



UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA
UNDERGRADUATE PROGRAM IN COMPUTER SCIENCE

Isabella Viviani de Aquino

**Extracting Information from Brazilian Legal Documents with
Retrieval-Augmented Generation**

Florianópolis
2024

Isabella Viviani de Aquino

**Extracting Information from Brazilian Legal Documents with
Retrieval-Augmented Generation**

Bachelor's Thesis submitted to the Undergraduate Program in Computer Sciences of Universidade Federal de Santa Catarina for degree acquirement in Bacharel em Ciências da Computação.

Supervisor: Prof. Coorientador, Dr. Jônata Tyska Carvalho

Second Supervisor: Prof. Coorientadora, Dra. Carina Friedrich Dorneles

Florianópolis

2024

Ficha catalográfica gerada por meio de sistema automatizado gerenciado pela BU/UFSC.
Dados inseridos pelo próprio autor.

de Aquino, Isabella Viviani

Extracting Information from Brazilian Legal Documents
with Retrieval-Augmented Generation / Isabella Viviani de
Aquino ; orientadora, Jônata Tyska Carvalho, coorientadora,
Carina Friedrich Dorneles, 2024.

68 p.

Trabalho de Conclusão de Curso (graduação) -
Universidade Federal de Santa Catarina, Centro Tecnológico,
Graduação em Ciências da Computação, Florianópolis, 2024.

Inclui referências.

1. Ciências da Computação. 2. Extração de Informação. 3.
Documentos Jurídicos. 4. Retrieval-Augmented Generation. 5.
NLP. I. Carvalho, Jônata Tyska. II. Dorneles, Carina
Friedrich. III. Universidade Federal de Santa Catarina.
Graduação em Ciências da Computação. IV. Título.

Isabella Viviani de Aquino

**Extracting Information from Brazilian Legal Documents with
Retrieval-Augmented Generation**

Este Trabalho de Conclusão de Curso foi julgado adequado para obtenção do Título de “Bacharel em Ciências da Computação” e aprovado em sua forma final pelo Curso de Graduação em Ciências da Computação.

Florianópolis, dia de mês de 2024.

Banca Examinadora:

Prof. Jônata Tyska Carvalho, Dr.
Orientador
Universidade Federal de Santa Catarina

Prof^a. Carina Friedrich Dorneles, Dr^a.
Coorientadora
Universidade Federal de Santa Catarina

Prof^a. Jerusa Marchi, Dr^a.
Avaliadora
Universidade Federal de Santa Catarina

Prof. Renato Fileto, Dr.
Avaliador
Universidade Federal de Santa Catarina

A todas as mulheres que vieram, que virão e que são. Às
mães, mães solo, donas de casa, cuidadoras, faxineiras,
cozinheiras, educadoras, marginalizadas e invisíveis à
sociedade. A todas que, ao longo de suas vidas, abdicaram
e ainda abdicam de tanto para que eu e tantas outras
filhas pudéssemos hoje ocupar este espaço.

ACKNOWLEDGEMENTS

Me faltam palavras para expressar a gratidão a todas as pessoas que me acompanharam, de alguma forma, durante o período da graduação. Agradeço aos meus pais, Taiana e Antonio, por caminharem comigo na jornada da vida. Vocês sempre foram minha base fundamental, meu porto seguro e minha inspiração para continuar, mesmo quando não há ventos a favor. Meu maior orgulho é poder dizer que sou sua filha.

Agradeço aos meus irmãos, Pedro, João, Thiago e Tomás, pela companhia incontestável ao longo da vida e por serem responsáveis por grande parte da minha identidade e de quem eu sou.

Aos amigos, alguns de longa data e hoje de longa distância, e outros que tive o prazer de conhecer em Florianópolis: vocês são a lembrança diária de que a vida é extremamente especial e muito mais prazerosa quando compartilhada. Obrigada pelo suporte, pelo ombro amigo e por serem parte essencial de tudo isso.

Agradeço aos professores Jônata e Carina pela orientação e por me receberem de braços abertos no projeto Céos em um momento tão incerto da minha vida. Foi um período de muito aprendizado e amadurecimento profissional, acadêmico e pessoal.

Agradeço também ao Ministério Público de Santa Catarina que possibilitou e financiou essa pesquisa através do projeto *Céos: Inteligência Artificial em benefício da sociedade*.

ABSTRACT

Extracting information from unstructured data is a challenge that has drawn increasing attention over time due to the exponential growth of stored digital data in modern society. In addition, Large Language Models (LLMs) have recently emerged as powerful tools that benefit from this abundance of data and have shown remarkable capabilities in Natural Language Processing tasks, including question answering, summarization and extraction. Nonetheless, these models still encounter limitations on extraction tasks, such as hallucinations and restricted context windows, making it unfeasible to feed long documents into prompts. Given the foregoing, Retrieval Augmented Generation (RAG) is a novel approach that combines classic retrieval techniques and LLMs to address some of these limitations. This work proposes a workflow that allows the assessment of RAG experimental setups, including the exploration and evaluation of multiple possibilities of parameters and LLMs, to extract structured data from Brazilian legal documents related to fraud in public procurements, together with a user-friendly interface that will utilize the workflow in the background to promote a user-oriented way of extracting information from legal documents. We validated our workflow with experiments using forty legal documents and the extraction of two target variables. The best results obtained with our workflow showed an average extraction accuracy of 92.5%, significantly outperforming a regular expression strategy, with 58.75% average accuracy. Furthermore, our results show that each extracted variable potentially holds an optimal combination of parameters, highlighting the context-dependency of each extraction and, therefore, the proposed workflow's usefulness. Finally, our work poses a promising approach on extracting information by entities from different backgrounds and expertise, allowing the usage of RAG pipelines on a higher level of abstraction when utilizing the interface.

Keywords: RAG; LLMs; information extraction; legal documents.

LIST OF FIGURES

Figure 1 – Example of entities to be extracted in a text excerpt.	15
Figure 2 – Relationship between Artificial Intelligence, Machine Learning and Deep Learning.	16
Figure 3 – Example of an artificial neural network.	18
Figure 4 – RAG approach with PDF documents.	21
Figure 5 – Example of a representation of word vectors in a vector space, in which semantically similar words have approximated values.	22
Figure 6 – Main workflow overview.	27
Figure 7 – Example of file found in MPSC database and used in our dataset. . . .	31
Figure 8 – Embeddings creation overview.	32
Figure 9 – Single extraction pipeline overview.	33
Figure 10 – Single extraction pipeline overview.	33
Figure 11 – English translation of prompt template used in every experiment. . . .	34
Figure 12 – Best and Worst results per model vs Regular Expression	37
Figure 13 – Top three best results configurations per LLM.	38
Figure 14 – Top K evolution with fixed chunk size as 128 and chunk overlap as 20. .	39
Figure 15 – Models frequency in top 50 experiments extracting public procurement process identifiers.	40
Figure 16 – The document upload interface, allowing drag-and-drop functionality for uploading <code>.pdf</code> documents.	42
Figure 17 – The variable creation interface, allowing users to create variables with custom prompts and labels.	43
Figure 18 – The annotation interface, allowing users to annotate subset of documents for each created variable.	44
Figure 19 – The parameter customization interface, allowing users to configure the workflow parameters.	45
Figure 20 – The extraction results interface.	46
Figure 21 – Example of NPA document found in MPSC database and used in our dataset, with an example of the chosen variable highlighted.	48
Figure 22 – Partial output evaluation detail on extracting the desired variable. . . .	49

LIST OF TABLES

Table 1 – Comparative Summary of Related Work (Part 1)	25
Table 2 – Comparative Summary of Related Work (Part 2)	25
Table 3 – RAG Parameters	28
Table 4 – Alternating Parameters	30
Table 5 – Set Parameters	30
Table 6 – Comparison of best obtained accuracies across new and previous models.	41
Table 7 – Columns of the results summary file, obtained from extracting the pro- posed variable for each experiment scenario.	48

LIST OF ABBREVIATIONS AND ACRONYMS

LLM	Large Language Model
MPSC	Ministério Público de Santa Catarina
RAG	Retrieval-Augmented Generated

CONTENTS

1	INTRODUCTION	12
1.1	OBJECTIVES	13
1.1.1	General Objective	13
1.1.2	Specific Objectives	13
1.2	ORGANIZATION	14
2	THEORETICAL BACKGROUND	15
2.1	INFORMATION EXTRACTION AND RETRIEVAL	15
2.2	MACHINE LEARNING AND DEEP LEARNING	16
2.2.1	Types of Machine Learning	17
2.2.2	Deep Learning and Neural Networks	17
2.2.3	Training Neural Networks	18
2.3	LARGE LANGUAGE MODELS	19
2.4	RETRIEVAL-AUGMENTED GENERATION	20
2.4.1	Sentence Embeddings	21
3	RELATED WORK	23
3.1	TRADITIONAL INFORMATION EXTRACTION APPROACHES	23
3.2	INFORMATION EXTRACTION IN THE LEGAL DOMAIN	23
3.3	NLP-DRIVEN INFORMATION EXTRACTION	24
3.4	CROSS-LANGUAGE INFORMATION EXTRACTION	24
3.5	PROPOSED CONTRIBUTIONS	24
4	WORKFLOW FOR INFORMATION EXTRACTION	26
5	WORKFLOW EXPERIMENTAL EVALUATION	30
5.1	DATA PREPARATION	30
5.2	EMBEDDINGS CREATION	32
5.3	EXTRACTING VARIABLES	32
5.4	EVALUATION METRICS	33
5.5	EVALUATED EXTRACTED VARIABLES	34
5.5.1	Public Procurement Process Identifier	34
5.5.2	Municipality Of Irregularity	35
6	WORKFLOW RESULTS AND DISCUSSION	36
6.1	EXTENDED RESULTS WITH RECENT MODELS	41
7	INTERFACE	42
7.1	UPLOADING DOCUMENTS	42
7.2	CREATING VARIABLES	43
7.3	ANNOTATING DOCUMENTS	44
7.4	CONFIGURING RAG PARAMETERS	45
7.5	EXTRACTING RESULTS	46

8	EXTENDED RESULTS	47
8.1	EXPERIMENT SETUP	47
8.2	EXTRACTION RESULTS	47
8.2.1	Type of legal procedure	48
8.3	DISCUSSION	50
9	CONCLUSION AND FUTURE WORK	52
9.1	FUTURE WORK	52
	REFERENCES	54
	APPENDIX A – Artigo do TCC	58
	APPENDIX B – CÓDIGO FONTE DO TCC	67

1 INTRODUCTION

The increasing digitization of judicial and administrative processes worldwide has led to massive production and storage of legal documents. These documents are commonly complex, diverge drastically in number of pages and structure hierarchy, and contain crucial information for lawyers, judges, and prosecutors. Extracting this information typically requires extensive human annotation and management in external systems such as relational databases. In line with this scenario, several efforts have been made to handle and process legal documents in various countries, for instance, explored in (BACH; AL., 2019) for extracting references from Vietnamese legal documents, and in (VIANNA; AL., 2022) for examining the processing and summarization of Portuguese legal documents.

In particular, the Brazilian public legal sector is an example of an organization dealing with great amounts of documents; almost 200,000¹ public procurement processes were successfully contracted from 2020 to 2023 by the Brazilian Federal Government, in which each one of them requires thorough documentation to formalize every step of the process. Consequently, each contract is associated with multiple documents encompassing various stages, including planning, solicitation, evaluation, award, contract execution, and monitoring. As a result, retrieving and extracting specific information, such as legal processes, contract identifiers, and involved municipalities, from these numerous complex documents poses a demanding task if done manually.

Moving forward, information extraction (IE) is an extensively researched subject in legal domains to overcome the presented challenges and has been applied and evaluated through multiple approaches, such as traditional pattern matching (CHENG; AL., 2009). Likewise, the work presented in (KOWSRIHAWAT; AL., 2015) achieves expressive results in extracting variables in legal documents through a proposed framework utilizing regular expressions. Overall, IE in legal domains is a rapidly evolving field with the potential to transform how legal professionals work. Automating information extraction can provide valuable insights from legal data, improving efficiency and better decision-making.

Then, Natural Language Processing tasks benefit from the large corpus² of available text to be analyzed, structured, and processed. Large Language Models have emerged as a transformative technology in the field, offering the promise of understanding and generating human-like text at scale and in the legal domain (KATZ; AL., 2023). However, despite their impressive performance and variety of applications, LLMs still face inherent limitations when extracting structured information from unstructured data sources such as PDF documents. LLMs knowingly struggle with domain-specific or knowledge-intensive tasks (KANDPAL; AL., 2023), have their performance degraded when dealing with relevant information in the middle of long contexts (LIU, N. F.; AL., 2023), such as legal documents, and finally tend to produce "hallucinations" (HUANG; AL., 2023) when searching for

¹ <https://portal.datransparencia.gov.br/licitacoes>

information beyond their training data.

In response to these challenges, Retrieval Augmented Generation (RAG) has emerged as a promising approach for enhancing the capabilities of LLMs in information extraction tasks (GAO; AL., 2024). By combining classic retrieval techniques with LLMs, RAG systems enable the retrieval of relevant information from external sources during text generation, thereby mitigating domain-specific and context window limitations of LLMs and improving the accuracy and coherence of the generated text.

This work proposes:

- a workflow that leverages local LLMs within RAG pipelines to extract and structure information from Brazilian legal documents related to fraud in public procurement processes. However, RAG is a data-driven general framework, and its setup can be demanding once several different parameter types are required to be set beforehand. Our objective is to demonstrate the effectiveness of RAG in overcoming the challenges associated with information extraction from complex, domain-specific documents and propose a workflow that evaluates and indicates the best RAG parameter configurations for extracting a given variable. We extracted and evaluated two different variables of interest in forty different Brazilian legal documents utilizing the proposed workflow. The results showed the proposed methodology's effectiveness, which achieved an average accuracy of 92.5%, outperforming a baseline strategy based on regular expressions, which achieved 58.5% and;
- An interface designed for non-technical users, such as legal professionals, to leverage the proposed RAG workflow for extracting information from long and complex legal documents. This interface facilitates document annotation and allows users to extract variables from any chosen set of documents, streamlining the process for those outside the technical field.

1.1 OBJECTIVES

The following sections are described the main and the specific objectives of this work.

1.1.1 General Objective

Propose an extractor tool to be used by legal entities to extract information from legal documents, utilizing the proposed workflow.

1.1.2 Specific Objectives

- Propose a RAG workflow for extracting information from legal documents and obtaining the best parameter configurations for each variable.

- Run experiments to validate the proposed workflow.
- Propose a user-friendly interface for legal entities to extract and manage information from complex legal documents, enabling annotation and variable extraction without requiring a technical background.
- Validate the interface running experiments on a different dataset.

