment at document section level, complementary NLP metrics offer a view on how the system preserves factual accuracy and clinical meaning. This section demonstrates the described NLP evaluation methodology 4.4 applied to the generated reports in three stages: BERTScore analysis at section-level, a comparison on the influence of embedding models in full document semantic and a final document evaluation using Rouge_L and Med_F1

Similar to Ragas, BERTScore is also applied to these generated reports as described in section 5.1.

Although it's used, it should be highlighted, again, that BERTScore has a max input token limitation meaning that even though we switched its use from full document to section only up to 512 tokens will be evaluated in each section. It is a limitation of the BERT architecture.

The idea of BERTScore is to understand how similar can the generated reports be to the expert written reports, this is done with the help of the *google-bert/bert-base-multilingual-cased* model. This model is trained on 104 languages, one of those being Portuguese, making it suitable for this task. It is also a general purpose model meaning that is able to capture the keywords from the pharmacology domain.

Following Figures 5.3, 5.4 and 5.5 present the mean value of BERTScore across all the report sections for Single Drug Evaluation, Comparative Evaluation and Initiation, Continuation or Suspension Criteria report. respectively.

Across Figure 5.3, referencing to Single Drug Evaluation, BERTScore ranges from 0.602 (Pharmacoeconomic Evaluation Section) to 0.739 (Pharmacology Section). The Pharmacology section achieving the highest score, as well as its complementary parts, such as , Indications and Posology are consistent with the architectural design. These sections are essentially grounded by the constructed Knowledge

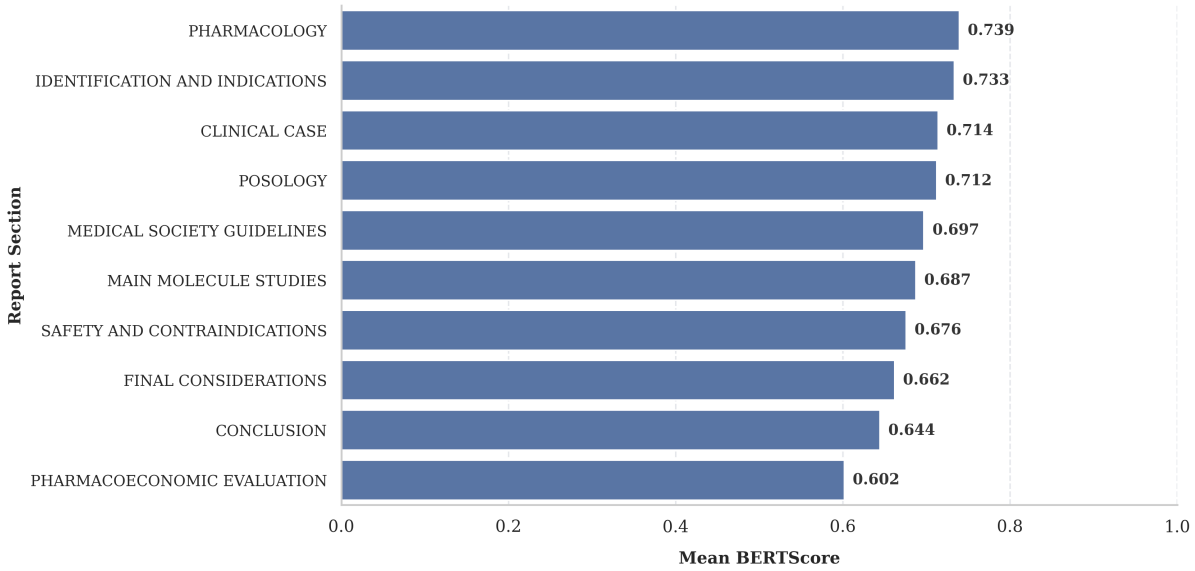


Figure 5.3: Average BERTScore performance across distinct sections within Single Drug Evaluation reports.

Graph, which provides structured and deterministic drug information reducing the LLMs variability and keeping the output closer to the pharmacologists writing.

The lower scores appearing in the Conclusion and Pharmacoeconomic are also expected. Conclusion section is essentially the model synthesizing and interpreting all the information generated until that point, giving it freedom to derive its own conclusions with a higher freedom than one the more textually grounded sections. The Pharmacoeconomic section underperforms on all report types. These is due to the retrieval process that goes into building it. Information is fetched in most part from NICE instead of only using national sources (such as Infarmed, which is the main source of this section for the UFC department). This results in the sections not aligning entirely on the content they have, leading to an inferior score.

Comparative Evaluation reports, from Figure 5.4, show a slightly different distribution. The pharmacology sections remain strong, possibly because of the grounded information retrieved from the Knowledge Graph. The Clinical Case section high score comes from the entities retrieved in the original report with NER and the following of the template document on both reports making this section in both cases equal in content. Significantly, the score achieved by the Comparative Efficacy Studies (0.699) also shows that the pipeline produces semantically adequate content even when comparing different drugs. The lowest scores refer again to the sections that build up both pharmacoeconomic analysis and the conclusion. As explained above, these sections are primarily based on a different source than the human written reports resulting on a less semantically close section build up on the generated report.

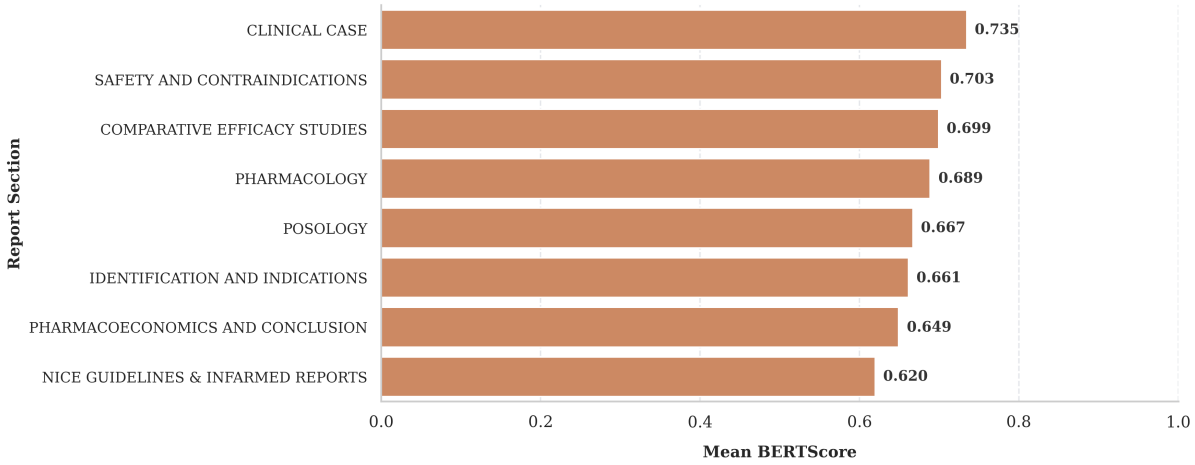


Figure 5.4: Average BERTScore performance across distinct sections within Comparative Evaluation reports.

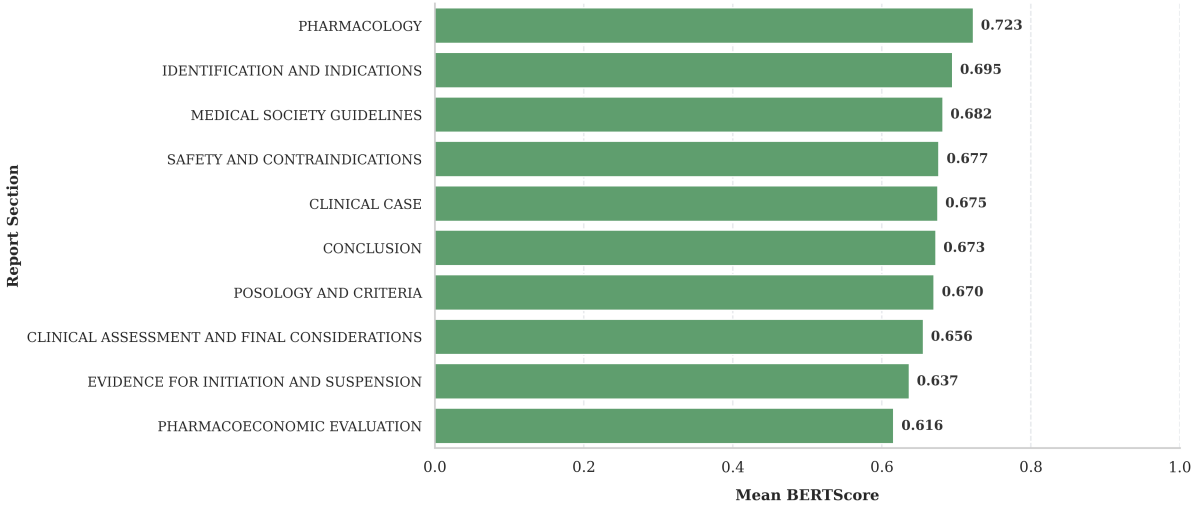


Figure 5.5: Average BERTScore performance across distinct sections within Initiation and Suspension Criteria reports.

Regarding the Initiation,Continuation or Suspension Criteria reports, a similar pattern to the previous reports is followed. Figure 5.5, shows this pattern while also its needed to highlight a particular score in the Evidence for Initiation and Suspension section. This section demands a greater contextualization with the patient's history and clinical evidence in a way the current architecture still does not fully support clearly.

Figure 5.6 shows the mean values for BERTScore in each one of the different report categories, offering a more global view of which is the best performing report. Single Drug Evaluation achieved the highest mean (0.690) followed by Comparative Evaluation (0.678) and lastly the Initiation, Continuation or Suspension Criteria (0.670). This ranking reflects the complexity effect in each one of the reports.