1.2 ORGANIZATION

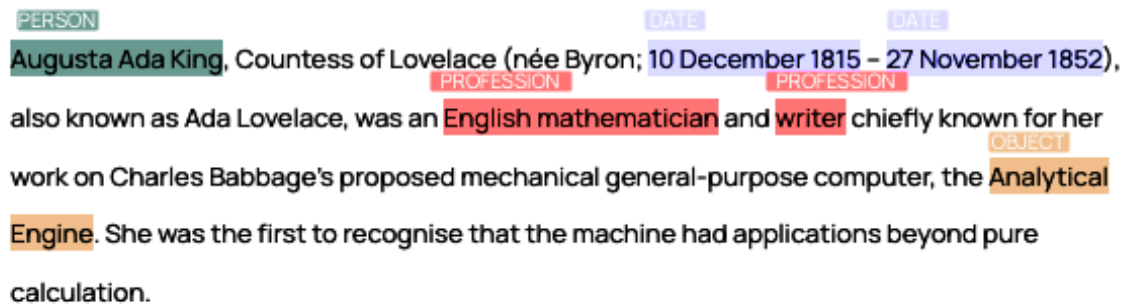
The rest of the work is structured as follows. Chapter 1 offers an overview of the different applications and approaches of information extraction and its application in legal domains. Then, Chapter 2 outlines the theoretical background for the current work while Chapter 3 depict related work. Chapter 4 defines and details the proposed RAG workflow, detailing it step-by-step. Chapter 5 introduces details of how the experiments were conducted using the proposed method and the obtained results evaluation. The concluding analysis around the pipeline and experimental results are discussed in Chapter 6. Chapter 7 presents the interface, with images illustrating key steps: document upload, variable creation, annotation, RAG parameter setup, and extraction results. On the same note, Chapter 8 details of extended experiments and results obtained utilizing the proposed interface in order to validate it. Finally, the concluding analysis, limitations discussion, and future work are discussed in Chapter 9.

2 THEORETICAL BACKGROUND

In this chapter, we explore fundamental concepts and methodologies central to the extraction of structured information from unstructured data sources. We introduce and discuss key advancements in Natural Language Processing (NLP) and the emerging role of large language models (LLMs) in tackling these challenges. Specifically, we address the integration of Information Retrieval (IR) techniques with generative models, as well as the incorporation of domain-specific knowledge to improve the accuracy of Information Extraction (IE) systems. These advancements have proven pivotal in addressing the inherent complexities of processing legal documents. We will then examine how Retrieval-Augmented Generation (RAG) systems and sentence embeddings can enhance the capabilities of LLMs, making them more effective at handling specialized tasks, such as extracting relevant data from legal corpora.

2.1 INFORMATION EXTRACTION AND RETRIEVAL

Information Extraction (IE) refers to the process of automatically identifying and structuring relevant data from unstructured text sources. The goal of IE is to transform unstructured text, such as documents, web pages, or transcripts, into a structured format that can be easily analyzed and queried. By identifying specific entities, relationships, and events within a text, IE systems facilitate downstream applications, including question answering, summarization, and knowledge base population.



PERSON
Augusta Ada King, Countess of Lovelace (née Byron; 10 December 1815 – 27 November 1852),
PROFESSION
also known as Ada Lovelace, was an English mathematician and writer chiefly known for her
OBJECT
work on Charles Babbage's proposed mechanical general-purpose computer, the Analytical
Engine. She was the first to recognise that the machine had applications beyond pure
calculation.

Figure 1 – Example of entities to be extracted in a text excerpt.

Source: The Author.

Common techniques in IE include named entity recognition (NER), which focuses on identifying and classifying nouns and proper nouns in text as entities such as people, organizations, dates, and locations; relationship extraction, which aims to identify and categorize relationships between entities; and event extraction, which seeks to detect occurrences described within a text, often along with temporal or causal information. Advanced IE models leverage machine learning approaches, including supervised models

trained on labeled datasets, as well as pre-trained language models fine-tuned for specific extraction tasks (MINTZ et al., 2009; LI, X. et al., 2019; LUAN, Y. et al., 2018).

The integration of information retrieval (IR) methods with IE has further enhanced the ability to extract relevant data from large corpora efficiently. IR systems prioritize retrieving the most relevant documents or passages in response to a query, which can then be processed by IE modules to distill structured information. For example, in legal document analysis, IR techniques might surface the most pertinent cases, while IE techniques extract structured data, such as legal statutes or precedents, from within those cases.

Information Extraction systems often face challenges when dealing with domain-specific texts, such as legal, medical, or technical documents, due to the complexity and specificity of the language used. Incorporating domain-specific knowledge bases, rule-based systems, and transfer learning methods has proven useful in mitigating such challenges (GANIN et al., 2016; SONG et al., 2021). As IE continues to evolve, it increasingly relies on deep learning approaches, making use of pre-trained transformers and hybrid models to achieve greater accuracy and adaptability across diverse applications.

2.2 MACHINE LEARNING AND DEEP LEARNING

Machine Learning (ML) is a branch of artificial intelligence (AI) and (MITCHELL, 1997) defines as the study of algorithms that improve automatically through experience. Instead of being explicitly programmed with task-specific rules, ML algorithms identify patterns and relationships within data, allowing them to make predictions or perform tasks autonomously. This ability to learn from experience is what makes ML a powerful tool for a variety of applications, including natural language processing, speech recognition and computer vision.

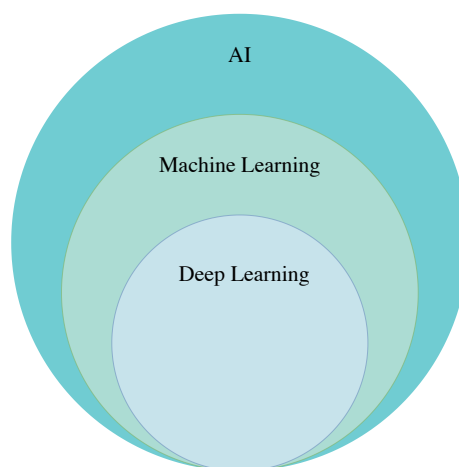


Figure 2 – Relationship between Artificial Intelligence, Machine Learning and Deep Learning.

Source: The Author.

2.2.1 Types of Machine Learning

Machine Learning can be broadly categorized into three main types: supervised, unsupervised, and reinforcement learning.

- **Supervised Learning:** In supervised learning, the algorithm is trained on labeled data, meaning that each data point has an associated correct output. The model learns to map inputs to outputs based on these examples and can then generalize to make predictions on new data. Common tasks for supervised learning include classification (e.g., classifying legal documents) and structured output prediction.
- **Unsupervised Learning:** In unsupervised learning, the algorithm is trained on data without labeled outputs. Instead, it identifies patterns or groupings within the data. Clustering (e.g., grouping customers by purchasing behavior) and dimensionality reduction (e.g., reducing the number of variables in a dataset while preserving important information) are common applications of unsupervised learning.
- **Reinforcement Learning:** In reinforcement learning, the algorithm involves training an agent to make sequences of decisions by rewarding or penalizing it based on its actions. This type of learning is commonly used in environments where the agent must interact with its surroundings, such as in game playing or robotic control.

2.2.2 Deep Learning and Neural Networks

Deep learning is a subfield of machine learning that focuses on algorithms inspired by the structure and function of the human brain. Conventional machine-learning techniques were often constrained in their ability to process natural data in their raw form, often requiring substantial manual feature engineering to extract relevant information (LECUN; BENGIO; HINTON, G., 2015). In this sense, representation learning emerged as a critical advancement for the rise of deep learning. It is a set of methods which enable models to automatically learn and capture useful features from raw data, thus reducing the need for extensive human intervention. Then, deep learning compose stacks of multiple layers that transform the representation in terms of the other, simpler representations.

A deep-learning architecture is often represented by deep learning neural networks. They are consisted of stacks of layers of simple, learnable modules. Each layer refines the input by transforming it through a series of mathematical operations, allowing the network to learn complex patterns and representations. Neural networks are inspired by the structure of the human brain, consisting of interconnected layers of "neurons" (or nodes) that process data.

A basic neural network has three types of layers:

- **Input Layer:** The input layer receives the initial data, such as text, images, or numerical values. Each node in this layer represents a feature of the input data.

- **Hidden Layers:** Between the input and output layers are hidden layers, where most of the processing happens. Each node in a hidden layer takes inputs from nodes in the previous layer, applies weights, and passes the result through an activation function. This allows the network to learn complex relationships in the data.
- **Output Layer:** The output layer produces the final result, which can be a prediction, classification, or other outcome, depending on the task.

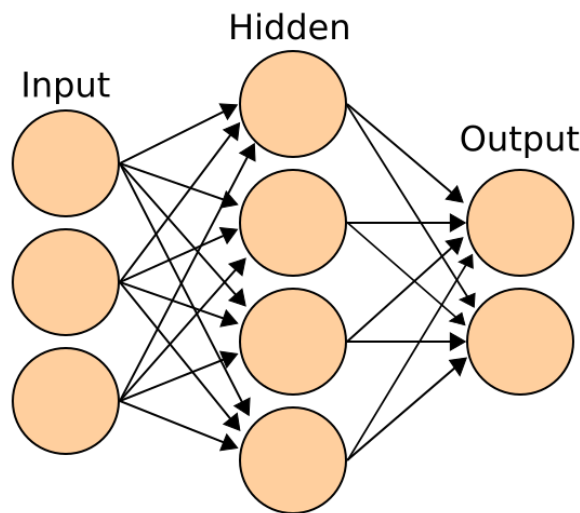


Figure 3 – Example of an artificial neural network.

Source: Cburnett, Wikimedia Commons, licensed under GFDL.

2.2.3 Training Neural Networks

Training a neural network involves adjusting the weights of connections between nodes to minimize the difference between the network's predictions and the actual values. This is done through an optimization process known as backpropagation, introduced by (RUMELHART; HINTON, G. E.; WILLIAMS, 1986), which allows neural networks to learn from its errors by efficiently computing the gradient of the loss function with respect to the weights of the network.

This is achieved through a two-phase process: the forward pass and the backward pass. In the forward pass, the input is passed through the network to generate an output, which is then compared to the target value to compute the error. During the backward pass, the error is propagated back through the network layers, allowing for the calculation of gradients using the chain rule of calculus. These gradients are subsequently used to update the weights, minimizing the loss function and improving the model's accuracy over time.

Finally, one of the most relevant advancements in this area is the use of large language models. These models, which will be introduced in the next section, have shown

to be particularly effective for tasks like text classification, named entity recognition, and information retrieval, making them highly suitable for extracting meaningful data from complex texts and documents.

2.3 LARGE LANGUAGE MODELS

Historically, there has been a persistent research challenge to allow computers to read, write, and communicate like humans (TURING, 1950). Language modeling is one of the approaches to advancing language intelligence of machines and aims to reproduce the generative likelihood of word sequences, predicting the probabilities of tokens ahead (ZHAO et al., 2023).

Following that, Large Language Models represent a significant leap in the capabilities of language modeling and natural language processing systems. These models, built primarily on transformer architecture (VASWANI, 2017), utilize self-attention mechanisms to process and generate human-like text. Unlike traditional models that rely on fixed-length context windows, LLMs can consider long-range dependencies in text, enabling them to understand context and semantics more effectively. This capacity allows LLMs to excel in various applications, from conversational agents to advanced text generation, demonstrating a remarkable ability to produce coherent and contextually relevant content.

The recent success of large language models and their increasing integration into daily tasks, popularized by *ChatGPT*¹, can be attributed to several key factors. A significant advancement has been in scaling: larger models and increased computational resources have led to enhanced capabilities and more accurate results (KAPLAN et al., 2020; WEI, J. et al., 2022; HOFFMANN et al., 2022). Additionally, improvements in training techniques have allowed these models to better generalize across tasks, making them more adaptable and effective in diverse applications (BROWN, 2020; XIE et al., 2020).

Training LLMs typically involves unsupervised learning on vast corpora of text data, where they learn to predict the next word in a sentence given its preceding context (DEVLIN, 2018; RADFORD, 2018; RADFORD et al., 2019; BROWN, 2020). This process not only helps the models capture grammatical structures and vocabulary but also allows them to internalize world knowledge reflected in the training data. These advancements underscore the role of model scaling and training innovations in driving the success and versatility of LLMs, enabling them to perform a wide range of complex tasks with remarkable accuracy and adaptability.

¹ <https://openai.com/blog/chatgpt/>

2.4 RETRIEVAL-AUGMENTED GENERATION

As introduced in Chapter 1, despite their impressive performance and wide range of applications, large language models (LLMs) still exhibit inherent challenges when tasked with extracting structured information from unstructured sources, such as PDF documents. LLMs often struggle with domain-specific or knowledge-intensive tasks (KANDPAL; AL., 2023), experience performance degradation when processing relevant information embedded within lengthy contexts, such as legal documents (LIU, N. F.; AL., 2023), and frequently generate "hallucinations" (HUANG; AL., 2023) when attempting to access information beyond their training data.

In attempt to address some of these limitations, Retrieval-Augmented Generation is a hybrid approach in natural language processing that combines information retrieval mechanisms with generative capabilities to enhance the accuracy and relevance of generated responses. Unlike conventional language models that rely solely on pre-existing training data, RAG models leverage an external knowledge base during inference to retrieve relevant context through information retrieval and integrate it into generated outputs. This strategy allows for up-to-date, factual, and contextually grounded responses, which is particularly valuable when dealing with complex or highly specialized queries, such as extracting information from legal documents.

In RAG systems, the process typically involves two main components: a retriever and a generator. The retriever identifies relevant passages or documents from a predefined corpus based on the input query, often using similarity search techniques and dense embeddings to find contextually pertinent information. Once the retrieval step is completed, a generator model, usually a transformer-based LLM, processes the retrieved content and formulates a coherent response that incorporates both the query's context and the external data. This modular design allows RAG systems to overcome the limitations of closed-model architectures by making it possible to update their responses with new knowledge without requiring extensive retraining of the base model (LEWIS, P. et al., 2020; IZACARD; GRAVE, 2020). Figure 4 presents a generic example for a RAG approach with PDF documents.

The use of RAG is particularly advantageous in domains such as legal document processing, where accuracy, context-awareness, and adherence to complex legal language are essential. Legal texts are characterized by dense terminology, extensive cross-referencing, and evolving case law, making traditional model training approaches insufficient for accurate interpretation and extraction. By integrating retrieval mechanisms, RAG models can access current statutes, case precedents, and domain-specific regulations during generation, leading to more precise and contextually aligned outputs (NOGUEIRA; CHO, 2019; LEWIS, P. et al., 2020).

For applications in extracting information from Brazilian legal documents, RAG models hold substantial promise due to their ability to address the diversity and complexity

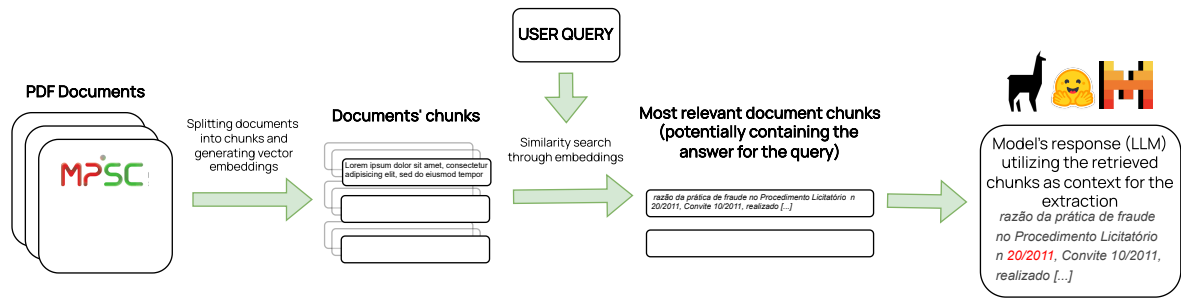


Figure 4 – RAG approach with PDF documents.

Source: The Author.

of legal norms in Brazil. Legal practitioners can benefit from systems that not only generate summaries, insights, and legal arguments but also ensure that these outputs are grounded in relevant legislative texts and case law, thereby enhancing both efficiency and reliability in legal research and document processing.