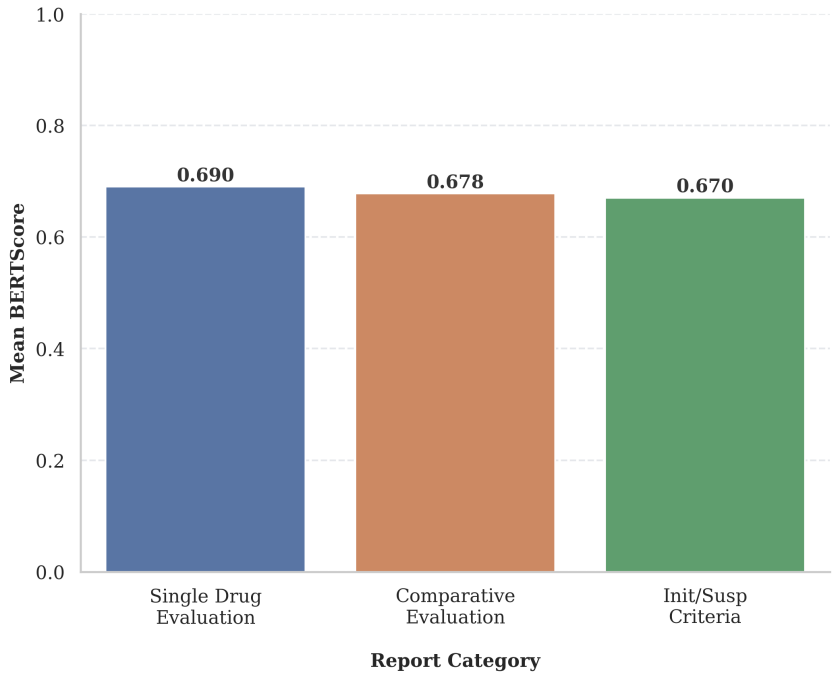


Figure 5.6: Aggregated BERTScore distributions categorized by comprehensive report type.

Single Drug Evaluation reports involve less variables and have simpler and more targeted queries and prompt templates while Comparative and Initiation,Continuation or Suspension criteria need a higher degree of contextualization and handling of multiple variables. This progressive decrease indicates also that prompt engineering and other contextualization improvements should be prioritized for better results.

Finally, Figure 5.7 aggregates all the values for each sections and shows its mean score. It validates the Pharmacology section and its subsections as the best performing section across all sections, all benefiting from the structured and deterministic content retrieved by the Knowledge Graph. At the lower end of scores, sections such as Conclusion as explained previously, are dependent on the incorporation of all the generated sections and on the LLMs interpretation of data. Other like NICE Guidelines and Infarmed Reports or Pharmacoeconomic Evaluation require the focus of retrieving to change to a more national view that the current architecture handles with less precision.

Moving onto NLP evaluation at full-document level, an approach is employed using large-context embedding models that, unlike BERT, do not have a low token constraint. Three embedding models were used to assess if the results stayed consistent across different semantic representations. Table 5.3 shows the outcome of that experience with focus on Semantic Similarity. These experiments are also done with all 23 reports generated unlike BERTScore's six reports.

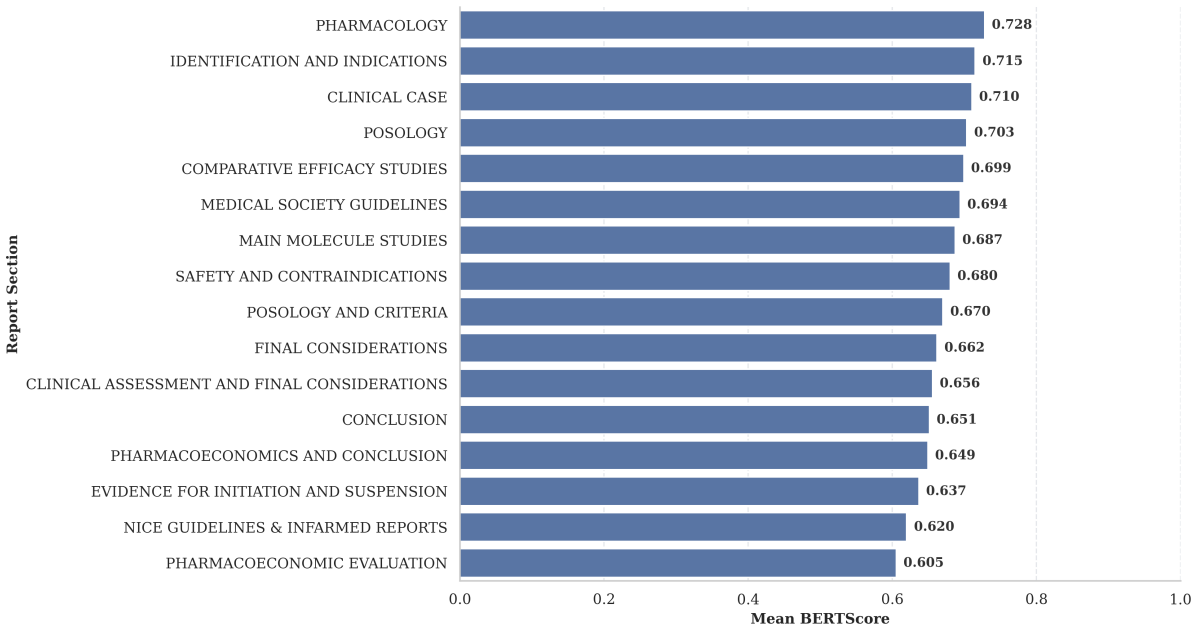


Figure 5.7: Average BERTScore per section aggregated across all report types

Table 5.3: Semantic Similarity evaluated across different massive-context embedding models configuration.

Embedding Model	**Single Drug**	**Comparative**	**Init/Susp Criteria**
bge-m3:567m	0.80	0.79	0.90
qwen3-embedding:8b	0.89	0.89	0.88
snowflake-arctic-embed2:568m	0.89	0.88	0.89

From Table 5.3, *qwen3-embedding:8b* and *snowflake-artic-embed2:568m* produce similar scores across all three report categories. This shows that the high scores for semantic similarity do not depend on a single embedding space, reflecting alignment between the generated and expert reports. The model *bge-m3:567m* scores a bit lower for the first two categories but shows the highest score for Initiation,Continuation or Suspension Criteria. This value appears as an anomaly possibly caused by the fact of having only two reports available for this category, making that score sensitive to the content present on those reports instead of a reliable number of generated reports.

Overall, full document Semantic Similarity score is in the range of 0.88-0.89 indicating that the generated reports preserve most of the clinical meaning of expert-written documents even with the different phrasing of sentences. This experience also helped considering *qwen3-embedding:8b* as the primary embedding model for Ragas evaluation.

Another experiment was done regarding Rouge_L and Med-F1 metrics. Table 5.4 presents the global scores for each full document metric used in each one of the report types.

Table 5.4: Global NLP document-level evaluation metrics categorized by pharmaceutical report type.

NLP Metric	Single Drug	Comparative	Init/Susp Criteria
Semantic Similarity	0.890	0.887	0.878
ROUGE_L	0.290	0.290	0.280
Med-F1	0.210	0.190	0.190

Both Rouge_L and Med-F1 scored low with a mean of 0.29 and 0.20, respectively. While both scores may appear bad, they must be interpreted in the context on how the reports are generated and compared. Rouge_L measures the longest common subsequence of matching keywords between generated and expert written reports. Clinical writing can be expressed in multiple ways while still referring to the same pharmacological evidence and this variability is even intensified by the translation process applied to each report section (report generated in English and then turned into Portuguese), which can lead to divergences in the text while keeping the meaning behind it. From a clinical pharmacologist perspective, a low Rouge_L score does not translate in the report being clinically inadequate. It indicates that the reports do not have the same lexical and structural approach. For a pharmacologist reviewing the generated report, they would only need to focus on editing the style and structure of the report rather than correcting factual errors (since they have high Semantic Similarity), reducing the drafting burden.

Regarding Med-F1, that measures overlap of medical vocabulary between the generated and original report, the discrepancy might be justifiable with the use of clinical synonyms or post translation terminology shifts that similarly to Rouge_L lead to a reducement of the final score but do not indicate that the content generated is clinical irrelevant.

The contrast between Rouge_L and Med-F1 low values with the consistent high scores of Semantic Similarity might indicate that the system generates content that is semantically aligned with the expected pharmacologists drafts but expressed with a different vocabulary and structure. This interpretation is supported by the qualitative findings in the upcoming Section 5.4.1

5.4 Human Evaluation

The human evaluation was carried with the help of two clinical pharmacologists from UFC of HSJ, both with significant professional experience in the preparation, writing and revision of technical pharmacology reports. To each evaluator, the same files were shared. These contained: a Google Drive that is storing three reports (one from each category), an image showing the created interface design, and a Google Forms file from where their feedback is collected. The reports were presented alongside their corresponding originals to allow direct comparison between the generated output and the human ground-truth.

The given formulary is composed by two parts. Firstly, a structured questionnaire with quantitative items using the Likert scale (ranging from 1 to 5, where lower values indicate poor quality and higher

values indicate better quality) is supplied. This questionnaire assesses clinical precision, text completeness, hallucination detections and report structure and presentation for each one of the reports. These parameters are selected since they reflect the most general criteria for evaluating the performance of the generated reports in the clinical context and to identify possible gaps corresponding to the known limitations.

The second part consists of open-ended questions, allowing evaluators to give their feedback freely on the strengths, limitations and possible improvements for each report. At the end of the quantitative evaluation of each report there is also an open-ended optional answer for feedback on it. Both evaluators were also asked on if they would use this tool on a real world case after future optimizations provided that it would serve only as a decision support system.

Table 5.5 shows the questions present in the questionnaire and the way and why they are used for evaluation.

Table 5.5: Human evaluation questionnaire (quantitative items)

Dimension	**Question (short)**	**Scale**	**Purpose**
Clinical precision	How clinically accurate and evidence-supported is the report?	1–5	Assess medical correctness and guideline alignment
Completeness	Does the report include all relevant information needed for a decision?	1–5	Measure information coverage and omission risk
Hallucination / Accuracy	Are there unsupported, incorrect, or misleading claims?	1–5	Detect factual errors and hallucinations
Structure & Presentation	Is the report clear, well organized and usable in practice?	1–5	Assess readability and professional format
Adoption potential	Would you consider using this tool in practice (after optimizations)?	Yes / No	Measure perceived operational value

Given the limited number of available experts, the use of two evaluators should be considered a methodological constraint.

5.4.1 Quantitative and Qualitative Findings

The responses from the questionnaire were averaged across both evaluators to establish a baseline for report quality. The evaluation was split across the three generated reports representing the main system tasks: Single Drug Evaluation, Comparative Evaluation, and Initiation,Continuation or Suspension Criteria.

To visualize the system performance, Figure 5.8 illustrates the mean scores across the four evaluated dimensions for each report type.

Furthermore, Figure 5.9 demonstrates the overall rating for each one of the reports sent for evaluation.

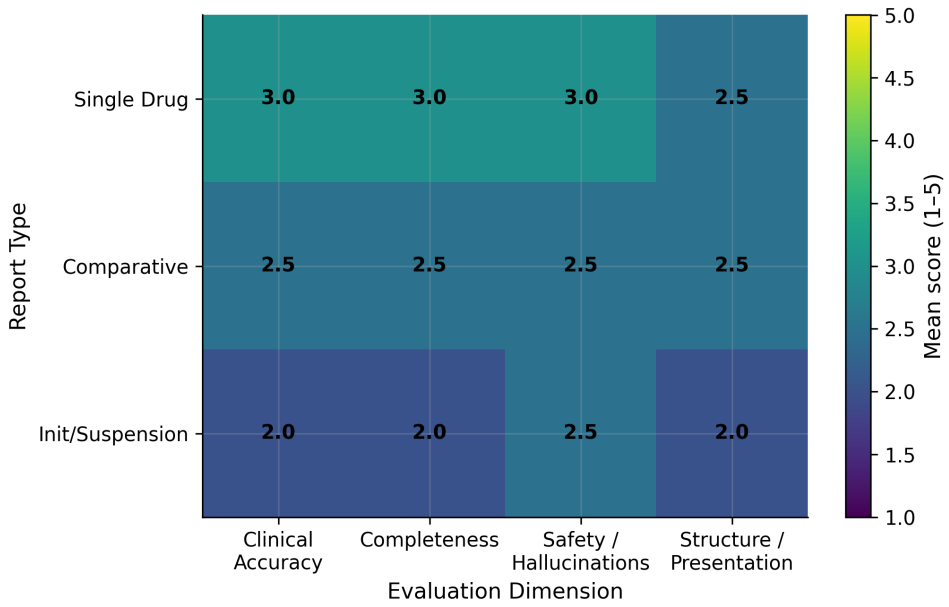


Figure 5.8: Mean Expert Evaluation Scores by Report Type

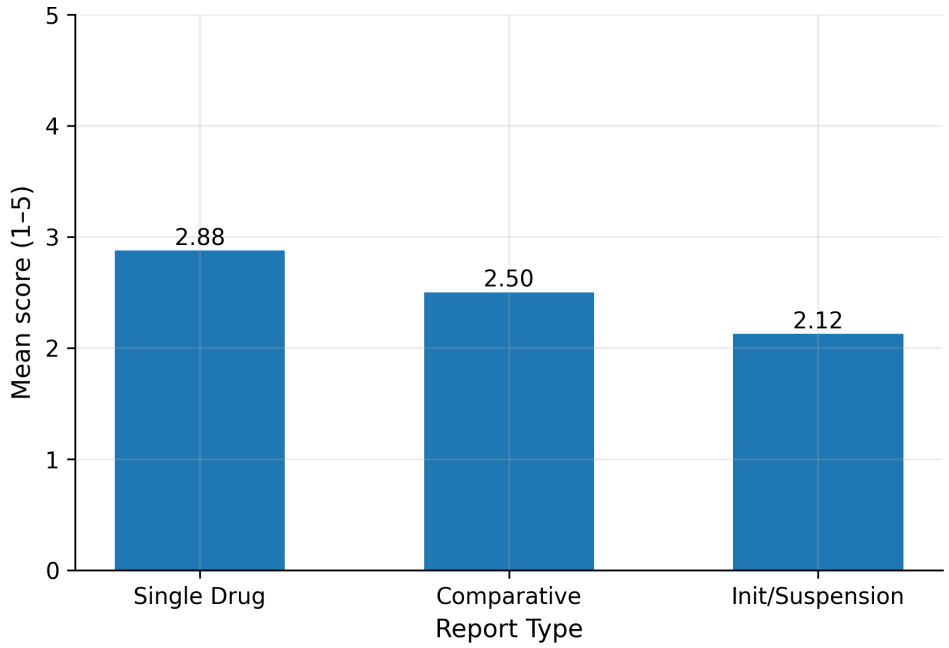


Figure 5.9: Overall Mean Evaluation Score by Report Type

Lastly, Figure 5.10 summarizes the qualitative themes identified across the open text responses from both evaluators, categorized by whether they represent a strength, a concern or a limitation

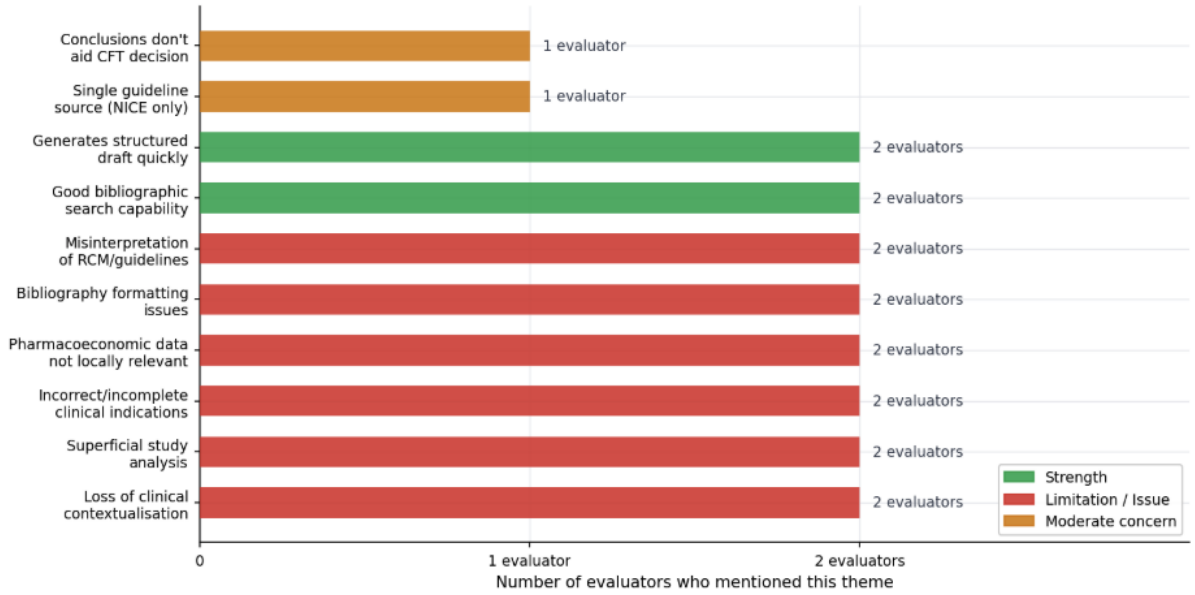


Figure 5.10: Qualitative Feedback from Open Text Comments

From these findings, several conclusions can be taken. According to the evaluators, the most acceptable report, at this stage, is the Single Drug Evaluation. It achieved the highest scores overall, reflecting the more straightforward nature of this report where, since only drug and condition are considered, the retrieved documents provide well-structured and less ambiguous pharmacological information. It is also a simpler report to generate in comparison to the other categories, which require more variables to incorporate into prompt design. Regarding the qualitative aspect of this report type, the pharmacologists highlighted three particular observations. First, they declare that although the report presents overall relevant information, it seems to lose context regarding a particular aspect of the report's goal, which was, to take into consideration a specific thematic perspective not related to the patient's condition. Second, concerns were raised referring to the pharmacoeconomic section, particularly due to the origin of the generated content (that is relying more on external societies/organizations rather than national entities such as Infarmed). Finally, the evaluators noted that some of the bibliographic references presented some inconsistencies,formatting wise, making source verification more difficult.

The Comparative Evaluation report showed moderate performance across all evaluation dimensions. Notably, evaluators assigned different scores across all questions, showing a certain degree of subjectivity in the quality of this report category. Qualitative feedback demonstrated, once again, the pharmacoeconomic section as problematic due to the reliance on non-national sources. However, the major problem identified in this report concerned the lack of development in the main studies section, which was expected to provide a detailed description of the existing literature or clinical trials for each drug as well as the comparative evidence to justify their combined evaluation. Evaluators considered this section as overly summarized, lacking on text development for supportive evidence and drug interactions. There was one hallucination being pointed out in the pharmacology section where an incorrect on-label instruc-

tion was presented and should instead have been placed within posology recommendations. Moreover, one of the evaluators highlighted a potential security risk resulting from the LLM incorrect interpretation of the RCM, leading to the recommendation of a treatment that is not allowed under certain clinical circumstances. Excessive overuse of the RCM information was also emphasized in some sections. Despite these limitations, one of the evaluators pointed that the report still represented a good draft, although lacking again on sufficient contextualization regarding the specialized thematic focus requested (almost like a secondary objective). Finally, the conclusion section was considered underdeveloped as it didn't quite present sufficiently strong arguments to support decision-making between the drug alternatives.

The final report category, Initiation,Continuation or Suspension Criteria, obtained the weakest results of all three categories. This outcome is consistent with the feedback already described, as this report type requires deeper contextualization between patient history and the specific clinical case under evaluation, which has so far been one of the common points to discuss in each report and an area of limitation within the architecture. Evaluators, once again, pointed that the LLM derived conclusions from the RCM that did not reflect real world clinical interpretation. Issues regarding the pharmacoeconomic section and the lack of contextualization of the requested specialization also persisted. Additionaly, evaluators emphasized the need for more extensive and detailed discussion of the retrieved studies related to the drug being evaluated and mentioned that the conclusions do not correspond to current evidence. Bibliography section was also identified as problematic with inconsistent numerical ordering of the references.

Moving on from the reports evaluation, evaluators were also tasked with giving their feedback regarding the entire system, including the previously mentioned question present in table 5.5 concerning the tool's adoption potential, as well as its strongest and weakest features.

From these assessments, results show that both evaluators expressed a positive opinion regarding the adoption potential of the system. One evaluator highlighted that the tool would be useful as a draft generation system needing strong revision while the other evaluator took a different position stating that the system would have great practical utility.

About the system's strongest aspects, evaluators highlighted bibliography search and the ability to generate a structured and comprehensive draft with reasonable section organization. The critical limitations refer to the section organization and document structuring, as well as, the need to enhance the detail on the selected studies for the required drugs and the contextualization loss in more complex scenarios.

5.5 Discussion

This section provides an overlook at the findings from the three evaluation phases (Ragas,NLP and Human Evaluation) and interprets them with focus on answering the research questions defined at Section 1.4. It also reflects on some constraints found during evaluation and gives further explanation at the

quantitative results given by the pharmacologists through the description of a follow-up meeting conducted with them.