2.4.1 Sentence Embeddings

Sentence embeddings play a crucial role in the operation of Retrieval-Augmented Generation (RAG) systems by providing a numerical representation of text in a high-dimensional vector space. These embeddings are designed to capture the semantic meaning of entire sentences or passages, enabling the RAG system to perform more accurate and context-aware retrieval. Unlike traditional token-based representations, which often focus on individual words, sentence embeddings encapsulate the broader context and relationships between words in a sentence, making them highly effective for downstream tasks such as information retrieval, clustering, and similarity matching.

Modern approaches to generating sentence embeddings often involve pre-trained transformer models, such as BERT (Bidirectional Encoder Representations from Transformers) (DEVILIN, 2018), RoBERTa (LIU, Y., 2019), and BERTimbau (SOUZA; AL., 2020), a BERT model pre-trained in Brazilian Portuguese data utilized in this work. These models are typically fine-tuned using contrastive learning objectives to ensure that similar sentences produce closely aligned embeddings in the vector space, while dissimilar ones are positioned farther apart. This capability enhances retrieval precision, as the system can identify not just lexical matches but also deeper semantic similarities across texts.

In the context of RAG, sentence embeddings allow for efficient retrieval of relevant data from vast corpora by comparing dense vector representations rather than relying solely on keyword matching. This is especially beneficial for legal document extraction, where precise semantic matching is critical due to the highly nuanced and context-specific language used in legal texts (KARPUKHIN et al., 2020). Additionally, embeddings facilitate improved generalization across queries, helping the RAG system adapt to various

input formats and contexts.

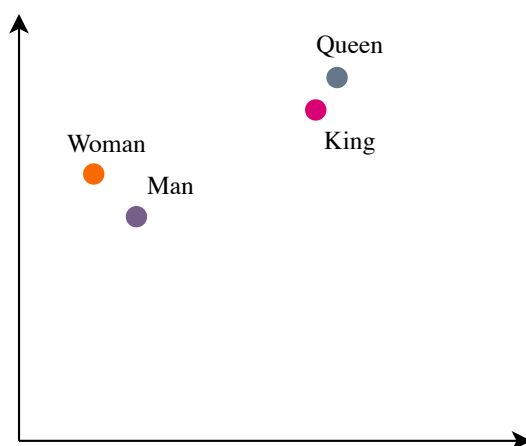


Figure 5 – Example of a representation of word vectors in a vector space, in which semantically similar words have approximated values.

Source: The Author.

3 RELATED WORK

Information extraction (IE) is a widely researched area in response to the exponential growth of unstructured data. Various methods have been proposed to address the challenge of extracting meaningful and structured data from unstructured sources. This section explores several existing works, their approaches, advantages, and limitations while highlighting their relevance to the task of extracting information from legal documents.

3.1 TRADITIONAL INFORMATION EXTRACTION APPROACHES

IE can be traced back to approaches such as pattern matching and rule-based systems. The work of (CHENG; AL., 2009) exemplifies traditional methods, focusing on extracting variables using rule-based approaches. While such methods can yield high precision in structured and predictable data environments, they often struggle with unstructured and semi-structured data, particularly when confronted with noisy or inconsistent formats, as is common in legal documents.

Similarly, (KOWSRIHAWAT; AL., 2015) proposed a framework using regular expressions to extract variables from legal texts. The advantage of this approach is its simplicity and high interpretability. However, its performance is limited to pre-defined patterns and templates, lacking flexibility in dealing with unforeseen data variations and contextualization.

3.2 INFORMATION EXTRACTION IN THE LEGAL DOMAIN

The application of IE in legal texts presents unique challenges due to their length, complexity, and context-specific language. Bhattacharya et al. (BHATTACHARYA; AL., 2019) explored identifying rhetorical roles in Indian legal cases using machine learning methods. This work demonstrated that automated identification of legal roles can improve document comprehension and legal case analysis. Despite its strengths, such an approach is contextually bound to specific legal norms and may not generalize across varying legal frameworks without extensive retraining and adaptation.

An example closer to Brazilian legal scenarios is the work by (PEREIRA; AL., 2024), which integrates LLMs with RAG in one of its many proposals to extract information from audit court documents. This approach allows dynamic and context-aware data retrieval, effectively addressing challenges related to data retrieval accuracy in extensive unstructured texts. However, this integration still suffers from potential issues with scalability and flexibility, as RAG systems are highly data-driven and dependent on the underlying data structure, prompting and chunking configurations used for each task.

3.3 NLP-DRIVEN INFORMATION EXTRACTION

The rise of Natural Language Processing (NLP) has paved the way for complex IE tasks. (HAN; AL., 2023) analyzed and evaluated IE with ChatGPT, emphasizing the limitations of LLMs, such as handling domain-specific data. This evaluation underscores challenges with context precision and output consistency, particularly when LLMs encounter long, complex legal documents. The study highlighted the need for supplementary methods, such as retrieval-augmented generation, to enhance LLM performance.

Moving further, (WEI, X.; AL., 2024) proposed systems where ChatGPT performs IE through zero-shot settings, offering flexibility but potentially sacrificing precision. While useful for quick analyses, such methods require domain-specific adaptation, as seen with complex legal texts.

3.4 CROSS-LANGUAGE INFORMATION EXTRACTION

The study conducted by (BACH; AL., 2019) focused on extracting references from Vietnamese legal documents, employing IE methods to address language-specific challenges. Language structure and syntax differences make cross-language adaptation difficult, as evident when comparing this study to Portuguese legal document processing, as discussed in (VIANNA; AL., 2022). These approaches emphasize the importance of tailoring IE methods to specific languages and legal contexts to ensure accuracy and relevance.

3.5 PROPOSED CONTRIBUTIONS

This work seeks to overcome limitations in traditional IE and NLP-driven approaches by leveraging a local RAG pipeline specifically tailored for Brazilian legal documents. Unlike existing pattern-matching or sole LLM methods, RAG offers flexibility in handling diverse data structures and contextualization of variables without requiring the entire document as input. This flexibility and adaptability are key to improving accuracy and efficiency when extracting critical information from legal documents. Combined with the above, this work is too focused on adding a systematic approach for testing and evaluating parameter configurations, thereby providing a methodology for fine-tuning the extraction configuration. The iterative execution of various combinations of parameters within the RAG framework allows for comprehensive comparisons and enables the identification of optimal settings for extracting different variables from legal documents.

The proposed workflow's systematic parameter optimization, evaluation strategy, and focus on Brazilian legal documents set it apart by targeting the needs of a specific legal corpus. It ensures that extraction processes are tailored to the nuances and structural complexities of legal data, which can vary significantly from other types of datasets. This offers a clearer path for highly contextualized extractions, addressing limitations such as

scalability and adaptability seen in previous methodologies.

Overall, by focusing on both methodological flexibility and specificity to the Brazilian legal domain, this work contributes to a more effective and versatile pipeline that can adapt to complex information extraction needs while maintaining high precision and relevance across diverse legal documents.

Table 1 – Comparative Summary of Related Work (Part 1)

Reference	Objective	Methodology	Dataset
(KOWSRIHAWAT; AL., 2015)	Extract variables using regular expressions in legal texts	Regular expressions	Thai legal documents
(BACH; AL., 2019)	Extract references from Vietnamese legal documents	Pattern matching, NLP techniques	Vietnamese legal documents
(VIANNA; AL., 2022)	Process and summarize Portuguese legal documents	Neural network-based summarization	Portuguese legal corpus
(HAN; AL., 2023)	Analyze IE using ChatGPT	LLM (ChatGPT)	Diverse legal datasets
(PEREIRA; AL., 2024)	Extract data from Brazilian audit court documents using RAG	RAG framework with LLMs	Brazilian audit documents

Table 2 – Comparative Summary of Related Work (Part 2)

Reference	Advantages	Limitations
(KOWSRIHAWAT; AL., 2015)	High performance for simple structures; quick deployment	Difficult adaptation for unstructured or complex patterns
(BACH; AL., 2019)	High precision for specific structured information	Limited scalability to more complex structures
(VIANNA; AL., 2022)	Effective summarization of text; improved access to information	High computational cost; limited explainability
(HAN; AL., 2023)	Zero-shot capabilities; ease of interaction	Performance degradation for complex contexts; hallucinations; No external data
(PEREIRA; AL., 2024)	Combines contextual retrieval with generation for enhanced results	Complexity of tuning RAG parameters

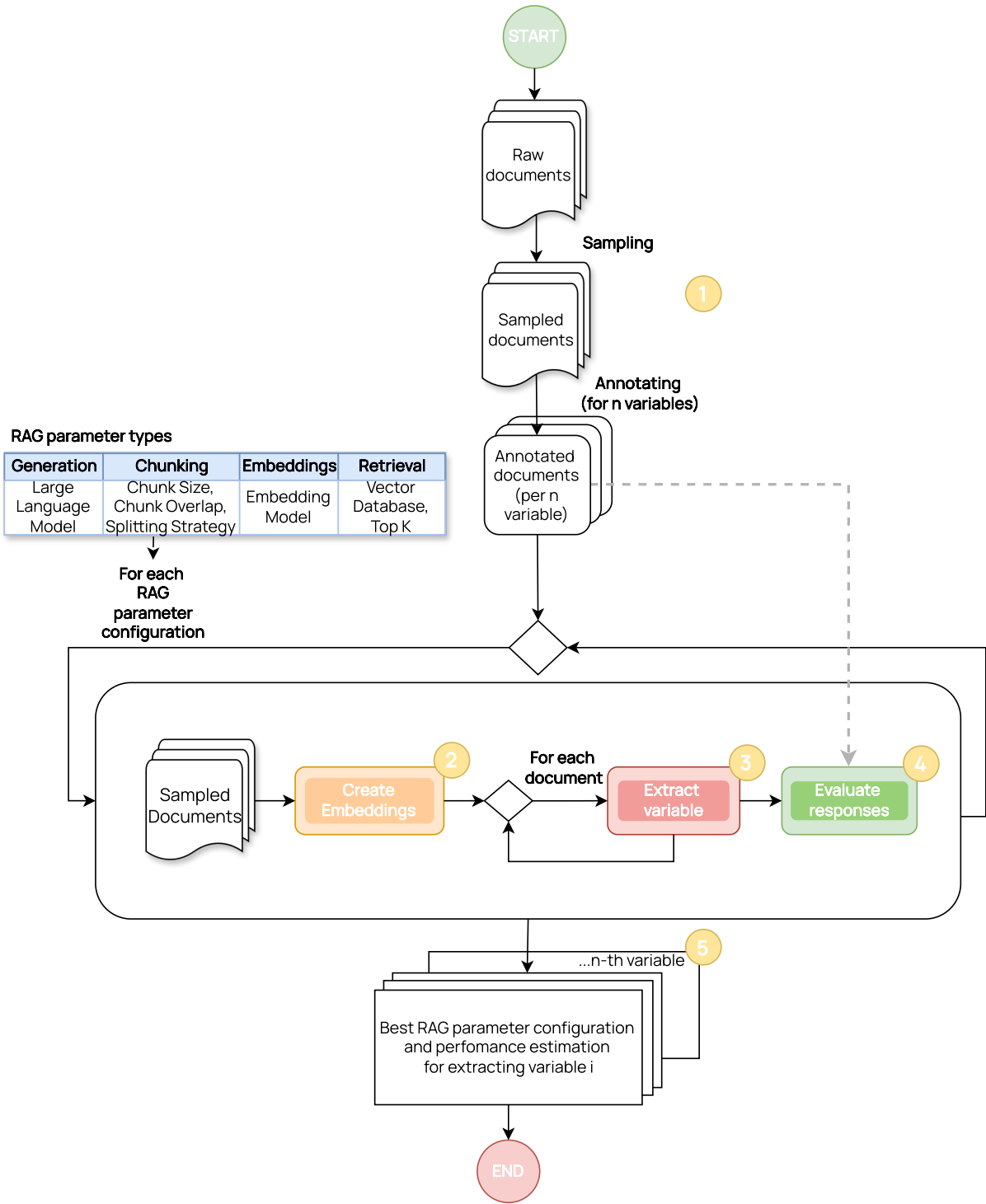
4 WORKFLOW FOR INFORMATION EXTRACTION

This chapter presents our proposed workflow for extracting information from Brazilian legal documents using retrieval-augmented generation (RAG). We developed a structured approach centered around a main RAG pipeline, iteratively executing it across multiple parameter configurations to identify the most effective settings for extracting specific variables. By systematically testing and comparing various parameter sets, our workflow aims to find optimal configurations tailored to each variable, enhancing extraction accuracy and adaptability across diverse document structures.

Figure 6 depicts the overview of the workflow, which will cover all combinations of parameter possible to extract the same chosen variable and compare the results among each other. A parameter configuration is a unique set of values for each of the RAG parameter types listed in Table 3 as a result of iterating through all the available parameter options. The proposal is based on the following steps:

- **Step 1** establishes the beginning of our proposed workflow, initiated by sampling the documents available. We ensure all of them will have an expected value to be extracted for a given query and manually annotate every sampled document with its expected values. This step is the base of our further evaluation assessment step, represented by step number 4.
- **Step 2** iterates all possible parameter configuration combinations within the selected options for each parameter type. Step 2 represents the embeddings creation for every sampled PDF document using the current configuration for chunking and embeddings. These embeddings will be used during the extraction step in the workflow.
- **Step 3** constitutes the main RAG pipeline. It will retrieve the most relevant embeddings generated in the previous step related to a given query. It will insert them as the context in a prompt template and return direct responses containing or not the answer for the task.
- **Step 4** evaluates all the extracted responses by comparing them directly to the foregoing annotated values, labeling as correct the responses that contain the exact expected information for a query.
- **Step 5** outputs the best results parameter configurations and performance estimation for each extracted variable, facilitating decision-making in determining the best parameter settings for a certain variable to be extracted while structuring experimental results.

Figure 6 – Main workflow overview.



Source: The Author.

As introduced previously, Table 3 outlines these parameters, grouping them by types:

- **Generation:** Parameters related to the generation step of RAG. The following parameter can be tested: Large Language Models (LLMs);
- **Chunking:** Parameters related to the documents chunking strategy. The following parameters can be tested: chunk size, which is the size of each of the split chunks of text; chunk overlap, which is the size of text overlap between adjacent chunks; and splitting strategy, which is usually the text splitter used to execute the chunking;
- **Embeddings:** Parameters related to the embeddings to be generated. The following parameter can be tested: embedding model, used to generate the embeddings from the documents' chunks, e.g. BERT models;
- **Retrieval:** Parameters related to the retrieval step of RAG. The following parameters can be tested: vector database, responsible to store and retrieve the embeddings and Top K value, which is the K-amount of retrieved chunks to serve as context on the extraction.

Table 3 – RAG Parameters

Generation	Chunking	Embeddings	Retrieval
Large Language Model	Chunk Size, Chunk Overlap, Splitting Strategy	Embedding Model	Vector database, Top K

Lastly, as illustrated in the workflow overview, a key aspect of this process is the iterative nature of the main pipeline execution. Various parameters are systematically altered through multiple nested for loops based on the previously chosen parameter values to be tested. This approach enables the evaluation of each executed pipeline's effectiveness and identifies the potential most suitable parameter setting for extracting a certain variable from the documents. Finally, it also aids decision-making in selecting from the numerous potential options and adjustments that can be applied to general RAG pipeline parameters by offering structured measured results for each configuration and highlighting the best-obtained ones. In our proposed workflow, any of the previously stated parameters can be iteratively tested and analyzed by determining which options for each parameter type will be covered.