5.5.1 RQ1: Is it possible to automatically generate reliable technical pharmacology reports using data retrieved from public repositories and hospital information systems?

Through the results presented so far across the automated evaluation stages, it looks as the system is capable of generating pharmacology reports that are semantically aligned with the human-written reports from UFC. The quality of these reports though can be slightly different depending on which report type is needed to generate. This claim is supported by the obtained results of Semantic Similarity (with an average of 0.88/0.89) across all report categories. Although this score doesn't guarantee an exact copy of the expert-written version of the report, it indicates that the overall clinical meaning is present. Perhaps if the translation step from English to Portuguese was removed, meaning that the reports would be delivered in English, this lexical divergences could disappear and give a closer vocabulary to the original document.

Ragas evaluation also shows a somewhat positive approval of the system in terms of being able to generate reliable reports. Answer Relevancy, the metric used to assess the ability of the pipeline to answer each section query, scores between 0.58 to 0.662 across all three types demonstrating a tolerable generation component, struggling partially on some particular sections that could use more contextualization and prompt engineering. Even looking at the sections individually, BERTScore also reinforces the generation phase ability to build each section with comparable quality to UFC written reports, even though some sections during evaluation might have been truncated due to the token limit imposed by the BERT model.

There are also the ROUGE_L and Med-F1 metrics. Although the scores obtained by them are low, it should not be understood as an evidence of failure. As explained on Section 5.3, these metrics measure vocabulary overlap which means that if the content is expressed through synonyms or through different sentence structuring it performs poorly. The contrast between these and the high Semantic Similarity suggests that the system is producing a document with equivalent clinical meaning but with distinct phrasing and maybe less professional vocabulary when compared with the original reports. This is also important due to the goal of this system which is to be a decision-support tool meaning that the pharmacologists need to make their own assessments regarding the level of appropriate vocabulary use.

Furthermore, and one of the most important parts of evaluation, Human Evaluation indicated some current system strengths, with one of those being "The ability to generate a structured and comprehensive draft with reasonable section organization" (as mentioned in Section 5.4). The quantitative results obtained through the assessment made in collaboration with the pharmacologists from UFC, however, show a mean of 2.5 score in the 1-5 Likert scale across the report types. Even though these scores are lower than expected through the automated metrics conclusions, they should be interpreted with caution.

Following the submission of the questionnaire and the obtainment of the experts evaluation, a meeting was held with both clinical evaluators to better discuss their assessments in greater depth and to allow a more detailed exchange regarding the reasoning behind their critiques.

During this meeting, each component of the pipeline presented in Section 4.3 was individually explained thoroughly, addressing questions that the pharmacologists had raised regarding the system's functioning and the limitations they had identified. During this explanation, some clarifications were made. The lack of textual depth in certain sections was attributed to the token limits imposed by the use of *Groq* free tier API during the English to Portuguese translation step, rather than a generation phase failure. It was also noted that the translation pipeline, while a reasonable approach to leverage the use of more powerful embedding models and to extract the best performance of the English trained open-source LLMs, could be reconsidered in future implementations. Delivering reports in English would remove the token constraint brought by *Groq* API and allow for a more developed section. The contextualization issues were also discussed and explained as a current architectural limitation of the use of NER, which was not designed to extract patient specific clinical details and whose associated prompts were kept general to ensure compatibility across all cases to be generated.

Concerns regarding patient data privacy were also raised and addressed. It was revealed that the patient clinical data is not an important part of the pipeline and is even an optional input part of the request handler component. Without it, the pipeline still works as it should, with the particularity of not having data to fill in the Clinical Data section. This essentially means that no patient data is required for the system to function so any data protection requirements is complied.

Another question was to understand why don't other specific societies or literature sources are used. It was clarified that the current implementation relies on the use of PubMed as the primary source and ClinicalTrials.gov as a complementary one, a constraint that comes from the lack of availability of free APIs and the extensive amount of medical societies that needed to be covered for each specific medicine field. The introduction of additional sources was identified as a priority for future development.

Following the full discussion of the system's architecture and its known limitations, both evaluators expressed that the tool already represents a strong foundation for automating the drafting of these reports, working as a decision-support system. It was concluded that with targeted improvements, with special focus on increasing the depth and quality of the studies sections the system could serve as a genuine practical utility, and that in its current state it already would serve as a useful starting point from which relevant sections could be directly used with minor adjustments and others be a basis for further development.

It is worth mentioning that this meeting does not invalidate the quantitative scores given through the questionnaire, which remain as obtained and reflect a first-impression evaluation conducted without the understanding of the system's architecture and design restrictions.

In summary, the answer to **RQ1** is a nuanced yes. The system shows the capacity to automatically generate a semantically strong and grounded report from the use of free literature sources and information systems such as INFARMED, for example. However reliability is not uniform across all

available report categories and the generated output, as anticipated, requires expert review as it serves as a decision-support tool.

5.5.2 RQ2 — What are the main challenges regarding information retrieval, contextual grounding, and data integration?

The experimental results reveal some challenges that affect the pipeline's performance at different stages.

Regarding information retrieval, Ragas evaluation shows that Context Precision is the most underperforming metric across all model configurations. This means that even though it identifies the relevant chunks, it struggles to rank them above noisier ones. It is also noted that for the *LLaMA* model, however, this metric performed really good as well as Context Recall. It is important to emphasize that Ragas metrics are computed for each configuration independently and do not involve the actual report generation pipeline. In the full system, reports were generated using a combination of Gemma (for initial English generation) and *LLaMA* (for Portuguese translation), as described in Section 4.3.4. Therefore, the strong retrieval performance observed for *LLaMA* in the isolated Ragas tests does not necessarily imply that the same retrieval quality holds when Gemma is used as the primary generator. It remains possible that *LLaMA's* generation capabilities are better aligned with the retrieval pipeline's output, or that other models interact less effectively with the same retrieval mechanism, an idea that requires further investigation. This Precision limitation is also evidenced in the *Qwen* studies, where increasing the number of retrieved chunks led to improvements in Faithfulness and Recall but reduced Precision.

Data integration presents a challenge that is perhaps the most clear finding during the evaluation. Some sections, such as Pharmacoeconomic section specially, consistently underperformed in every evaluation made. The root cause for this is identifiable and corresponds to the mismatch of sources between the generated and original report that make the system generate grounded content based on international evidence but misaligned with the national context that the CFT requires. This should not be considered a retrieval failure but instead a knowledge base gap that cannot be solved via prompt engineering or chunk optimization alone. It has also been mentioned the support of other societies to include in the system's retrieval process. As already discussed, the lack of available free APIs in the medical field presents a constraint in the attempt to fetch from multiple data sources. Most of the data used by the pipeline comes from crawlers built specifically to answer to the system's needs and to control the sources from where the information is coming, a practical solution that currently limits scalability and coverage.

The request handling layer also presents an integration challenge. The automated classification of the report using Regex and NER showed inconsistencies in the multiple tests performed in the building of this feature. Because of this, the classification is now done by the user, introducing a subjectivity task in the pipeline. Similarly, the inability to retrieve Infarmed reports due to crawler protections on the Infarmed website also forced the implementation of requiring the user to input the document manually, limiting the degree of automation achievable in the current implementation.

Finally, regarding contextual grounding and source traceability, the architectural design successfully mitigated the risk of untraceable generative claims. By associating document metadata (such as

document IDs, titles, and URLs) to each text chunk during the chunk ingestion, the pipeline preserves information throughout its processes. The dedicated post-processing step uses this embedded metadata to map in-text tags, formatting them into an organized numerical sequence of citations and constructing a bibliography section. This ensures that the generated reports remain verifiable and grounded in the retrieved medical literature.

5.5.3 RQ3 — To what level can the integration of structured knowledge sources mitigate hallucinations and improve the accuracy of generated reports?

Answering this question presents a challenge in itself, as the experimental setup evaluated the hybrid RAG pipeline as a whole system rather than conducting an isolated study on the use of the KG on and off. However, the qualitative assessments and section level evaluation results, suggest that integrating structured knowledge significantly mitigates factual hallucinations.

During the construction of the generation component and the enhancement of each report template, attempts were made to generate the sections that are now supported by the KG, without the KG itself (only using vector retrieval). In many of the cases, those sections would come up empty due to the prompts not being as targetive as they should or due to the chunk number being used wasn't enough to pick up information regarding what was needed. Even though the information was mostly present in the RCM file, the pipeline could not reliably fetch it. When switched to use the KG, that problem disappeared and all the needed information was being finally included in the final output file.

Going further from these local tests, the Knowledge Graph introduced to this system has shown, in the assessments made, a notable improvement in the section's Faithfulness and BERTScore values across all report types, a pattern that is directly of the responsibility of the inclusion of the KG (see results from Figure 5.2 and 5.7). By providing structured information with verified pharmacological facts from the EMA and FDA sources, it allows to bypass the retrieval noise that influences the vector-based sections and ensure that relevant information is consistently present in the context window.

However, the Knowledge Graph integration did not fully eliminate hallucination risk across the full report. The human evaluation identified at least one occurrence, in a Comparative Evaluation report, where the LLM produced an incorrect on-label instruction and another where a misinterpretation of the RCM led to the LLM to output a clinically unsafe recommendation. Both of these happened in sections that, although are supported by the KG, are also based on the RCM content retrieved through the vector pipeline.

This indicates that the KG, even though, enhances the quality of the output produced with the content it allows to bring, is not sufficiently enough to protect against reasoning errors that emerge during multi-source fetching or conclusions generation. Essentially, the KG contribution to this project is limited by the scope of the graph model and sections outside that scope remain vulnerable to both retrieval noise and generation reasoning errors. Expanding the Knowledge Graph to cover additional entity types and relationships would likely bring significant improvements in the other currently underperforming areas.

Chapter 6

Conclusion and Future Work

This dissertation proposed to investigate the feasibility of automatically generating technical pharmacology reports using a RAG pipeline combined with LLMs within the context of UFC at Hospital de São João, Porto. The research idea was motivated by the laborious work associated with the production of these reports, that come as a burden for the clinical pharmacologists who have to assess and synthesize vast amounts of biomedical information off their work hours.

The primary contribution of this work comes with the design, implementation and evaluation of an hybrid RAG architecture that integrates literature from public repositories (such as PubMed, ClinicalTrials.gov and NICE) with structured knowledge from a tailor-made Knowledge Graph built from regulatory sources (such as the European Medicines Agency (EMA) and the Food and Drug Administration (FDA)). The proposed architecture employs a dual-path retrieval strategy where pharmacological facts are obtained from the KG and biomedical literature is retrieved through semantic vector search over chunked documents. Furthermore, the system supports the generation of the pharmacology reports section by section in a modular way. It implements a template-driven approach where each report section is independently generated according to specific retrieval configurations, prompt instructions and data source constraints. This design choice proved itself as vital for the management of API limitations and reducing hallucination risks compared to a full single pass generation, while also enabling personalized section-level customization.

The secondary contribution comes from the evaluation work combining both automated metrics (Ragas, BERTScore, Rouge_L, Med-F1 and Semantic Similarity) with expert human evaluation facilitated by two clinical pharmacologists. This approach enabled a comprehensive assessment of the overall system, providing both quantitative measurements as well as crucial qualitative feedback into the system's strengths and limitations from an end-user perspective.

Some limitations influenced this work and should be acknowledged. Firstly, the ground truth dataset was composed by a total of 23 reports but only a subset of 6 was used for Ragas evaluation due to their section formatting and content presented. This comes as a limitation of the robustness of the quantitative conclusions especially for the Initiation, Continuation and Suspension criteria that was composed by 2

existing reports. Secondly, the human evaluation panel was composed by two clinical pharmacologists, introducing as well a limitation on the generalizability of the qualitative findings. At the pipeline construction level, reliance on free tier APIs imposed token constraints during the generation process and conversion to Portuguese text, limiting the depth of generated sections. Finally, it is important to reiterate, again that healthcare systems require high standards of reliability and transparency reinforcing that this tool should operate as a decision-support tool.

Several opportunities exist to extend and improve the work done in this dissertation. One of the most promising directions to take into improving the retrieval functioning starts with the extension of the KG to cover more relationships, such as patient-relevant contextual factors and the addition of related studies to each drug node. The incorporation of a richer knowledge graph would translate into stronger contextual grounding, improved explainability and lead to the mitigation of hallucinations. In the same subject, the retrieval process would also benefit immensely from a rework on some sections prompts and a refactor of the initial request handling process to better include patient specific information into the entities that pass for either data ingestion and the retrieval components. The removal of the translation phase would also be a step to follow, as agreed with the pharmacologists during the follow-up meeting, and would aid in the development of sections due to the removal of token limit. Alternatively, migrating the system to use paid API tiers of more powerful LLMs, such as Claude [1], would achieve a similar goal while preserving the Portuguese output. However, the high cost of commercial LLM APIs makes this option impractical for an academic project, which is why the current implementation relies exclusively on free tiers and open-source models.

Particular attention needs to be given to the Pharmacoeconomic section where NICE should be deprioritized as the primary source in favor of Infarmed information, which better reflects the national regulatory context. Lastly, the finding of new ways to introduce new medical societies literature and to handle them during the generation process would also add depth to the reports done making them more valuable for the experts to assess.

From a production standpoint, the system is already close to a deployable state. The entire pipeline has been containerized using Docker, meaning that making the tool available to UFC pharmacologists requires no complex infrastructure setup and is easily achievable. The KG retrieval path, the section-by-section generation framework and the post-processing pipeline for citation mapping and bibliography construction are the components that demonstrated the most consistent behavior across all generation tasks. The existing web interface is also a good starting point for pharmacologists to interact with the system. All of these components can be deployed immediately. On the other hand, some parts of the system can be easily replaced with better-performing alternatives. For instance, the translation pipeline can be removed entirely, as mentioned, delivering the reports in English only. The request handler could also benefit from a change in how it works, for example by replacing it with a LLM call tasked with parsing and extracting the necessary components and patient history (with careful attention to GDPR compliance). An appropriate plan to send this system to production is a phased deployment where

[1] https://platform.claude.com/

the full system is made available for the pharmacologists. However, for the two most complex report categories (Comparative Evaluation and Initiation/Suspension Criteria), iterative improvements would need to be carried out, building on future work described above. As a result, pharmacologists should expect to give more attention to these report types when reviewing generated reports, compared to Single Drug Evaluation reports which already operate at a decent quality level.

Finally, this dissertation demonstrates that it is possible to build a system that is capable of automatically generating pharmacology reports semantically grounded and structured enough to serve as a meaningful starting point for expert review. The tool, as it currently stands, represents a solid foundation that reduces the time associated with the search and drafting work of the pharmacologists and, with the mentioned improvements, has the potential to become a valuable asset in the workflow of clinical pharmacologists from UFC. The work done supplies pharmacologists with a way to dedicate their efforts into critical judgement rather than doing the whole retrieval and writing process of reports, keeping them as the final decision maker in this process.

This work is also currently being extended into a scientific article [48] intended for submission to a peer-reviewed journal (to be yet decided) in the biomedical informatics domain, further spreading the contributions and findings of this research to the scientific community.

References

[1] Sanchit Ahuja et al. "MEGAVERSE: Benchmarking Large Language Models Across Languages, Modalities, Models and Tasks". In: *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*. Ed. by Kevin Duh, Helena Gomez, and Steven Bethard. Mexico City, Mexico: Association for Computational Linguistics, June 2024, pp. 2598–2637. DOI: 10.18653/v1/2024.naacl-long.143. URL: https://aclanthology.org/2024.naacl-long.143/.

[2] Mohammad Alkhalaf et al. "Applying generative AI with retrieval augmented generation to summarize and extract key clinical information from electronic health records". In: *Journal of Biomedical Informatics* 156 (June 2024), p. 104662. DOI: 10.1016/j.jbi.2024.104662.

[3] Amazon Web Services. *Reducing hallucinations in large language models with custom intervention using Amazon Bedrock Agents*. Accessed: 2025-10-09. Amazon Web Services. 2024. URL: https://aws.amazon.com/blogs/machine-learning/reducing-hallucinations-in-large-language-models-with-custom-intervention-using-amazon-bedrock-agents/.

[4] Amazon Web Services, Inc. *What is RAG? - Retrieval-Augmented Generation AI Explained*. Accessed: 2025-10-09. Amazon Web Services, Inc. 2025. URL: https://aws.amazon.com/what-is/retrieval-augmented-generation/.

[5] Abhinand Balachandran. *MedEmbed: Medical-Focused Embedding Models*. 2024. URL: https://github.com/abhinand5/MedEmbed.

[6] British Pharmacological Society. *What is Clinical Pharmacology*. Accessed: 2025-10-04. British Pharmacological Society. 2025. URL: https://www.bps.ac.uk/about/about-pharmacology/what-is-clinical-pharmacology.

[7] Yupeng Chang et al. *A Survey on Evaluation of Large Language Models*. 2023. arXiv: 2307.03109 [cs.CL]. URL: https://arxiv.org/abs/2307.03109.

[8] Mingyue Cheng et al. *A Survey on Knowledge-Oriented Retrieval-Augmented Generation*. 2025. arXiv: 2503.10677 [cs.CL]. URL: https://arxiv.org/abs/2503.10677.

[9] Cheng-Han Chiang and Hung-yi Lee. *Can Large Language Models Be an Alternative to Human Evaluations?* 2023. arXiv: 2305.01937 [cs.CL]. URL: https://arxiv.org/abs/2305.01937.

[10] Cloudflare. *What is an LLM (large language model)?* Accessed: 2025-10-09. Cloudflare. 2025. URL: https://www.cloudflare.com/learning/ai/what-is-large-language-model/.

[11] Rahul Desai and Ananya Gupta. "The Impact of Workflow Optimization on Patient Safety and Service Quality". In: *IJRIPP - International Journal of Research and Innovations in Pharmacy Practice* 1.2 (2024). Accessed: 2025-10-09; Received: 2024-04-10; Revised: 2024-05-02; Accepted: 2024-06-18; Published: 2024-07-30. URL: https://jagunifiedinternational.in/journals/ijripp/.

[12] Yanru Dong et al. "A Fusion Model-Based Label Embedding and Self-Interaction Attention for Text Classification". In: *IEEE Access* PP (Nov. 2019), pp. 1–1. DOI: 10.1109/ACCESS.2019.2954985.

[13] Yongping Du et al. "A Novel RAG Framework with Knowledge-Enhancement for Biomedical Question Answering". In: *2024 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. 2024, pp. 3188–3191. DOI: 10.1109/BIBM62325.2024.10822837.

[14] Shahul Es et al. *Ragas: Automated Evaluation of Retrieval Augmented Generation*. 2025. arXiv: 2309.15217 [cs.CL]. URL: https://arxiv.org/abs/2309.15217.

[15] Wenqi Fan et al. *A Survey on RAG Meeting LLMs: Towards Retrieval-Augmented Large Language Models*. 2024. arXiv: 2405.06211 [cs.CL]. URL: https://arxiv.org/abs/2405.06211.

[16] Isabel O. Gallegos et al. *Bias and Fairness in Large Language Models: A Survey*. 2024. arXiv: 2309.00770 [cs.CL]. URL: https://arxiv.org/abs/2309.00770.

[17] Yunfan Gao et al. *Retrieval-Augmented Generation for Large Language Models: A Survey*. 2024. arXiv: 2312.10997 [cs.CL]. URL: https://arxiv.org/abs/2312.10997.

[18] Ranjan Ghosh et al. "Automation Opportunities in Pharmacovigilance: An Industry Survey". In: *Pharmaceutical Medicine* 34.1 (2020), pp. 7–18. DOI: 10.1007/s40290-019-00320-0.

[19] Cesar Abraham Gomez-Cabello et al. "Comparative Evaluation of Advanced Chunking for Retrieval-Augmented Generation in Large Language Models for Clinical Decision Support". In: *Bioengineering* 12.11 (2025). DOI: 10.3390/bioengineering12111194. URL: https://www.mdpi.com/2306-5354/12/11/1194.

[20] Rita González-Márquez et al. "The landscape of biomedical research". In: *Patterns* 5.6 (2024), p. 100968. ISSN: 2666-3899. DOI: https://doi.org/10.1016/j.patter.2024.100968. URL: https://www.sciencedirect.com/science/article/pii/S266638992400076X.

[21] M. U. Hadi et al. "Large Language Models: A Comprehensive Survey of its Applications, Challenges, Limitations, and Future Prospects". In: (2025). DOI: 10.36227/techrxiv.23589741.v8.

[22] Muhammad Usman Hadi et al. "A Survey on Large Language Models: Applications, Challenges, Limitations, and Practical Usage". In: URL: https://api.semanticscholar.org/CorpusID:272333785.