Our proposed workflow differs from existing general RAG tools by offering a structured approach tailored to optimizing parameter configurations for information extraction. While these tools emphasize high-level abstractions and ease of use, our workflow focuses on:

- **Automated Parameter Optimization:** Systematically tests all parameter combinations to identify optimal configurations for specific variables and document types.
- **Evaluation Mechanism:** Incorporates a structured step to compare extracted results with annotated values, providing quantitative insights into accuracy.
- **Domain-Specific Focus:** Tailored to handle long and complex Brazilian legal documents, addressing challenges like variable structures and legal jargon.

By offering these distinct advantages, our workflow provides a more tailored and precise approach to information extraction, distinguishing it from other existing general-purpose RAG frameworks.

5 WORKFLOW EXPERIMENTAL EVALUATION

This chapter will detail our study’s approach with the introduced method to extract two different variables from Brazilian legal documents, covering each of the introduced steps for the proposed workflow. We analyzed and ran our experiments with forty selected Brazilian legal documents provided by MPSC, with an average of 26 pages and 60,000 characters each. The number of documents is due to a reduced amount of initially provided documents and the sampling step detailed in Chapter 4. The documents focus on investigating frauds committed in public procurements across various municipalities within the state of Santa Catarina, Brazil.

Moreover, as mentioned in Chapter 4, while our proposed workflow allows the variation of any of the general RAG parameters, our experiments focused on alternating the parameters listed in Table 4 and maintaining the remaining as fixed parameters as listed in Table 5.

Table 4 – Alternating Parameters

Large Language Model	Chunk size	Chunk Overlap size	Top K
Llama-7b, Llama-13b, Mistral-7B-v0.2	128, 256, 512	20, 50, 100, 200	1, 3, 5, 8, 10, 12

Table 5 – Set Parameters

BERT Model	Splitting Strategy	Vector database
bert-large-portuguese-cased	RecursiveCharacterTextSplitter	Chroma

Additionally, the experiments were conducted on a local server, which utilized a single 24GB NVIDIA RTX 3090 GPU to load the LLMs locally. The following sections detail each step undertaken in our experiments, from data preparation to evaluation, before finally outputting the comparison of the obtained results between all tested parameter configurations.

5.1 DATA PREPARATION

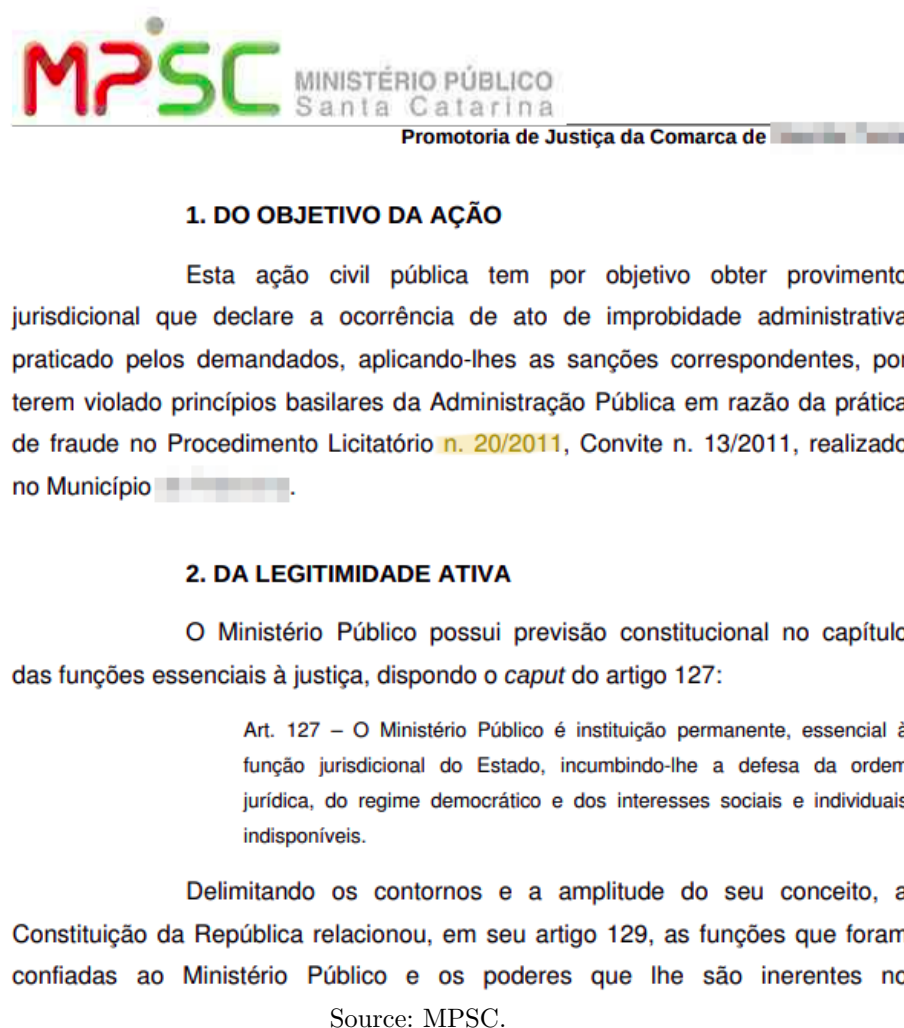
The first step in the experiment, equivalent to step number 1 in Figure 6, is to prepare the available documents dataset to be used. This step is divided into two sub-steps: sampling and annotating.

In the first sub-step, we filtered a smaller sample of the legal documents provided by MPSC, ensuring that all documents contained our study’s analyzed variables. The legal documents, all in PDF format, vary significantly regarding page count and structure of presented information. However, due to filtering, they will all share a common feature: each document alludes to a certain fraud committed in a public procurement process and, therefore, a municipality where the fraud occurred. These two variables form the basis of

our study and will be detailed further in this section. This step will also be particularly useful for assessing the accuracy of each iteration, given the fact that every document will hold an annotated expected value to be compared and evaluated when extracting the analyzed variables.

Figure 7 presents a section of an example document, highlighting the public procurement process identifier, one of our analyzed variables that will be annotated and potentially extracted through the extraction step (step 3 in Figure 6).

Figure 7 – Example of file found in MPSC database and used in our dataset.



Moreover, to evaluate the accuracy of the experiments, we manually annotated each selected document on CSV files, mapping other useful information such as the object of the action, prosecutor's name, public procurement identifier, name of the municipality where the fraud was committed, and more. The annotations will be used to directly compare the model's responses to the constructed prompts, thereby assessing the accuracy of each extraction on every experiment.

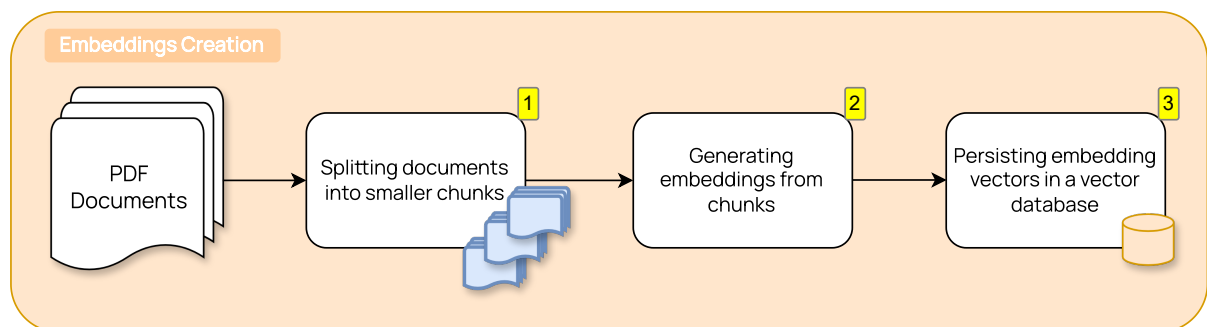
5.2 EMBEDDINGS CREATION

After being separated and annotated, these documents need to be split into smaller chunks of text to represent them as embeddings, representing step 1 in Figure 8. Embeddings are a mathematical representation of text in a vector space, making it possible to navigate and search through them with classic retrieval techniques. To make this possible, we parsed the text on each document with PyPDF. Then, we split these large corpora of texts into smaller chunks using LangChain text splitters, alternating the parameters options indicated in Table 3 under the Chunking column.

Finally, embeddings were generated from the pieces of text parsed and chunked previously using BERTimbau Large (SOUZA; AL., 2020), a BERT model pre-trained in Brazilian Portuguese, representing step 2 in Figure 8. Embeddings capture the semantic meaning of the text and can be used to measure similarity between different pieces of previously split text. It is worth reinforcing that the embeddings were recreated in every iteration, differentiated by the current set of parameters that build them, which directly impacts the measured accuracies across experiments, elaborated in Chapter 6 later.

The embeddings were then stored in a vector database to be manipulated and retrieved accordingly. Vector databases are optimized for storing and retrieving high-dimensional vectors through classic retrieval techniques, such as similarity search, MMR, etc. Chroma¹ was used as our option for vector database, a commonly chosen option for general RAG pipelines, highlighted for being open-source.

Figure 8 – Embeddings creation overview.



Source: The Author.

5.3 EXTRACTING VARIABLES

With the embeddings stored in the database, the next step was to embed the user's query, retrieve the most similar embeddings through a similar search, and finally use them

¹ <https://github.com/chroma-core/chroma>

as context on the prompt final form. These steps correspond to the main RAG pipeline, illustrated by Figure 10.

Figure 9 – Single extraction pipeline overview.

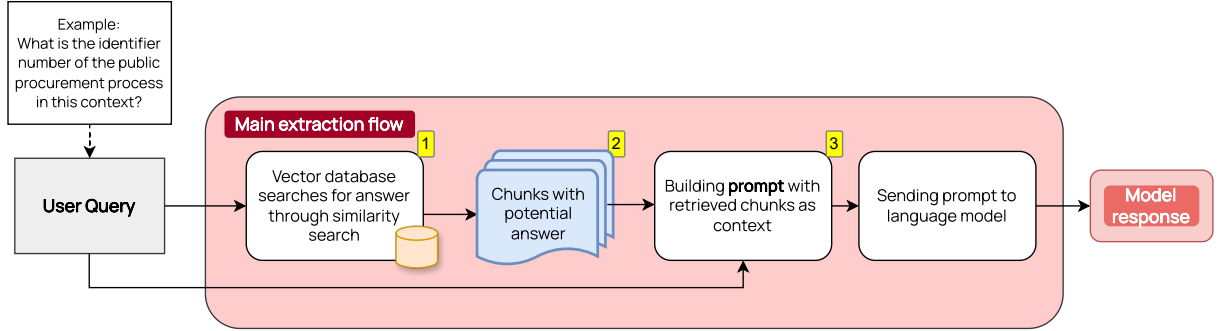


Figure 10 – Single extraction pipeline overview.

Source: The Author.

As introduced in Figure 6, every extraction will take a singular document and retrieve the most relevant embeddings of text through similarity search, comparing it to the embedded user query, step identified by number 1 in Figure 10. Every retrieval only searches through the current document’s embeddings, preventing the possibility of populating the context with information from other documents. The amount of retrieved chunks, illustrated by number 2, depends on the current value of Top K and influences the assertiveness of the retrieval step, whether the retrieved context holds the correct expected answer or not.

Upon obtaining the most relevant pieces of text, they were incorporated into the prompt template shown in Figure 11. Along with the extraction instructions and the initial user query, we form a final version of a zero-shot prompt, as no examples for the task are ever provided to the model, represented by step 3.

The LLMs options used in our experiments were the Llama-2 family chat models (except 70B) (TOUVRON; AL., 2023) and Mistral-7B-Instruct (JIANG; AL., 2023), all of them loaded locally on the previously mentioned server. This setup ensured privacy to handle the legal document’s sensitive information, however, limited the involved models used in our study, making it impossible to handle bigger models on the current analysis (e.g. Llama-2-70b, Mixtral-8x7B, etc).

5.4 EVALUATION METRICS

While several aspects of evaluation around RAG can be measured (GAO; AL., 2024), our work primarily concentrates on direct accuracy assessment. We specifically examine whether the generated response by the model precisely matches the annotated value associated with a particular document. This evaluation occurs in step 4 in Figure 6,

Figure 11 – English translation of prompt template used in every experiment.

```

## Instructions

You are a helpful AI assistant and provide the response in Portuguese to the question based on the provided context.

Use the following chunks of context to answer the question at the end. If it is not possible to answer the question from the
context, just answer that you didn't find the answer.

## Context built with Top K relevant retrieved chunks

CONTEXT: [RETRIEVED CHUNKS]

>>>QUESTION<<<: [USER QUERY]

>>>ANSWER<<<:

```

Source: The Author.

which will divide the quantity of successfully matched extracted values by its annotation value by the total amount of documents. In addition to evaluating the primary accuracy, we also measure the maximum LLM accuracy, labeled as successful every time the retriever successfully returns a segment containing the expected answer, step presented with number 1 in Figure 10. This metric proves insightful during the analysis of primary accuracy, as an incorrect response could not have been correct if the retriever had not returned a piece of context that contains the answer previously. This scenario may occur if the user-built query is not adequately optimized to retrieve the correct answer consistently.

5.5 EVALUATED EXTRACTED VARIABLES

As previously stated, our work focuses on extracting different variables: public procurement process identifiers and municipalities of irregularity. The following sections will detail each one and their associated experiments.

5.5.1 Public Procurement Process Identifier

The public procurement process identifier is the first variable examined in our study. It is a string that identifies a certain public procurement process for a municipality, and it is consistently presented in the format X/YYYY, where 'X' represents any numerical sequence and 'YYYY' denotes a four-digit year, as shown highlighted previously in Figure 7.

Specifically, this variable has 145 associated experiments, one for every unique configuration possible when interchanging the parameters listed in Table 4 (with some exceptions, for instance, chunk overlap can't be bigger than chunk size, Top K multiplied

by size of chunk can't overcome the maximum amount of tokens inputted on the prompt for the LLMs and hardware limitations). The English-translated prompt used in our RAG pipeline for extracting this variable is demonstrated below, with the appropriate description and formatting of the variable.

Given the following description: The public procurement process number (or Public Procurement number, Public Procurement No.) refers to the unique identification assigned to each public procurement process, usually in the format 'number/year'.

What is the number of the public procurement process mentioned in the context above?

ANSWER ONLY THE PROCESS NUMBER.

5.5.2 Municipality Of Irregularity

The municipality of irregularity is the second and final analyzed variable. It consists of the name of the municipality where fraud was committed through public procurement processes. All municipalities are within the state of MPSC, Brazil.

Likewise, this variable has 145 associated experiments, with the same approach mentioned above for public procurement identifiers, adapting the prompt to appropriate instruction and contextualization. The English-translated prompt used in our RAG pipeline for extracting this variable is demonstrated below, with the proper description of what constitutes the variable.

Given the following description: The municipality of irregularity is generally the municipality where a certain irregularity, fraud, or crime was committed.

What is the municipality of irregularity mentioned in the context above?

ANSWER ONLY THE NAME OF THE MUNICIPALITY, WITHOUT APOSTROPHES.

6 WORKFLOW RESULTS AND DISCUSSION

This chapter presents the results of our study on extracting information from Brazilian legal documents using our proposed workflow. We analyze the performance of our information extraction techniques and compare them with an existing extraction method, pattern matching utilizing regular expressions.