[23] Zakaria Hammane, Fatima-Ezzahraa Ben-Bouazza, and Abdelhadi Fennan. "SelfRewardRAG: Enhancing Medical Reasoning with Retrieval-Augmented Generation and Self-Evaluation in Large Language Models". In: *2024 International Conference on Intelligent Systems and Computer Vision (ISCV)*. 2024, pp. 1–8. DOI: 10.1109/ISCV60512.2024.10620139.

[24] Haoyu Han et al. *RAG vs. GraphRAG: A Systematic Evaluation and Key Insights*. 2025. arXiv: 2502.11371 [cs.IR]. URL: https://arxiv.org/abs/2502.11371.

[25] IBM. *What is LLM Temperature?* Accessed 11 May 2026. 2026. URL: https://www.ibm.com/think/topics/llm-temperature.

[26] Albert Q. Jiang et al. *Mistral 7B*. 2023. arXiv: 2310.06825 [cs.CL]. URL: https://arxiv.org/abs/2310.06825.

[27] Jared Kaplan et al. *Scaling Laws for Neural Language Models*. 2020. arXiv: 2001.08361 [cs.LG]. URL: https://arxiv.org/abs/2001.08361.

[28] Marko Kardum. "Rudolf Carnap–The Grandfather of Artificial Neural Networks: The Influence of Carnap's Philosophy on Walter Pitts". In: Jan. 2020, pp. 55–66. ISBN: 978-3-030-37590-4. DOI: 10.1007/978-3-030-37591-1_6.

[29] Md. Rezaul Karim et al. *From Large Language Models to Knowledge Graphs for Biomarker Discovery in Cancer*. 2023. arXiv: 2310.08365 [cs.CL]. URL: https://arxiv.org/abs/2310.08365.

[30] Enkelejda Kasneci et al. "ChatGPT for Good? On Opportunities and Challenges of Large Language Models for Education". In: *Learning and Individual Differences* 103 (Apr. 2023), p. 102274. DOI: 10.1016/j.lindif.2023.102274.

[31] Michael Klesel and H. Felix Wittmann. "Retrieval-Augmented Generation (RAG)". In: *Business & Information Systems Engineering* 67.4 (2025), pp. 551–561. ISSN: 1867-0202. DOI: 10.1007/s12599-025-00945-3. URL: https://doi.org/10.1007/s12599-025-00945-3.

[32] Sheng-Ming Kuo et al. "Automated Clinical Trial Data Analysis and Report Generation by Integrating Retrieval-Augmented Generation (RAG) and Large Language Model (LLM) Technologies". In: *AI* 6.8 (2025), p. 188. DOI: 10.3390/ai6080188. URL: https://doi.org/10.3390/ai6080188.

[33] Ngoc Luyen Le and Marie-Hélène Abel. "How Well Do LLMs Predict Prerequisite Skills? Zero-Shot Comparison to Expert-Defined Concepts". In: *arXiv preprint arXiv:2507.18479* (2025). arXiv: 2507.18479 [cs.IR]. URL: https://arxiv.org/abs/2507.18479.

[34] Miles Q. Li and Benjamin C. M. Fung. *Security Concerns for Large Language Models: A Survey*. 2025. arXiv: 2505.18889 [cs.CR]. URL: https://arxiv.org/abs/2505.18889.

[35] Yingjie Li et al. "Research on Alignment and Evaluation Methods for Large Language Models". In: *Proceedings of the 2024 Guangdong-Hong Kong-Macao Greater Bay Area International Conference on Digital Economy and Artificial Intelligence*. DEAI '24. Hongkong, China: Association for Computing Machinery, 2024, pp. 710–715. ISBN: 9798400717147. DOI: 10.1145/3675417.3675536. URL: https://doi.org/10.1145/3675417.3675536.

[36] Percy Liang et al. *Holistic Evaluation of Language Models*. 2023. arXiv: 2211.09110 [cs.CL]. URL: https://arxiv.org/abs/2211.09110.

[37] Chin-Yew Lin. "ROUGE: A Package for Automatic Evaluation of Summaries". In: *Text Summarization Branches Out*. Barcelona, Spain: Association for Computational Linguistics, July 2004, pp. 74–81. URL: https://aclanthology.org/W04-1013/.

[38] Nelson F. Liu et al. *Lost in the Middle: How Language Models Use Long Contexts*. 2023. arXiv: 2307.03172 [cs.CL]. URL: https://arxiv.org/abs/2307.03172.

[39] Suxuan Liu. "Bias Mitigation Techniques in Large Language Models: An Empirical Evaluation of Post-Training and In-Training Approaches". In: *Journal of Computer Science and Artificial Intelligence* 4 (Oct. 2025), pp. 10–19. DOI: 10.54097/kptt9j67.

[40] Nikolaos Livathinos et al. *Docling: An Efficient Open-Source Toolkit for AI-driven Document Conversion*. 2025. arXiv: 2501.17887 [cs.CL]. URL: https://arxiv.org/abs/2501.17887.

[41] I. Lopez, A. Swaminathan, K. Vedula, et al. "Clinical entity augmented retrieval for clinical information extraction". In: *npj Digital Medicine* 8 (2025), p. 45. DOI: 10.1038/s41746-024-01377-1.

[42] F. Neha, D. Bhati, and D. K. Shukla. "Retrieval-Augmented Generation (RAG) in Healthcare: A Comprehensive Review". In: *AI* 6.9 (2025), p. 226. DOI: 10.3390/ai6090226.

[43] Agada Joseph Oche et al. *A Systematic Review of Key Retrieval-Augmented Generation (RAG) Systems: Progress, Gaps, and Future Directions*. 2025. arXiv: 2507.18910 [cs.CL]. URL: https://arxiv.org/abs/2507.18910.

[44] Mohaimenul Azam Khan Raiaan et al. "A Review on Large Language Models: Architectures, Applications, Taxonomies, Open Issues and Challenges". In: *IEEE Access* 12 (2024). Cited by: 527; All Open Access, Gold Open Access, Green Open Access, pp. 26839–26874. DOI: 10.1109/ACCESS.2024.3365742. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85185128981&doi=10.1109%2fACCESS.2024.3365742&partnerID=40&md5=2b8272d893e7d9bb9ee3ee8fc17356ed.

[45] Prashanth Rao. *Vector databases (1): What makes each one different?* Licensed under CC BY-NC-SA 4.0. June 2023. URL: https://thedataquarry.com/blog/vector-db-1.

[46] JOÃO CARLOS MARQUES RIBEIRO. "IA na geração de relatórios técnico-científicos de apoio à decisão em farmacologia clínica". 2025.

[47] David K. Ryan et al. "Artificial Intelligence in Clinical Pharmacology: An Introduction for the Clinical Pharmacologist". In: *British Journal of Clinical Pharmacology* 90.3 (2024). Accessed: 2025-10-09, pp. 629–639. DOI: 10.1111/bcp.15930. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/bcp.15930.

[48] Joao Sequeira et al. "Automated Generation of Clinical Pharmacology Reports Using Retrieval-Augmented Generation and Large Language Models". Manuscript under review. 2026.

[49] Muhammad Ammar Shahid, Muhammad Afzal, and Huseyin Seker. "Extracting Health Evidence Information from Biomedical Literature using Large Language Models". In: *2024 IEEE/ACM International Conference on Big Data Computing, Applications and Technologies (BDCAT)*. 2024, pp. 159–164. DOI: 10.1109/BDCAT63179.2024.00035.

[50] Richard Shan and Tony Shan. "Retrieval-Augmented Generation Architecture Framework: Harnessing the Power of RAG". In: *Cognitive Computing - ICCC 2024: 8th International Conference, Held as Part of the Services Conference Federation, SCF 2024, Bangkok, Thailand, November 16–19, 2024, Proceedings*. Bangkok, Thailand: Springer-Verlag, 2024, pp. 88–104. ISBN: 978-3-031-77953-4. DOI: 10.1007/978-3-031-77954-1_6. URL: https://doi.org/10.1007/978-3-031-77954-1_6.

[51] Minghao Shao et al. "Survey of Different Large Language Model Architectures: Trends, Benchmarks, and Challenges". In: *IEEE Access* 12 (2024), pp. 188664–188706. DOI: 10.1109/ACCESS.2024.3482107.

[52] Karthik Soman et al. *Biomedical knowledge graph-optimized prompt generation for large language models*. 2024. arXiv: 2311.17330 [cs.CL]. URL: https://arxiv.org/abs/2311.17330.

[53] Weihang Su et al. "DRAGIN: Dynamic Retrieval Augmented Generation based on the Real-time Information Needs of Large Language Models". In: *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Ed. by Lun-Wei Ku, Andre Martins, and Vivek Srikumar. Bangkok, Thailand: Association for Computational Linguistics, Aug. 2024, pp. 12991–13013. DOI: 10.18653/v1/2024.acl-long.702. URL: https://aclanthology.org/2024.acl-long.702/.

[54] Kaiser Sun and Mark Dredze. *Amuro and Char: Analyzing the Relationship between Pre-Training and Fine-Tuning of Large Language Models*. 2025. arXiv: 2408.06663 [cs.CL]. URL: https://arxiv.org/abs/2408.06663.

[55] Thomas Yu Chow Tam et al. *A Framework for Human Evaluation of Large Language Models in Healthcare Derived from Literature Review*. 2024. arXiv: 2405.02559 [cs.CL]. URL: https://arxiv.org/abs/2405.02559.

[56] M. Onat Topal, Anil Bas, and Imke van Heerden. *Exploring Transformers in Natural Language Generation: GPT, BERT, and XLNet*. 2021. arXiv: 2102.08036 [cs.CL]. URL: https://arxiv.org/abs/2102.08036.

[57] Hugo Touvron et al. *LLaMA: Open and Efficient Foundation Language Models*. 2023. arXiv: 2302.13971 [cs.CL]. URL: https://arxiv.org/abs/2302.13971.

[58] Ashish Vaswani et al. *Attention Is All You Need*. 2023. arXiv: 1706.03762 [cs.CL]. URL: https://arxiv.org/abs/1706.03762.

[59] J. Vrdoljak et al. "A Review of Large Language Models in Medical Education, Clinical Decision Support, and Healthcare Administration". In: *Healthcare* 13.6 (2025). DOI: 10.3390/healthcare13060603.