Regular expressions have a downside that our pipeline will suffice: contextualization. The dealt legal documents cite other information in the same format, such as the public procurement modality identifier, contract identifiers, and other formalities. As for municipalities, these documents usually also cite other different municipalities, creating ambiguity for matching these patterns. As a result, regular expressions fetch multiple different matches on these scenarios and are not be able to choose the correct one among all of them, whereas LLMs and retrievers with the appropriate contextualization on the prompt will be more likely to find the correct match.

In a comparison analysis, we built a regular expression that looked for matches using the mentioned format. When comparing the mode of the matches, extracting the variable with a regular expression reached a maximum of 35% accuracy against the best accuracy of 88% using our workflow. We also built a comparative regular expression for extracting municipalities and evaluated it with our annotated values. We achieved a maximum accuracy of 82.5% compared with our best accuracy of 93% using our proposed workflow. This comparison is visible in Figure 12, where the best obtained accuracies through our proposed method overcomes expressively the result obtained by the regular expression, when extracting public procurement process identifiers. For extracting municipalities, our experiments still bests the regular expression results by 10.5%. These results underscore the effectiveness of our method, overcoming regular expressions by contextualizing the variables on the prompt fed to the LLMs. The built regular expressions are detailed below.

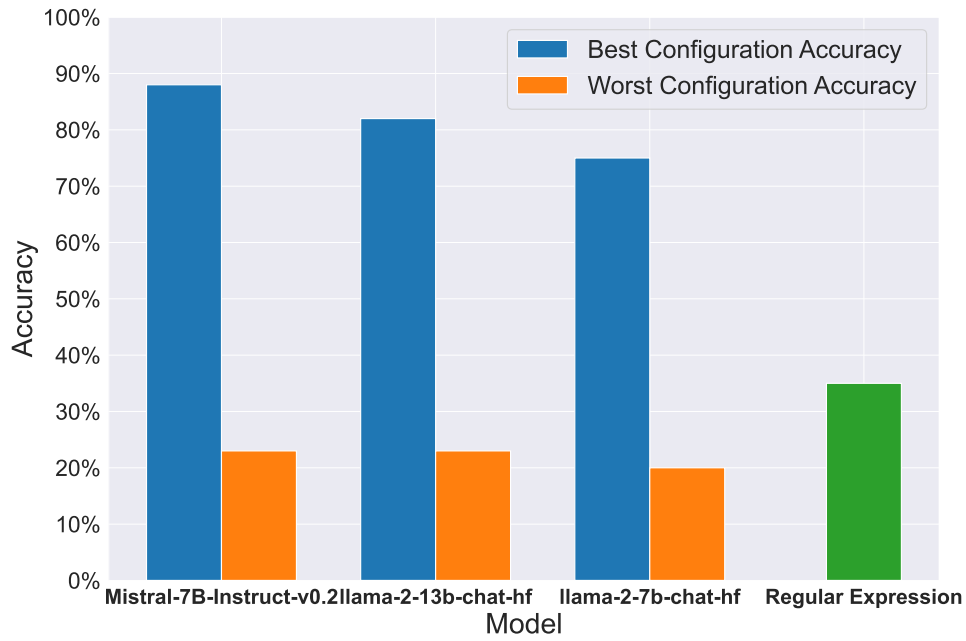
Public Procurement Process Identifier

`\b\d+/\d{4}\b`

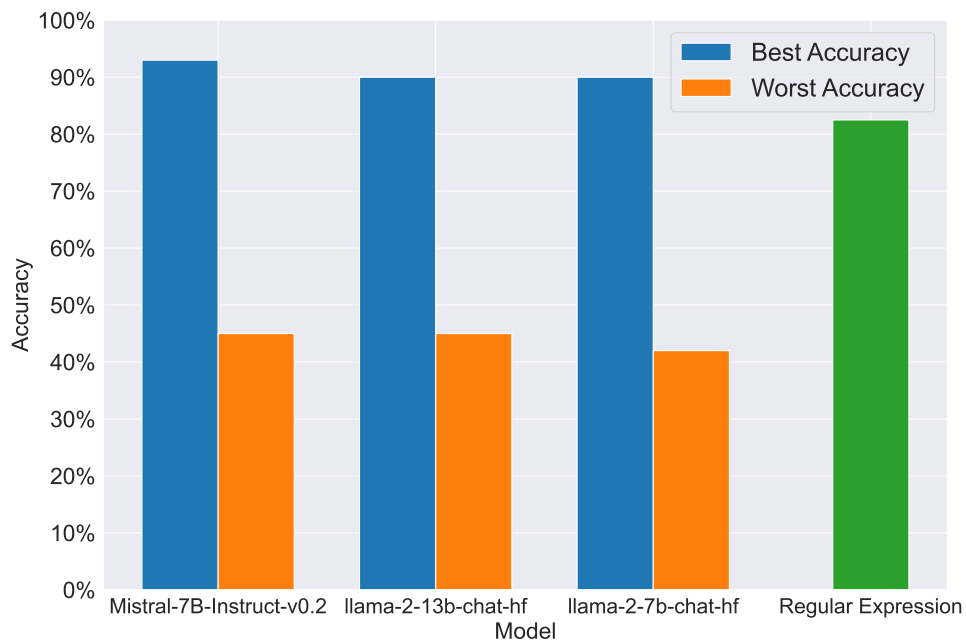
Municipality of Irregularity

`Município de ([A-Z][a-z]+(?:\s[A-Z][a-z]+)*)`

Furthermore, another important comparison is displayed in Figure 13, introducing the three best accuracy ranked parameter configurations for each explored LLM, which too proposes that smaller chunks are more suitable for extracting both public procurement process identifiers and municipalities of irregularity in RAG pipelines. It also highlights



(a) Public Procurement Process Identifier.

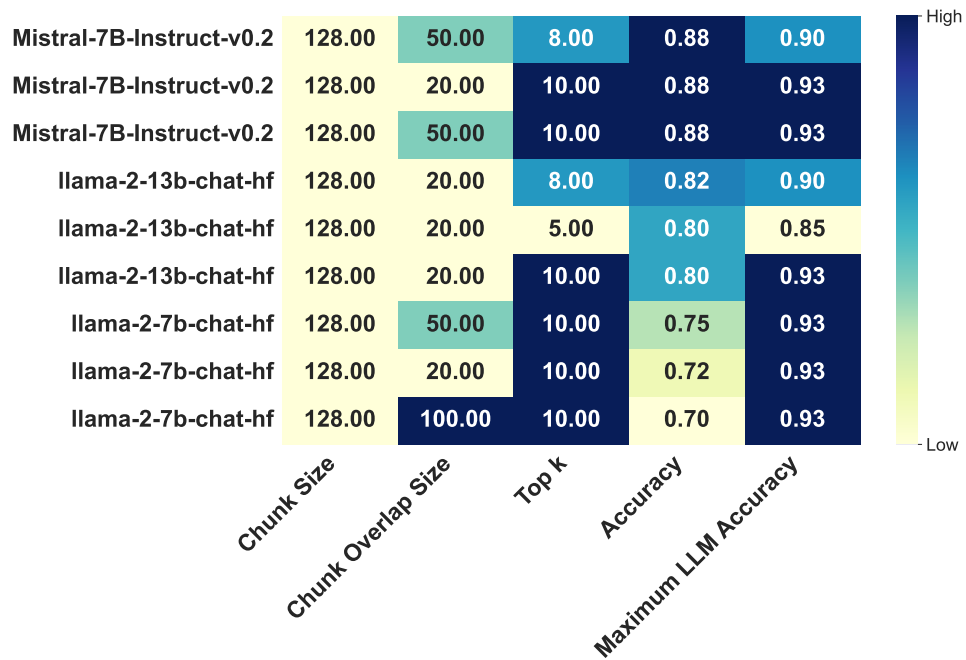


(b) Municipality of Irregularity

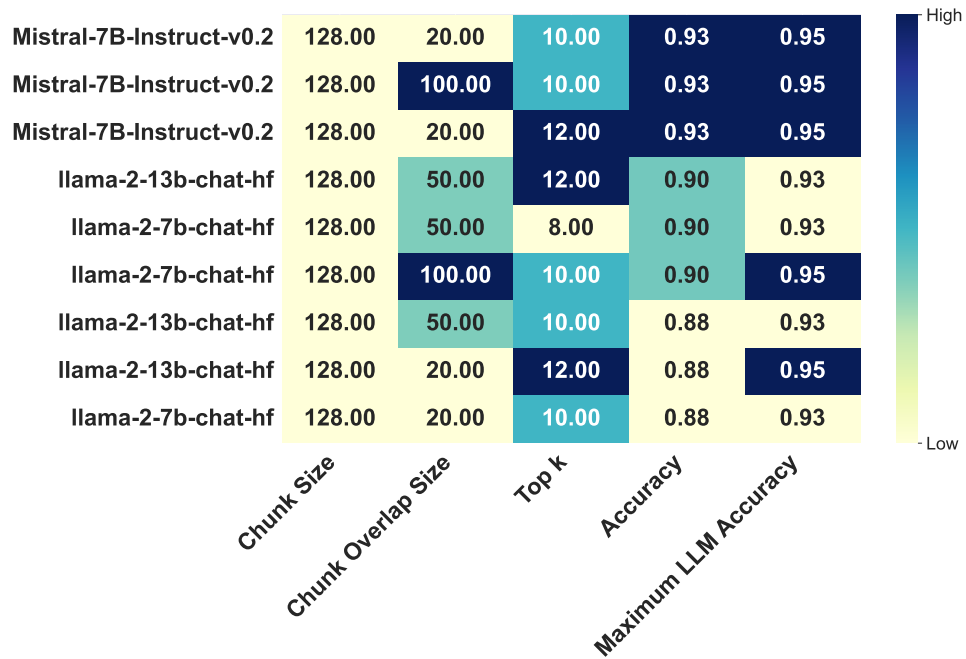
Figure 12 – Best and Worst results per model vs Regular Expression

Source: The Author.

how Mistral's 7b model, even though has half of the parameters, practically outperforms Llama-2-13b model, reaching the highest obtained accuracy of 88% and 93% of extracted answers. Then, Figure 14 illustrates the Top K evolution and its impact on obtained accuracies on extracting both variables on fixed chunk sizes and chunk overlap. It suggests that the increase of Top k values directly impacts on the accuracy in pipelines with small chunks, increasing the probability of the retriever returning the accurate answer among the available embeddings, step corresponding as number 1 in Figure 10.



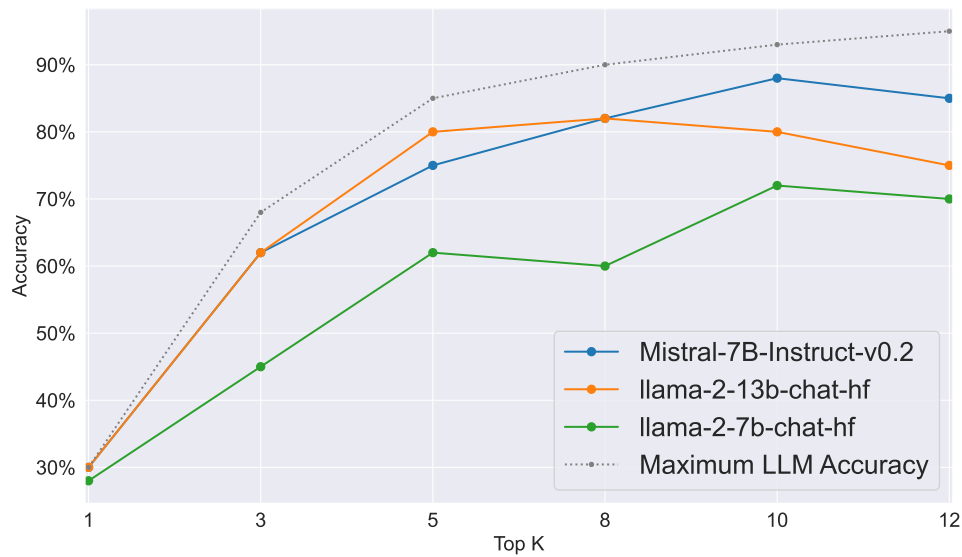
(a) Public Procurement Process Id.



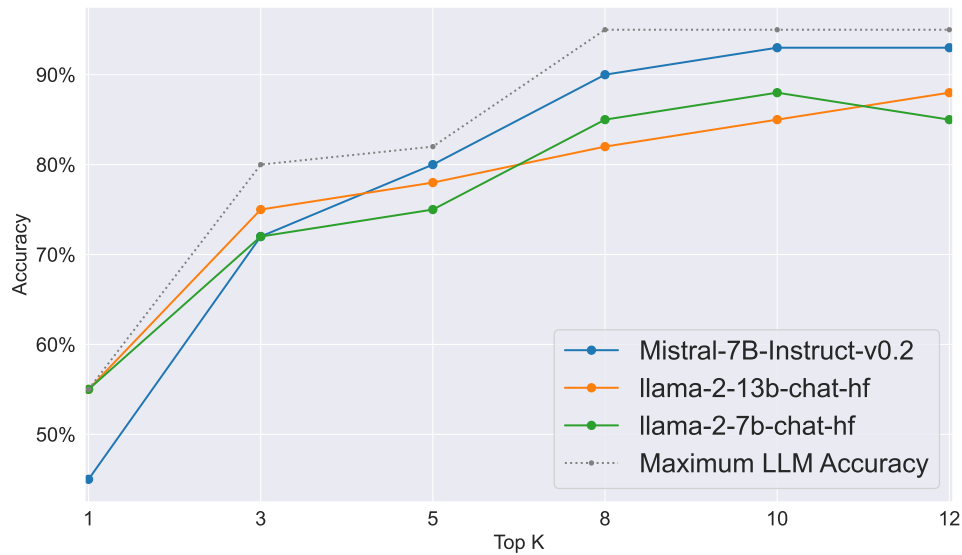
(b) Municipality of Irregularity

Figure 13 – Top three best results configurations per LLM.

Source: The Author.



(a) Public Procurement Process Identifier



(b) Municipality of Irregularity

Figure 14 – Top K evolution with fixed chunk size as 128 and chunk overlap as 20.

Source: The Author.

Mistral’s model outperforming LLama-2 models can also be seen in Figure 15, where it is present in 27 out of the top 50 accuracy results extracting public procurement process identifiers.

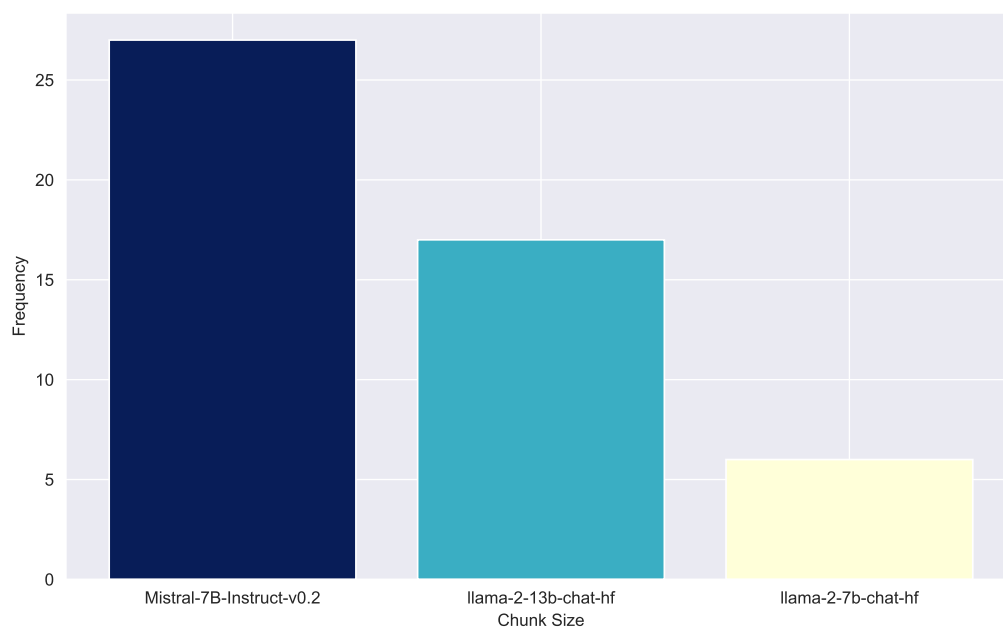


Figure 15 – Models frequency in top 50 experiments extracting public procurement process identifiers.

Source: The Author.

6.1 EXTENDED RESULTS WITH RECENT MODELS

To complement the primary analysis, we evaluated newer language models to extract the same variables within the same experiment scenarios as described above: Llama 3.1 70b (DUBEY et al., 2024) and Qwen 2.5 72b (YANG et al., 2024). These models showed improvements in accuracy, as summarized in Table 6, more specifically when extracting public procurement identifiers. For instance, Qwen 2.5 72b achieved an expressive 93% top accuracy for public procurement identifiers, the best obtained accuracy across every model.

Table 6 – Comparison of best obtained accuracies across new and previous models.

Model	Public Procurement Identifiers (%)	Municipality of Irregularity (%)
Qwen 2.5 72b	93	88
Llama 3.1 70b	88	90
Mistral-7b-v2	88	93
Llama-2-13b	88	90
Llama-2-7b	75	90

The extended evaluation with recently released models revealed that improvements in accuracy were selective and task-specific. For extracting public procurement process identifiers, the Qwen 2.5 72b model demonstrated a significant increase in accuracy, outperforming the previously evaluated models by 5 percentage points. On the other hand, for the task of extracting municipalities of irregularity, no improvements were observed when comparing to the previous results, with accuracy remaining consistent across the newly evaluated models. These findings highlight the importance of aligning model capabilities with task requirements and suggest that future improvements in information extraction pipelines may rely on complementary strategies, such as more tailored prompt engineering or fine-tuning, to fully exploit the potential of newer models.

In conclusion, these results highlight the flexibility and adaptability of the proposed workflow. Its design allowed for the easy inclusion of newer models released later in the research, enabling updates to the analysis. This demonstrates its effectiveness as a tool for iteratively refining performance as models and parameters evolve.

7 INTERFACE

This chapter details the design and functionality of the interface developed to streamline information extraction using Retrieval-Augmented Generation (RAG) for complex documents. The primary objective of the interface is to facilitate an efficient and structured workflow for users to upload documents, define extraction variables, annotate documents, configure RAG parameters, and review the results. The interface is designed to support users through five key stages: *Uploading Documents*, *Creating Variables*, *Annotating Documents*, *Configuring RAG Parameters*, and *Extracting Results*. Each step is designed with user experience in mind, ensuring that users, both with and without technical background in RAG, can easily navigate the process and achieve accurate and reliable information extraction outcomes.

7.1 UPLOADING DOCUMENTS

The first step in the workflow is to upload the documents in which will have information extracted. The interface supports drag-and-drop functionality for adding files, specifically `.pdf` documents, which are the accepted format for processing (Figure 16). This choice ensures compatibility with the information extraction algorithms used later in the workflow. Users can add multiple documents at once, and the interface provides a visual list of the uploaded files, including a preview button for each document. Feedback mechanisms inform users of the file types allowed and whether any upload errors occurred, including colored alerts with its corresponding error or success alert messages.

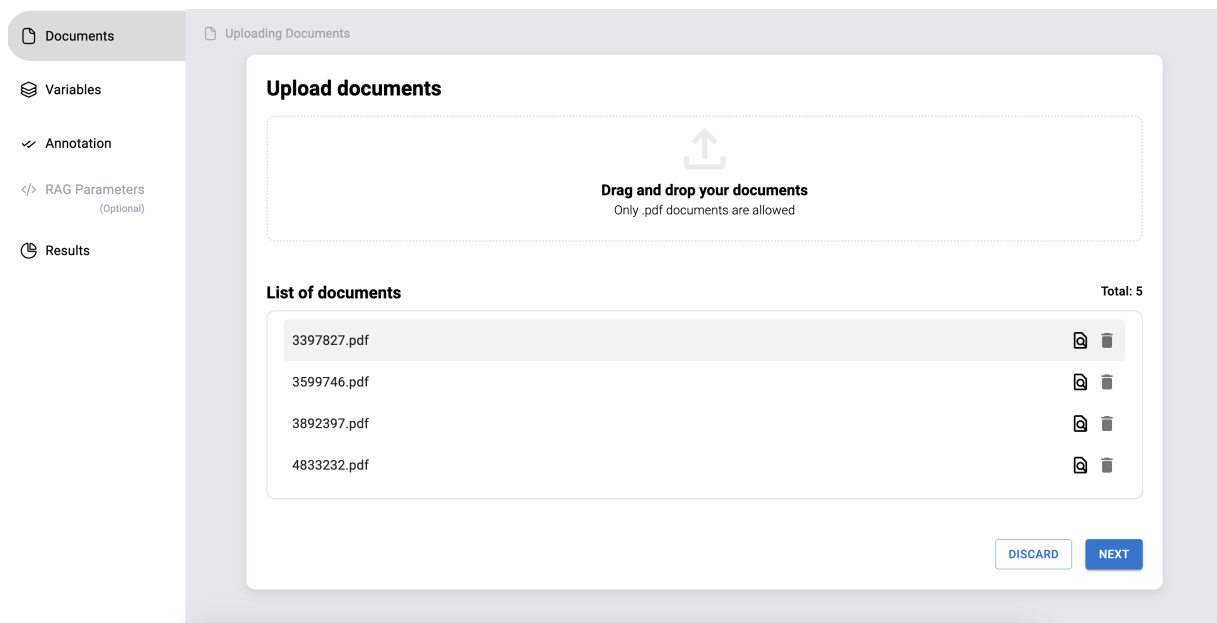


Figure 16 – The document upload interface, allowing drag-and-drop functionality for uploading `.pdf` documents.

Source: The Author.

7.2 CREATING VARIABLES

Once documents have been uploaded, the next step involves creating variables. Variables are used to specify the information users wish to extract from the documents, and the interface allows users to define variables with a name, a prompt label and a custom prompt.

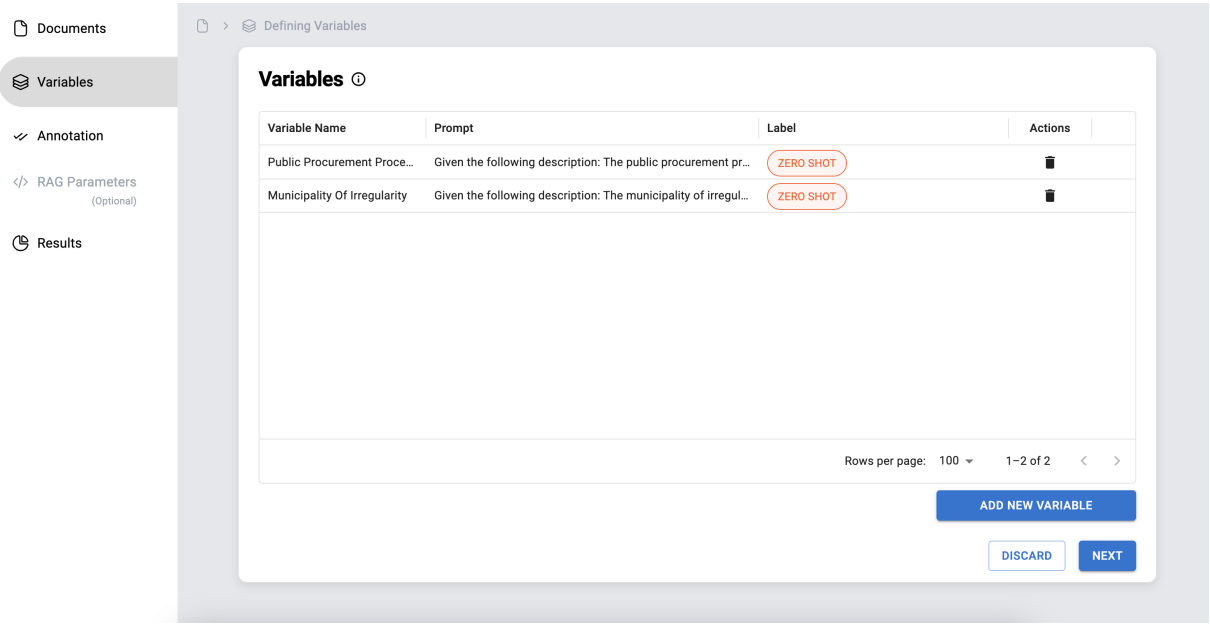


Figure 17 – The variable creation interface, allowing users to create variables with custom prompts and labels.

Source: The Author.

7.3 ANNOTATING DOCUMENTS

After defining the variables to be extracted, the user proceeds to the annotation step. In this phase, documents are listed with annotation text inputs that enable users to annotate each document according to the variables created.

Annotations can be made across all uploaded documents or limited to a subset, depending on the user's requirements. The annotation process helps users gain an understanding of the document structure, potentially identifying patterns that can improve information extraction performance, in addition as well as serve as the base of the evaluation process, as displayed in Figure 6. The interface includes an icon that, when clicked, opens a detailed view of each PDF. This allows users to search and analyze each document within the application. Additionally, documents with completed annotations are highlighted with their names displayed in green.

List of documents		Total: 5
3397827.pdf	<input type="text" value="Fill with correct value..."/>	
3599746.pdf	<input type="text" value="18/2023"/>	
3892397.pdf	<input type="text" value="Fill with correct value..."/>	
4833232.pdf	<input type="text" value="11/2015"/>	
5242773.pdf	<input type="text" value="Fill with correct value..."/>	

Figure 18 – The annotation interface, allowing users to annotate subset of documents for each created variable.

Source: The Author.

7.4 CONFIGURING RAG PARAMETERS

The fourth step in the interface workflow is configuring the RAG parameters. This optional step allows users to customize the RAG workflow by selecting one or more configuration options for each parameter. These parameters are the same as those listed in Table 3 in Chapter 4.

For users with a technical background in RAG, this step offers the flexibility to fine-tune the workflow's behavior by experimenting with different parameter configurations. However, for those without technical expertise, the interface provides default settings that are designed to work effectively without any additional customization. This ensures that users can proceed with the information extraction process smoothly, even if they prefer not to engage with this step. Advanced users can compare the results of different configurations to further optimize the extraction process, but it remains entirely optional for others.

The screenshot shows a web application interface for configuring RAG parameters. On the left is a sidebar with navigation links: Documents, Variables, Annotation, RAG Parameters (Optional) (highlighted), and Results. The main content area is titled 'Choosing Parameters' and contains a 'RAG Parameters Customization (Optional)' panel. This panel is organized into three sections: Models, Embeddings, and Retrieval. The Models section has an LLM dropdown set to 'llama-2-7b, llama3.2'. The Embeddings section includes dropdowns for Chunk Size (128, 256), Chunk Overlap (20, 50, 100), Text Splitter (RecursiveCharacter...), Vector Database (Chroma), Bert Model (bert-large-portugue...), and Embedding Model (HuggingFaceBgeE...). The Retrieval section has dropdowns for Top K (1, 3, 5, 10), Chain Type (stuff), and a Device Map input field set to 'mps'. An 'EXTRACT' button is located at the bottom right of the customization panel.

Figure 19 – The parameter customization interface, allowing users to configure the workflow parameters.

Source: The Author.

7.5 EXTRACTING RESULTS

The final step in the workflow is extracting and reviewing the results. Once the documents have been annotated and RAG parameters configured, the interface runs the information extraction process and displays the results in a tabular format. Each extracted value is linked to its corresponding document and variable, allowing users to trace back the source and verify the accuracy of the extracted information.

As presented in Figure 20, results are compressed and made available to download on the *Result File* column. Upon download, it is possible to obtain two sets of files:

- the results summary, listing every combination of parameter result on each line, and;
- the detailed outputs evaluation for each variable, comparing the expected value versus the extracted value.

Extraction ID	Status	Started On	Completed On	Result File
894	COMPLETED	Sun, 17 Nov 2024 18:03:47 GMT	Sun, 17 Nov 2024 18:08:01 GMT	↓
893	COMPLETED	Sun, 17 Nov 2024 18:03:09 GMT	Sun, 17 Nov 2024 18:03:26 GMT	↓
892	IN_PROGRESS	Sun, 17 Nov 2024 17:59:25 GMT	N/A	
891	IN_PROGRESS	Sun, 17 Nov 2024 17:58:10 GMT	N/A	
890	IN_PROGRESS	Sun, 10 Nov 2024 23:02:17 GMT	N/A	

Rows per page: 100 1-5 of 5

Figure 20 – The extraction results interface.

Source: The Author.

8 EXTENDED RESULTS

In this chapter, we present results obtained from applying our proposed workflow and interface on an alternative dataset. This additional test serves to validate the adaptability of the interface combined with the workflow for extracting information from varied sources, beyond the dataset used in initial evaluations. By evaluating our approach with a different dataset, we aim to further confirm the efficacy of our system in handling diverse document structures and content.

8.1 EXPERIMENT SETUP

For this experiment, we selected a dataset distinct from the one used in previous chapters to assess the applicability of our interface. This new dataset contains multiple civil non-prosecution agreement (NPA) documents, provided as well by the MPSC. These documents detail the terms, conditions, and obligations agreed upon between the involved parties to resolve civil disputes without pursuing litigation. Specifically, these agreements outline the commitments and concessions of individuals or entities to comply with certain stipulated actions, penalties, or remedies, thereby avoiding further legal proceedings initiated by the MPSC.

To prepare this dataset, we followed similar preprocessing steps as in the main workflow, including sampling 100 documents and annotating a set of selected variables for each sampled document, such as the value of compensatory fine and type of legal procedure.

In this evaluation, fewer testing scenarios were considered compared to the primary dataset experiments. This decision was driven by the primary goal of this extended test: to obtain results from a new dataset while emphasizing the functionality and adaptability of the developed interface.