[60] Luping Wang et al. *Parameter-Efficient Fine-Tuning in Large Models: A Survey of Methodologies*. 2025. arXiv: 2410.19878 [cs.CL]. URL: https://arxiv.org/abs/2410.19878.

[61] Velyn Wu and Jed Casauay. "OpenEvidence". In: *Family Medicine* 57.3 (Mar. 2025), pp. 232–233. DOI: 10.22454/FamMed.2024.587513. eprint: https://journals.stfm.org//media/slpjwot5/brwu0348docx-2025-03-05-17-36.pdf. URL: https://journals.stfm.org//familymedicine/2025/march/br-wu-0348/.

[62] Sheng Xu, Mike Chen, and Shuwen Chen. "Enhancing Retrieval-Augmented Generation Models with Knowledge Graphs: Innovative Practices Through a Dual-Pathway Approach". In: *Advanced Intelligent Computing Technology and Applications*. Ed. by De-Shuang Huang, Zhanjun Si, and Wei Chen. Singapore: Springer Nature Singapore, 2024, pp. 398–409. ISBN: 978-981-97-5678-0.

[63] An Yang et al. *Qwen3 Technical Report*. 2025. arXiv: 2505.09388 [cs.CL]. URL: https://arxiv.org/abs/2505.09388.

[64] Hao Yu et al. "Evaluation of Retrieval-Augmented Generation: A Survey". In: *Big Data*. Springer Nature Singapore, 2025, pp. 102–120. DOI: 10.1007/978-981-96-1024-2_8. URL: http://dx.doi.org/10.1007/978-981-96-1024-2_8.

[65] Cyril Zakka et al. *Almanac: Retrieval-Augmented Language Models for Clinical Medicine*. 2023. arXiv: 2303.01229 [cs.CL]. URL: https://arxiv.org/abs/2303.01229.

[66] Michael J. Zellinger and Matt Thomson. *Fail Fast, or Ask: Mitigating the Deficiencies of Reasoning LLMs with Human-in-the-Loop Systems Engineering*. 2025. arXiv: 2507.14406 [cs.AI]. URL: https://arxiv.org/abs/2507.14406.

[67] Genggeng Zhang et al. “A Retrieval-Augmented Framework Based on Knowledge Graphs and Vector Databases for Enhancing Large Language Model Performance”. In: *Cyberspace Simulation and Evaluation*. Ed. by Guangxia Xu et al. Singapore: Springer Nature Singapore, 2025, pp. 75–84.

[68] Tianyi Zhang et al. “BERTScore: Evaluating Text Generation with BERT”. In: *CoRR* abs/1904.09675 (2019). arXiv: 1904.09675. URL: http://arxiv.org/abs/1904.09675.

[69] Penghao Zhao et al. *Retrieval-Augmented Generation for AI-Generated Content: A Survey*. 2024. arXiv: 2402.19473 [cs.CV]. URL: https://arxiv.org/abs/2402.19473.

[70] Xuejiao Zhao et al. *MedRAG: Enhancing Retrieval-augmented Generation with Knowledge Graph-Elicited Reasoning for Healthcare Copilot*. 2025. arXiv: 2502.04413 [cs.CL]. URL: https://arxiv.org/abs/2502.04413.

Appendix A

Annexes

A.1 Human-written UFC report: Belimumab para doente com Lúpus Eritematoso Juvenil

The following document is reproduced in its original form as provided by the Unidade de Farmacologia Clínica (UFC).

UNIDADE DE FARMACOLOGIA CLÍNICA

Parecer nº 03/2024: Belimumab para doente com Lupus Eritematoso Juvenil, nefrite lúpica e provável imunodeficiência primária

ENQUADRAMENTO

A Comissão de Farmácia e Terapêutica (CFT) do Centro Hospitalar Universitário de S. João (CHUSJ) vem por este meio solicitar à Unidade de Farmacologia Clínica (UFC) parecer relativo à proposta de tratamento com Belimumab para doente com Lupus Eritematoso Juvenil, nefrite lúpica classe IV e provável imunodeficiência primária (ainda em estudo).

CASO CLÍNICO

Paciente:

Antecedentes pessoais:
Cardiopatia de Fallot
- Cirurgia corretiva em 2006, sem shunt residual
Bronquiolite Obliterante
- Pós ARDS com coinfeção Bordetella pertussis/influenza A (2006)
Colesteatoma congénito com hipoacusia a esquerda
- Mastoidectomia + Timpanoplastia (2015, revisão em 2016)
Sibilância recorrente (até aos 4 anos)
Lúpus Eritematoso Sistémico (LES) Juvenil - Diagnóstico em 2020
- Envolvimento cutâneo: Rash malar
- Envolvimento ME: Tenossinuvite
- Envolvimento renal: Nefrite lúpica classe IV

Analiticamente: ANA >1/1000 homogéneo, anti dsDNA +; anti RNP +; anti SM +, Coomb direto +; ANA SS0 e SSA
Suspeita de imunodeficiência inata
- Infeções respiratórias de repetição e desde a infância e défice de complemento persistente
- Seguida em consulta de Pediatria-Imunodeficiências
- Aguarda estudo genético

Medicação habitual:
- Mofetil Micofenolato (MMF) 3g id;
- Prednisolona (PDN) 20 mg id;
- Hidroxicloroquina (HCQ) 200 mg;
- Lisinopril 5 mg id;

- Omeprazol 20 mg id;
- Suplemento alimentar (Ferro).

História Familiar relevante:
- Avó materna falecida de cancro do pulmão (fumadora);
- Avó paterna com síndrome Sjogren e artrite reumatoide;
- Pai com síndrome Sjogren.

História da doença atual:

Doente com diagnóstico de LES em 2020, tendo sido inicialmente medicada com PDN, HCQ e azatioprina (AZA). Internada em Nov2021 por flare da doença com envolvimento renal. Biópsia revelou nefrite lúpica classe IV. Fez indução com pulsos de metilprednisolona e com MMF (suspendeu AZA), com resposta. Em Maio de 2022 encontrava-se estável, pelo que iniciou redução da dose de MMF e PDN. Desde Julho de 2023, iniciou leucoeritrocitúria e proteinuria, com parca resposta ao aumento das doses de PDN e MMF (neste momento em dose de indução).

Estudo Analítico (09/01/2024):
- Hemograma: Hb 11.8g/dL; Leucócitos 7.01x109 /L (N 74.6%; L 14.7%); Plaquetas 253000/L; VHS 27mm/1ª h;
- Proteínas totais 63.8g/L; Albumina 37.6g/L
- Cr 0.76mg/dL; TFG 112mL/min/1.73²
- Complemento: Fração C3c: 50mg/dL; Fração C4: 6mg/dL; Atividade hemolítica (CH 50) 18.2U/mL
- Anticorpo Anti Ds DNA: 306.0UI/mL.
- ESU: Proteínas 1g/L; Albumina >150.0mg/L; Cr 2000mg/L; Razão Proteína/Cr >=0.50g/g; Razão Albumina/Cr >=150mg/g; Leucócitos 50.0; Eritrócitos 209.0; Proteína totais 1.11g/L.
- Proteinúria (em 24h 0.55>0.36>0.51g);
- Proteína/Cr: 0.22.

Consulta de **Grupo Imunidade e Infeção** (09/01/2024):

- até ao momento não foi possível confirmar ou excluir o diagnóstico de erro inato da imunidade nesta doente;
- o consumo de complemento nesta doente pode ser justificado apenas pela doença autoimune de base (LES);
- a doente ainda aguarda resultado de painel NGS de imunodeficiências primárias que inclui genes do complemento C1, C2 e C4.

Dado equacionar-se o início de novo fármaco imunossupressor (agravamento sistémico incluindo da proteinúria desde há 6 meses sem resposta a aumento de corticoterapia e MMF em dose de indução) para controlo do LES e na eventualidade do estudo genético em curso revelar alguma variante genética que explique o quadro clínico, esta doente beneficia de um tratamento imunosupressor não deplector de células B. Assim, dado que a hipótese de erro inato da imunidade não se encontra completamente excluída, a utilização de belimumab poderá ter interesse face a outras alternativas farmacológicas como o rituximab.

De acordo com a avaliação da consulta de grupo, foi então solicitada autorização para prescrição de belimumab.

FUNDAMENTAÇÃO

FARMACOLOGIA

Nome Comercial e Titular da AIM: Benlysta®; GSK

Indicações *on-label*:

O Belilumab encontra-se aprovado pela *European Medicines Agency* (EMA) no tratamento(1):

- ⇒ De doentes com idade ≥ 5 anos com LES ativo, positivo para autoanticorpos, com um elevado grau de atividade da doença (por exemplo, positivo para anti-dsADN e complemento baixo) apesar de estarem a receber terapêutica padrão.
- ⇒ De adultos com nefrite lúpica em atividade, em associação outros diferentes imunossupressores.

E pela *U.S. Food and Drug Administration* (FDA) no tratamento(2):

- ⇒ De doentes com idade ≥ 5 anos com LES ativo sob terapia padrão.
- ⇒ De doentes com idade ≥ 5 anos com nefrite lúpica ativa sob terapia padrão.

Propriedades Farmacodinâmicas(1):

- Grupo farmacoterapêutico: Imunossupressores seletivos.

- Mecanismo de ação:
Belilumab é um anticorpo monoclonal IgG1 humano, específico para a proteína humana solúvel Estimuladora dos Linfócitos B (BLyS, também referida como BAFF e TNFSF13B). Atua no bloqueio a ligação do BlyS solúvel, um fator de sobrevivência celular, aos seus recetores nas células B. O fármaco não se liga às células B diretamente, mas ao ligar-se ao BLyS, inibe a sobrevivência das células B, incluindo as células B autorreactivas, e reduz a diferenciação das células B em células plasmáticas produtoras de imunoglobulinas. Os níveis de BLyS estão elevados em doentes com LES e outras doenças autoimunes. Existe uma associação entre os níveis plasmáticos de BLyS e a atividade da doença LES. A contribuição relativa dos níveis de BLyS para a patofisiologia do LES não está completamente compreendida.