Eight new experiments were conducted during the extended evaluation to assess the impact of different retrieval configurations on our system’s performance. In every experiment, it was utilized a chunk size of 128 and a chunk overlap of 20, the combination that obtained the best results in the previous dataset. The experiments were then run varying the top-k value with two models, Llama 3.1 70b (DUBEY et al., 2024) and Qwen 2.5 72b (YANG et al., 2024). The remainder of unmentioned parameters remain the same as the previous experiments

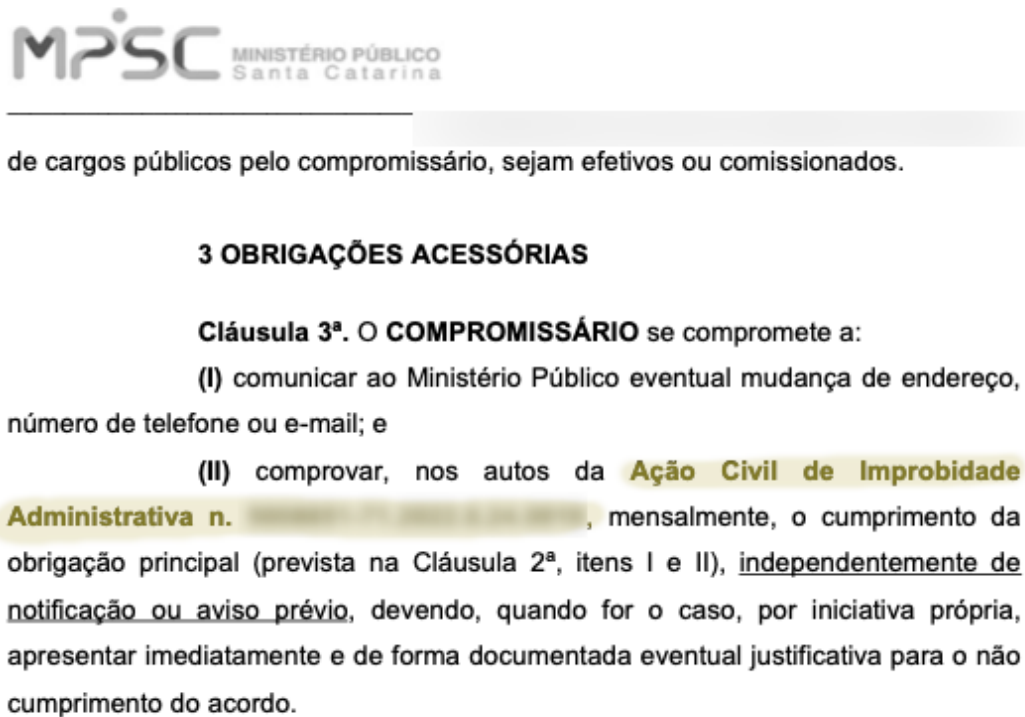
8.2 EXTRACTION RESULTS

The extraction workflow was then applied to the new dataset, targeting a new variable: type of legal procedure conducted within a certain NPA. Some examples are Civil Inquiry, Civil Action, Sentence Enforcement and others.

8.2.1 Type of legal procedure

In extracting the type of legal procedure, our workflow achieved a best accuracy of 84%, which is slightly lower than the original dataset's performance. This variable was generally well-suited to our method, and Figure 21 exemplifies an example of a type of legal procedure to be extracted within a certain document. The result summary file obtained is presented in Table 7 and the outputs for the best scenario is presented in Figure 22.

Figure 21 – Example of NPA document found in MPSC database and used in our dataset, with an example of the chosen variable highlighted.



Name	Chunk Size	Chunk Overlap Size	Top k	Result
llama3.1:70b	128	20	1	48%
llama3.1:70b	128	20	3	70%
llama3.1:70b	128	20	5	73%
llama3.1:70b	128	20	10	61%
qwen2.5:72b	128	20	1	55%
qwen2.5:72b	128	20	3	84%
qwen2.5:72b	128	20	5	84%
qwen2.5:72b	128	20	10	52%

Table 7 – Columns of the results summary file, obtained from extracting the proposed variable for each experiment scenario.

Figure 22 – Partial output evaluation detail on extracting the desired variable.

EVALUATION SUMMARY				
Document	Type of legal procedure			
	Expected Value	Extracted Value	Result	Did retriever found the answer?
	Inquérito Civil	Inquérito Civil	SUCCESS	YES
	Inquérito Civil	Inquérito Civil	SUCCESS	YES
	Inquérito Civil	Inquérito Civil	SUCCESS	YES
	Inquérito Civil	Inquérito Civil	SUCCESS	YES
	Inquérito Civil	Inquérito Civil	SUCCESS	YES
	Ação Civil Pública	Ação Civil de Improbidade Administrativa	ERROR	NO
	Inquérito Civil	Inquérito Civil	SUCCESS	YES
	Ação Civil de Improbidade Administrativa	Ação Civil Pública pela Prática de Ato de Improbidade Administrativa	ERROR	NO
	Ação Civil Pública	Ação Civil de Improbidade Administrativa	ERROR	NO
	Inquérito Civil	Inquérito Civil	SUCCESS	YES
	Ação Civil Pública	Inquérito Civil	ERROR	NO
	Inquérito Civil	Inquérito Civil	SUCCESS	YES
	Ação Civil Pública	Ação Civil Pública	SUCCESS	YES
	Inquérito Civil	Inquérito Civil	SUCCESS	YES
	Ação Civil de Improbidade Administrativa	Ação Civil Pública	ERROR	NO
	Inquérito Civil	Inquérito Civil	SUCCESS	YES
	Procedimento Preparatório	Sindicância	ERROR	NO
	Inquérito Civil	Inquérito Civil	SUCCESS	YES
	Inquérito Civil	Inquérito Civil	SUCCESS	YES
	Inquérito Civil	Inquérito Civil	SUCCESS	YES
	Cumprimento de sentença	Cumprimento de Sentença	SUCCESS	YES
	Ação Civil de Improbidade Administrativa	Inquérito Civil	ERROR	NO
	Inquérito Civil	Inquérito Civil	SUCCESS	YES
	Inquérito Civil	Inquérito Civil	SUCCESS	YES
	Inquérito Civil	Inquérito Civil	SUCCESS	YES
	Inquérito Civil	Inquérito Civil	SUCCESS	YES
	Procedimento Preparatório	Ação Civil Pública	ERROR	NO
	Ação Civil de Improbidade Administrativa	Ação Civil Pública de Responsabilidade por Ato de Improbidade Administrativa	ERROR	NO
	Ação Civil de Improbidade Administrativa	Ação Civil Pública	ERROR	NO
	Ação Civil Pública	Ação Civil Pública	SUCCESS	YES
	Ação Civil de Improbidade Administrativa	Ação Civil por Ato de Improbidade Administrativa	ERROR	NO

Finally, the English-translated prompt used in our interface and sent to the RAG pipeline for extracting this variable is demonstrated below, with the appropriate description and formatting of the variable.

Given the following description: Legal procedure types refer to different categories of processes or investigations mentioned in civil non-prosecution agreement documents, usually listed alongside an identifying number.

Examples of legal procedures:

- * *Civil Inquiry*
- * *Public Civil Action*
- * *Civil Action for Administrative Misconduct*
- * *Sentence Enforcement*
- * *Preparatory Procedure*

Instructions:

- * *If the document contains 'Civil Inquiry n.', respond with 'Civil Inquiry'.*
- * *If the document contains 'Public Civil Action n.', respond with 'Public Civil Action'.*
- * *If the document contains 'Public Civil Action for the Practice of an Act of Administrative Misconduct n.', respond with 'Civil Action for Administrative'.*

Misconduct’.

** If the document contains ‘Sentence Enforcement n.’, respond with ‘Sentence Enforcement’.*

** If the document contains ‘Preparatory Procedure n.’, respond with ‘Preparatory Procedure’.*

What is the type of legal procedure mentioned in the context above? ANSWER ONLY THE LEGAL PROCEDURE.

8.3 DISCUSSION

The application of our workflow and interface on a different dataset provides insights into the flexibility and limitations of our approach. The consistent performance across datasets suggests that our RAG-based extraction method can handle a variety of document structures, adapting well to differences in content format. This adaptability supports the regularity and flexibility of our method and its potential application to other domains within the legal field and beyond.

On the other hand, the slight decrease in accuracy observed with the alternative dataset underscores the importance of further attention and refining the multiple aspects involved in the workflow, such as the retrieval parameters and prompt fine-tuning. These results highlight that the effectiveness of our extraction process depends on the variable, its content, and the structure of the document being analyzed. The challenges faced in extracting this variable suggest that certain types of information may inherently pose more difficulties for our method, underscoring the need for additional strategies or targeted improvements. Despite these limitations, the workflow’s ability to reveal variables and configurations with lower extraction performance is valuable, as it helps identify which parameters and prompts require further refinement and enhanced approaches.

It is also possible to observe that while increasing the top k parameter initially improved retrieval performance, particularly for configurations with smaller values of top k, excessive increments led to decreased accuracy. This trend suggests that including too many retrieved results can introduce noise, making it harder to discern relevant information and increasing the likelihood of errors. Thus, determining an optimal range for top k is crucial to maintaining precision without overwhelming the retrieval mechanism with irrelevant data.

Furthermore, it is important to acknowledge that in complex documents, variations of the same term or concept can lead to confusion for language models (LLMs). LLMs, although powerful, may generate slightly different answers based on the context in which a term is used. This is a common challenge in legal documents, where the same concept might be referred to in multiple ways depending on the context, structure, or language nuances.

These variations, while sometimes leading to different outputs, can still be considered correct within the given context, highlighting the inherent flexibility of LLMs in adapting to the diverse nature of complex documents. The variations can be noticed in some of the failed evaluations produced in the best experiment case, highlighted in Figure 21.

9 CONCLUSION AND FUTURE WORK

In conclusion, legal documents are often extensive and irregularly structured, presenting significant challenges for extracting relevant and structured data. In this work, we presented and evaluated a promising approach utilizing retrieval-augmented generation (RAG) to extract variables of interest from legal agreements. Our primary experiments achieved an average accuracy of 90%, surpassing traditional pattern-matching techniques in contextualization, a key bottleneck in legal information extraction (IE). Additionally, our zero-shot IE paradigm required no training or fine-tuning, marking a significant step forward in information extraction within the legal domain.

The extended results obtained by applying our workflow to an alternative dataset of civil non-prosecution agreements (NPAs) further validated the flexibility and adaptability of our approach. Despite encountering a slightly lower best accuracy of 84% when extracting the type of legal procedure variable, the system demonstrated resilience and efficacy in adapting to new data structures. This underscores the importance of our method’s adaptability to diverse content while revealing areas for further optimization, such as fine-tuning retrieval parameters and refining prompts to suit distinct datasets.

9.1 FUTURE WORK

To further enhance the proposed solution, future work will prioritize key interface improvements. Simplifying user interactions and creating intuitive controls will ensure broader usability and accessibility for both legal practitioners and non-technical users. This will involve streamlining data input methods, refining results visualization, and incorporating user feedback mechanisms to continuously evolve the system’s ease of use.

Additionally, enhancing the RAG system through the integration of domain-specific knowledge graphs presents a promising avenue. By structuring legal knowledge and contextual relationships, knowledge graphs can further improve contextual relevance and the accuracy of extracted data. This enhancement would enable better entity disambiguation, particularly within complex legal documents that feature intertwined references, helping to narrow the gap between AI-based systems and human expertise.

Our results have demonstrated significant gains in extraction accuracy through contextualized prompts, achieving up to 88% for public procurement identifiers and 93% for municipal irregularity mentions in primary experiments, and 84% for extracting types of legal procedures from NPAs in extended limited evaluations in order to validate the proposed interface. These outcomes highlight the adaptability and utility of our approach while indicating opportunities for optimization through iterative testing, validation with more diverse datasets, and fine-tuning of workflow parameters. We will also explore additional large language model (LLM) configurations and a broader variety of legal documents to extend the solution’s applicability across legal domains.

Finally, formalizing the proposed workflow as a modular RAG parameter evaluator framework is a key next step. This framework will enable reproducibility and adaptation for a wide range of tasks beyond legal documents, ensuring a flexible and scalable solution that can be tailored to different domains and user needs.

REFERENCES

- BACH; AL., et. Reference Extraction from Vietnamese Legal Documents. In: (SoICT '19), p. 486–493.
- BHATTACHARYA, Paheli; AL., et. **Identification of Rhetorical Roles of Sentences in Indian Legal Judgments**. [S.l.: s.n.], 2019. arXiv: [1911.05405 \[cs.IR\]](#).
- BROWN, Tom B. Language models are few-shot learners. **arXiv preprint arXiv:2005.14165**, 2020.
- CHENG; AL., et. Information extraction from legal documents. In: 2009 Eighth International Symposium on Natural Language Processing. [S.l.: s.n.], 2009.
- DEVLIN, Jacob. Bert: Pre-training of deep bidirectional transformers for language understanding. **arXiv preprint arXiv:1810.04805**, 2018.
- DUBEY, Abhimanyu et al. **The Llama 3 Herd of Models**. [S.l.: s.n.], 2024. arXiv: [2407.21783 \[cs.AI\]](#). Available from: <https://arxiv.org/abs/2407.21783>.
- GANIN, Yaroslav; USTINOVA, Evgeniya; AJAKAN, Hana; GERMAIN, Pascal; LAROCHELLE, Hugo; LAVIOLETTE, François; MARCH, Mario; LEMPITSKY, Victor. Domain-adversarial training of neural networks. **Journal of machine learning research**, v. 17, n. 59, p. 1–35, 2016.
- GAO, Yunfan; AL., et. **Retrieval-Augmented Generation for Large Language Models: A Survey**. [S.l.: s.n.], 2024. arXiv: [2312.10997 \[cs.CL\]](#).
- HAN, Ridong; AL., et. **Is Information Extraction Solved by ChatGPT? An Analysis of Performance, Evaluation Criteria, Robustness and Errors**. [S.l.: s.n.], 2023. arXiv: [2305.14450 \[cs.CL\]](#).
- HOFFMANN, Jordan et al. **Training Compute-Optimal Large Language Models**. [S.l.: s.n.], 2022. arXiv: [2203.15556 \[cs.CL\]](#). Available from: <https://arxiv.org/abs/2203.15556>.
- HUANG, Lei; AL., et. **A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions**. [S.l.: s.n.], 2023. arXiv: [2311.05232 \[cs.CL\]](#).

- IZACARD, Gautier; GRAVE, Edouard. Leveraging passage retrieval with generative models for open domain question answering. **arXiv preprint arXiv:2007.01282**, 2020.
- JIANG, Albert Q.; AL., et. Mistral 7B, 2023. arXiv: [2310.06825 \[cs.CL\]](#).
- KANDPAL, Nikhil; AL., et. **Large Language Models Struggle to Learn Long-Tail Knowledge**. [S.l.: s.n.], 2023. arXiv: [2211.08411 \[cs.CL\]](#).
- KAPLAN, Jared et al. **Scaling Laws for Neural Language Models**. [S.l.: s.n.], 2020. arXiv: [2001.08361 \[cs.LG\]](#). Available from: <https://arxiv.org/abs/2001.08361>.
- KARPUKHIN, Vladimir; OĞUZ, Barlas; MIN, Sewon; LEWIS, Patrick; WU, Ledell; EDUNOV, Sergey; CHEN, Danqi; YIH, Wen-tau. Dense passage retrieval for open-domain question answering. **arXiv preprint arXiv:2004.04906**, 2020.
- KATZ, Daniel Martin; AL., et. **Natural Language Processing in the Legal Domain**. [S.l.: s.n.], 2023. arXiv: [2302.12039 \[cs.CL\]](#).
- KOWSRIHAWAT; AL., et. An information extraction framework for legal documents: A case study of thai supreme court verdicts. In: IEEE. 2015 12th International Joint Conference on Computer Science and Software Engineering (JCSSE). [S.l.: s.n.], 2015. P. 275–280.
- LECUN, Yann; BENGIO, Yoshua; HINTON, Geoffrey. Deep learning. **nature**, Nature Publishing Group UK London, v. 521, n. 7553, p. 436–444, 2015.
- LEWIS, Patrick et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. **Advances in Neural Information Processing Systems**, v. 33, p. 9459–9474, 2020.
- LI, Xiaoya; SUN, Xiaofei; MENG, Yuxian; LIANG, Junjun; WU, Fei; LI, Jiwei. Dice loss for data-imbalanced NLP tasks. **arXiv preprint arXiv:1911.02855**, 2019.
- LIU, Nelson F.; AL., et. **Lost in the Middle: How Language Models Use Long Contexts**. [S.l.: s.n.], 2023. arXiv: [2307.03172 \[cs.CL\]](#).
- LIU, Yinhan. Roberta: A robustly optimized bert pretraining approach. **arXiv preprint arXiv:1907.11692**, v. 364, 2019.