- Efeitos farmacodinâmicos:
Foram observadas na semana 52 alterações nos biomarcadores em ensaios clínicos. Em doentes com hiperglobulinemia, foi observada a normalização dos níveis de IgG em 49% e 20% dos doentes a receberem Belilumab e placebo, respetivamente. Em doentes com anticorpos anti-dsDNA, 16% dos doentes tratados com Belilumab reverteram para anti-dsDNA negativos, comparativamente a 7% dos doentes a receber placebo. Em doentes com níveis de complemento baixos, foi observada a normalização de C3 e C4 em 38% e 44% dos doentes a receberem Belilumab e em 17% e 19% dos doentes a receberem placebo. Dos anticorpos antifosfolipídicos apenas foi medido o anticorpo anticardiolipina. Para o anticorpo anticardiolipina IgA foi observada uma redução de 37%, para o anticorpo anticardiolipina IgG foi observada uma redução de 26% e para o anticorpo anticardiolipina IgM foi observada uma redução de 25%. Foi observada uma redução significativa de células B circulantes, incluindo não ativadas (naive), ativadas, plasmáticas e o subtipo LES de células B.

Perfil Farmacocinético(1):

- Absorção: O Belilumab é administrado por perfusão intravenosa. As concentrações séricas máximas de belimumab foram geralmente observadas no final da perfusão ou pouco depois. A concentração máxima sérica foi 313 µg/ml (intervalo: 173-573 µg/ml) baseado na simulação do perfil concentração-tempo utilizando os parâmetros típicos para um modelo farmacocinético populacional. Após a administração por via subcutânea, a biodisponibilidade de belimumab foi aproximadamente 74%. A exposição no estado de equilíbrio foi atingida após aproximadamente 11 semanas. A concentração sérica máxima de belimumab no estado de equilíbrio foi de 108 µg/mL.

- Distribuição: Belimumab distribuiu-se para os tecidos com um volume de distribuição de aproximadamente 5 litros.

- Biotransformação: Belimumab é uma proteína para a qual a via metabólica expectável é a degradação em pequenos péptidos e aminoácidos individuais por enzimas proteolíticas largamente disseminadas. Não foram realizados estudos clássicos de biotransformação.

- Eliminação: As concentrações séricas de belimumab diminuíram de forma bi-exponencial, com um tempo de semivida de distribuição de 1,75 dias e um tempo de semivida de eliminação de 19,4 dias. A depuração sistémica foi de 215 mL/dia (intervalo: 69-622 mL/dia). Após a administração subcutânea, belimumab apresentou um tempo de semivida de eliminação de 18,3 dias e a depuração sistémica foi de 204 mL/dia.

Considerações posológicas(1):

O regime posológico recomendado da formulação intravenosa é de 10 mg/kg de Belilumab nos dias 0, 14 e 28, e seguidamente a cada intervalo de 4 semanas. A formulação subcutânea (apenas para adultos) é administrada na dose de 200 mg semanalmente no caso do LES e de 400 mg a cada semana nas primeiras 4 tomas seguido de 200 mg a partir desse momento no caso da nefrite lúpica. A descontinuação do tratamento com Belilumab deve ser considerada se não existirem melhorias no controlo da doença após 6 meses de tratamento.

Reações adversas(1):

- Muito frequentes (≥ 1/10): diarreia e náuseas, bronquite, infeção do trato urinário;

- Frequentes (≥ 1/100 e < 1/10): gastroenterite vírica, faringite, nasofaringite, leucopenia, reações de hipersensibilidade, depressão, insónia, enxaqueca, dores nas extremidades, reações relacionadas com a perfusão, febre.

Contraindicações(1):

Hipersensibilidade à substância ativa ou a qualquer um dos excipientes.

É importante salientar que o Belilumab não foi estudado nos seguintes grupos de doentes, e não é recomendado para:
- lúpus ativo grave com envolvimento do sistema nervoso central;
- VIH;
- história ou diagnóstico atual de hepatite B ou C;
- hipogamaglobulinemia (IgG < 400mg/dL) ou deficiência em IgA (IgA < 10mg/dL);
- história de transplante de órgão maior ou transplante hematopoiético estaminal/celular/medula óssea ou transplante renal.

PARECERES DE SOCIEDADES MÉDICAS E CIENTÍFICAS

EULAR(3)	LES	Todos os doentes devem ser tratados com HCQ, com ou sem glucocorticoides. Aos doentes sem resposta à HCQ ou corticodependentes, deve ser adicionado outro imunssupressor/imunomodulador (por exemplo, metotrexato, micofenolato, azatioprina, belimumab, anifrolumab). Nas formas graves com risco de vida ou de órgão, considerar ciclofosfamida; nos casos refratários, considerar o rituximab.	
	Nefrite lúpica	Indução	- Ciclofosfamida intravenosa em baixa dose ou micofenolato e glucocorticoides (pulsos intravenosos de metilprednisolona seguidos por doses orais mais baixas). - Considerar terapia combinada com belilumab (associado a ciclofosfamida ou micofenolato) ou inibidores da calcinerina (em especial voclosporina ou tacrolimus, combinado com micofenolato). Os autores ressalvam a preocupação que existe com a utilização de inibidores da calcinerina na terapêutica de manutenção dado o risco de nefrotoxicidade.
		Manutenção	O tratamento deve manter-se durante pelo menos 3 anos com o mesmo esquema utilizado em indução, exceto nos doentes tratados com ciclofosfamida que deve ser substituída por azatioprina ou micofenolato.
KDIGO(4)	Nefrite lúpica classe IV	Indução	Tratamento inicial com glucocorticoides associado com qualquer um dos seguintes: - Micofenolato; - Ciclofosfamida intravenosa em dose baixa (preferível em doentes com dificuldade de adesão a esquema oral); - Belimumab e micofenolato ou ciclofosfamida intravenosa em baixa dose (preferível nos doentes com múltiplos flares ou com alto risco de progressão para falência renal dado apresentarem doença renal crónica grave); - Micofenolato e inibidor da calcineurina quando a função renal não está gravemente comprometida (taxa de filtração glomerular estimada ≤ 45ml/min/1.73m^2), sobretudo em doentes com proteinúria nefrótica por dano podocitário; - Considerar rituximab nos doentes com doença ativa persistente ou resposta inadequada ao tratamento padrão.
		Manutenção	Tratamento de manutenção passa por micofenolato exceto: - Azatioprina, se intenção de engravidar ou intolerância a micofenolato; - Continuação do tratamento em combinação com belimumab ou eventualmente com inibidor da calcineurina se utilizado na indução.

PARECER DO INFARMED

O Belimumab encontra-se financiado para como terapêutica adjuvante em doentes adultos com LES ativo, com SELENA-SLEDAI ≥ 10, anti-dsDNA ≥ 30 UI/ml, baixos níveis de complemento (C3 baixo e/ou C4 baixo), sem evidência de lesões renais ou de compromisso do sistema nervoso central, sem resposta a terapêutica prévia com corticosteroides, antimaláricos e imunossupressores (ou existência de reações adversas a estes fármacos justificativas de interrupção de terapêutica).

Existem Programas de Acesso Precoce a Medicamentos (PAP) ativos referentes ao belimumab. Um preconiza a sua utilização como terapêutica adjuvante nos doentes com idade entre os 5 e 17 anos com LES ativo, com SELENA-SLEDAI ≥ 10, anti-dsDNA ≥ 30 UI/ml, baixos níveis de complemento (C3 baixo e/ou C4 baixo), sem evidência de nefrite lúpica ativa grave ou de envolvimento do sistema nervoso central ativo grave, sem resposta a terapêutica prévia com corticosteroides, antimaláricos e imunossupressores (ou existência de reações adversas a estes fármacos justificativas de interrupção de terapêutica). O outro é referente ao tratamento de doentes adultos com nefrite lúpica ativa, em associação com terapêuticas imunossupressoras de base, após falência ou contraindicação às alternativas terapêuticas existentes, nomeadamente o micofenolato de mofetil, rituximab, ciclofosfamida e azatioprina.

DISCUSSÃO

De acordo com as recomendações, a doente poderia adicionar ao tratamento com PDN, HCQ e MMF, o belimumab ou um inibidor da calcineurina ou suspender MMF e iniciar indução com ciclofosfamida ou rituximab. (3,4).

Tratando-se a doente de uma mulher jovem, dado o risco de toxicidade, nomeadamente reprodutiva, e não estando perante uma situação grave com risco de vida ou de disfunção orgânica, a ciclofosfamida não parece ser adequada para uma nova indução (3,4).

Não havendo proteinúria nefrótica e devido à incerteza quando à utilização de inibidores da calcineurina como terapêutica de indução e manutenção, sobretudo na população não asiática, estes também não parecem ser os mais adequados (3,4).

O rituximab é um fármaco anti-CD20, causando depleção dos linfócitos B. Por isso, foi excluído pela consulta de Grupo Imunidade e Infeção para esta doente com suspeita de imunodeficiência primária. Evidência limitada parece sugerir que o rituximab é relativamente seguro e efetivo em doentes com imunodeficiência comum variável(5). A imunodeficiência da doente ainda está em estudo, a aguardar caracterização adequada. A sua utilização no LES e nefrite lúpica é *off-label*, tendo tido resultados desanimadores nos ensaios clínicos, mas com resultados promissores em estudos observacionais.(6)

O Belilumab mostrou ser seguro e eficaz na nefrite lúpica e no LES em contexto de ensaios clínicos e da experiência de vida real. É também menos depletor de células B em comparação com o rituximab.(6)

CONCLUSÃO

Tendo em conta as recomendações internacionais de abordagem ao LES e à nefrite lúpica, as comorbilidades e o estudo atual da doente apresentada no caso clínico, a adição de belimumab ao esquema atual de imunossupressão parece ser uma adequada opção terapêutica.

Redação:
Revisão e homologação:

Unidade de Farmacologia Clínica, Fevereiro de 2024.

REFERÊNCIAS BIBLIOGRÁFICAS:

1) EMA. Resumo das Características do Medicamento – Benlysta; 2017.
2) USFDA. Highlights of Prescribing Information – Benlysta; 2017.
3) Fanouriakis, A., et al. (2024). "EULAR recommendations for the management of systemic lupus erythematosus: 2023 update." Annals of the Rheumatic Diseases **83**(1): 15-29.
4) KDIGO (2024). "Clinical Practice Guideline for the Management of Lupus Nephritis." Journal of the International Society of Nephrology **Volume 105**(1S): S1-S69.
5) Pecoraro, A., et al. (2019). "Immunosuppressive therapy with rituximab in common variable immunodeficiency." Clin Mol Allergy **17**: 9.
6) Wise, L. M. and W. Stohl (2020). "Belimumab and Rituximab in Systemic Lupus Erythematosus: A Tale of Two B Cell-Targeting Agents." Frontiers in Medicine **7**.