LUAN, Yi; HE, Luheng; OSTENDORF, Mari; HAJISHIRZI, Hannaneh. Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. **arXiv preprint arXiv:1808.09602**, 2018.

MINTZ, Mike; BILLS, Steven; SNOW, Rion; JURAFSKY, Dan. Distant supervision for relation extraction without labeled data. In: PROCEEDINGS of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP. [S.l.: s.n.], 2009. P. 1003–1011.

MITCHELL, T. M. **Machine Learning**. [S.l.]: McGraw-Hill, 1997. A foundational textbook often cited in machine learning papers, providing core definitions and algorithms.

NOGUEIRA, Rodrigo; CHO, Kyunghyun. Passage Re-ranking with BERT. **arXiv preprint arXiv:1901.04085**, 2019.

PEREIRA, Jayr; AL., et. INACIA: Integrating Large Language Models in Brazilian Audit Courts: Opportunities and Challenges. **Digit. Gov.: Res. Pract.**, Association for Computing Machinery, New York, NY, USA, Mar. 2024.

RADFORD, Alec. Improving language understanding by generative pre-training, 2018.

RADFORD, Alec; WU, Jeffrey; CHILD, Rewon; LUAN, David; AMODEI, Dario; SUTSKEVER, Ilya, et al. Language models are unsupervised multitask learners. **OpenAI blog**, v. 1, n. 8, p. 9, 2019.

RUMELHART, David E; HINTON, Geoffrey E; WILLIAMS, Ronald J. Learning representations by back-propagating errors. **nature**, Nature Publishing Group UK London, v. 323, n. 6088, p. 533–536, 1986.

SONG, Bosheng; LI, Fen; LIU, Yuansheng; ZENG, Xiangxiang. Deep learning methods for biomedical named entity recognition: a survey and qualitative comparison. **Briefings in Bioinformatics**, Oxford University Press, v. 22, n. 6, bbab282, 2021.

SOUZA, Fábio; AL., et. BERTimbau: pretrained BERT models for Brazilian Portuguese. In: 9TH Brazilian Conference on Intelligent Systems, BRACIS, Rio Grande do Sul, Brazil, October 20-23 (to appear). [S.l.: s.n.], 2020.

TOUVRON, Hugo; AL., et. Llama 2: Open Foundation and Fine-Tuned Chat Models, 2023. arXiv: [2307.09288 \[cs.CL\]](#).

TURING, Alan M. Computing Machinery and Intelligence. **Mind**, Oxford University Press, v. LIX, n. 236, p. 433–460, 1950.

VASWANI, A. Attention is all you need. **Advances in Neural Information Processing Systems**, 2017.

VIANNA; AL., et. Organizing Portuguese Legal Documents through Topic Discovery. In: PROCEEDINGS of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval. New York, NY, USA: Association for Computing Machinery, 2022.

WEI, Jason et al. **Emergent Abilities of Large Language Models**. [S.l.: s.n.], 2022. arXiv: [2206.07682 \[cs.CL\]](#). Available from: <https://arxiv.org/abs/2206.07682>.

WEI, Xiang; AL., et. **ChatIE: Zero-Shot Information Extraction via Chatting with ChatGPT**. [S.l.: s.n.], 2024. arXiv: [2402.10205 \[cs.CL\]](#).

XIE, Qizhe; LUONG, Minh-Thang; HOVY, Eduard; LE, Quoc V. Self-training with noisy student improves imagenet classification. In: PROCEEDINGS of the IEEE/CVF conference on computer vision and pattern recognition. [S.l.: s.n.], 2020. P. 10687–10698.

YANG, An et al. Qwen2 Technical Report. **arXiv preprint arXiv:2407.10671**, 2024.

ZHAO, Wayne Xin et al. A survey of large language models. **arXiv preprint arXiv:2303.18223**, 2023.

APPENDIX A – Artigo do TCC

Extracting Information from Brazilian Legal Documents with Retrieval Augmented Generation

Isabella V. de Aquino¹, Matheus M. dos Santos¹, Carina F. Dorneles¹, Jônata T. Carvalho¹

¹Department of Informatics and Statistics

Federal University of Santa Catarina (UFSC)

P.O. Box 5064 – 88.040-370 – Florianópolis – SC – Brazil

isabella.aquino@grad.ufsc.br, matheus.m.santos@posgrad.ufsc.br

{carina.dorneles, jonata.tyska}@ufsc.br

Abstract. *Extracting information from unstructured data is a challenge that has drawn increasing attention over time due to the exponential growth of stored digital data in modern society. Large Language Models (LLMs) have emerged as powerful tools that benefit from this abundance and have shown remarkable capabilities in Natural Language Processing tasks. Nonetheless, these models still encounter limitations on extraction tasks. Retrieval Augmented Generation (RAG) is a novel approach that combines classic retrieval techniques and LLMs to address some of these limitations. This paper proposes a workflow that allows the assessment of RAG experimental setups, including the multiple possibilities of parameters and LLMs, to extract structured data from Brazilian legal documents. We validated our proposal with experiments using forty legal documents and the extraction of two target variables. The best results obtained with our workflow showed an average extraction accuracy of 90%, significantly outperforming a regular expression strategy, with 58.75% average accuracy. Furthermore, our results show that each extracted variable potentially holds an optimal combination of parameters, highlighting the context-dependency of each extraction and, therefore, the proposed workflow's usefulness.*

1. Introduction

The increasing digitization of judicial and administrative processes worldwide has led to massive production and storage of legal documents. These documents are commonly unstructured, complex and contain crucial information for lawyers, judges, and prosecutors. Extracting this information typically requires extensive human annotation and management in external systems such as relational databases. In line with this scenario, several efforts have been made to handle and process legal documents in various countries, for instance, explored in [Bach and et al. 2019] for extracting references from Vietnamese legal documents, and in [Vianna and et al. 2022] for examining the processing and summarization of Portuguese legal documents.

In particular, the Brazilian public legal sector is an example of an organization dealing with great amounts of documents; almost 200,000¹ public procurement processes were successfully contracted from 2020 to 2023 by the Brazilian Federal Government, in which each one of them requires thorough documentation to formalize every step of the process. As a result, retrieving and extracting specific information, such as legal processes, contract identifiers, and involved municipalities from these numerous complex documents, poses a demanding task if done manually.

Moving forward, information extraction (IE) is an extensively researched subject in legal domains to overcome the presented challenges and has been applied and evaluated through multiple approaches, such as traditional pattern matching

¹<https://portal.datransparencia.gov.br/licitacoes>

[Cheng and et al. 2009]. Likewise, the work presented in [Kowsrihawatt and et al. 2015] achieves expressive results in extracting variables in legal documents through a proposed framework utilizing regular expressions. Overall, IE in legal domains is a rapidly evolving field with the potential to transform how legal professionals work and automating information extraction can provide valuable insights to legal entities and potentially aid broader analyses to detect and prevent fraud and corruption.

Then, Large Language Models have emerged offering the promise of understanding and generating human-like text at scale and in the legal domain [Katz and et al. 2023]. However, despite their impressive performance and variety of applications, LLMs still face inherent limitations when extracting structured information from unstructured data sources. LLMs knowingly struggle with domain-specific or knowledge-intensive tasks [Kandpal and et al. 2023], have their performance degraded when dealing with relevant information in the middle of long contexts [Liu and et al. 2023] and tend to produce “hallucinations” [Huang and et al. 2023] when searching for information beyond their training data.

In response to these challenges, Retrieval Augmented Generation (RAG) has emerged as a promising approach for enhancing the capabilities of LLMs in information extraction tasks [Gao and et al. 2024]. By combining classic retrieval techniques with LLMs, RAG systems enable the retrieval of relevant information from external sources during text generation, thereby mitigating domain-specific and context window limitations of LLMs and improving the accuracy and coherence of the generated text.

This paper proposes a workflow that leverages LLMs within RAG pipelines to extract structured information from Brazilian legal documents related to fraud in public procurement processes. However, RAG is a data-driven general framework, and its setup can be demanding once several different parameter types are required to be set beforehand. Our objective is to demonstrate the effectiveness of RAG in overcoming the challenges associated with information extraction from complex, domain-specific documents and propose a workflow that evaluates and indicates the best RAG parameter configurations for extracting a given variable. We extracted and evaluated two different variables of interest in forty different Brazilian legal documents utilizing our workflow. The results showed the proposed methodology’s effectiveness, which achieved an average accuracy of 90%, outperforming a baseline strategy based on regular expressions, which achieved 58.5%.

2. Related Work

Information extraction (IE) has become a significant explored subject [Doan and et al. 2006], keeping pace with the rapid increase of unstructured data availability in today’s data-driven world. IE tasks permeate various aspects of information and its forms of representation and structure, including visual aspects [Sarkhel and et al. 2021], and consider different languages [Zhu and et al. 2012]. It traditionally can be done by applying various approaches, such as the ones focused on annotating [Boisen and et al. 2000] or filtering [Wachsmuth and et al. 2013].

On that matter, Artificial Intelligence and NLP accompany IE advancements and research; [Han and et al. 2023] analyze and evaluate IE using ChatGPT, ranking LLMs encountered limitations, while [Wei and et al. 2024] explores IE systems chatting with ChatGPT in zero-shot settings.

Finally, IE is highly useful in legal applications, which commonly deal with expressive volumes of unstructured information. [Bhattacharya and et al. 2019] automatically identifies rhetorical roles in Indian legal cases. Then, [Pereira and et al. 2024] introduces basic information extraction from Brazilian audit court documents integrating

LLMs in a retrieval-augmented generation workflow.

Given the foregoing, extracting information from legal documents commonly encounters difficulties, such as formatting and structure variability, complicating pattern-matching strategies. As for NLP-driven strategies, sole LLMs are greatly affected by irrelevant and longer context, a big aspect of legal documents. Our work addresses these bottlenecks by enabling contextualization of variables and overcoming the need to feed entire documents into prompts with RAG, highlighting the usefulness of the paper.

3. Method

Our method is structured around a main RAG pipeline for the extraction, executing multiple times iteratively across multiple possible parameter configurations of RAG parameters. This approach facilitates comprehensive comparisons across various parameter sets, potentially identifying an optimal configuration for extracting a given variable. The list below outlines these parameters, grouping them by types:

- **Generation:** Parameters related to the generation step of RAG. The following parameter can be tested: Large Language Models (LLMs);
- **Chunking:** Parameters related to the documents chunking strategy. The following parameters can be tested: chunk size, which is the size of each of the split chunks of text; chunk overlap, which is the size of text overlap between adjacent chunks; and splitting strategy, which is usually the text splitter used to execute the chunking;
- **Embeddings:** Parameters related to the embeddings to be generated. The following parameter can be tested: embedding model, used to generate the embeddings from the documents' chunks, e.g. BERT models;
- **Retrieval:** Parameters related to the retrieval step of RAG. The following parameters can be tested: vector database, responsible to store and retrieve the embeddings and Top K value, which is the K-amount of retrieved chunks to serve as context on the extraction.

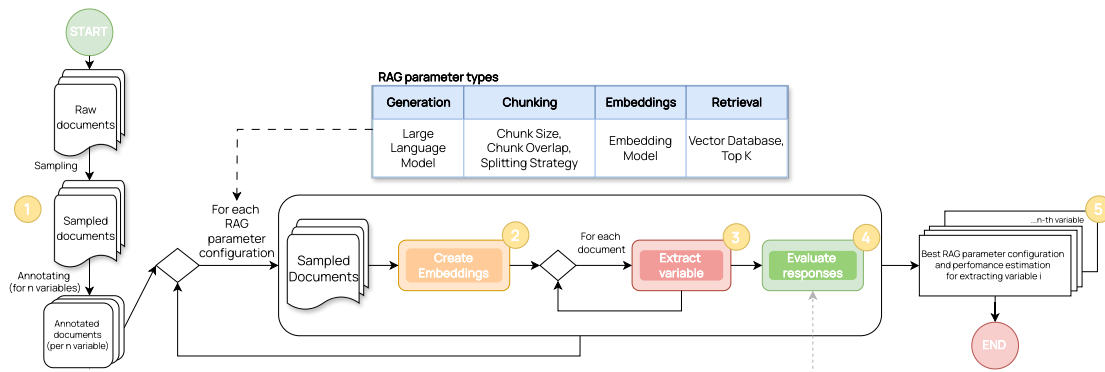


Figure 1. Main workflow overview.

Figure 1 depicts the overview of the proposed workflow, which will cover all combinations of parameter settings possible to extract the same chosen variable and compare the results among each other. A parameter configuration is a unique set of values for each of the RAG parameter types previously described due to iterating through all the available parameter options. Our proposal is based on the following steps:

- Step 1 establishes the beginning of our proposed workflow, initiated by sampling the documents available. We ensure all of them will have an expected value to be extracted for a given query and manually annotate every sampled document with its expected values. This step is the base of our further evaluation assessment step, represented by step number 4.

- Step 2 iterates all possible parameter configuration combinations within the selected options for each parameter type. Step 2 represents the embeddings creation for every sampled PDF document using the current configuration for chunking and embeddings. These embeddings will be used during the extraction step in the workflow.
- Step 3 constitutes the main RAG pipeline. It will retrieve the most relevant embeddings generated in the previous step related to a given query. It will insert them as the context in a prompt template and return direct responses containing or not the answer for the task.
- Step 4 evaluates all the extracted responses by comparing them directly to the foregoing annotated values, labeling as correct the responses that contain the exact expected information for a query.
- Step 5 outputs the best results parameter configurations and performance estimation for each extracted variable.

Lastly, a key aspect of this process is the iterative nature of the main pipeline execution. Various parameters are systematically altered based on a range of parameters values to be tested. This approach enables the evaluation across configurations and identification of the potential most suitable parameter setting for extracting a certain variable from the documents. It also aids decision-making in selecting from the potential options that can be applied to general RAG pipeline parameters by offering comparative results for each configuration and highlighting the best-obtained ones. In our proposed workflow, any of the previously stated parameters can be iteratively tested and analyzed by determining which options for each parameter type will be covered.

4. Experimental Evaluation

We analyzed and ran our experiments with forty selected Brazilian legal documents provided by Santa Catarina Government Agency for Law Enforcement and Prosecution of Crimes (MPSC), with an average of 26 pages and 60,000 characters each. Moreover, as mentioned in Section 3, while our proposed workflow allows the variation of any of the general RAG parameters, our experiments focused on alternating the parameters Large Language Models (Llama-2-7b, Llama-2-13b and Mistral-7B-v0.2), Chunk Size (128, 256, 512), Chunk Overlap Size (20, 50, 100, 200) and Top K (1, 3, 5, 8, 10, 12) and maintaining BERT Model, Splitting Strategy and Vector database as fixed parameters.

4.1. Data Preparation

The first step in the experiment, equivalent to step number 1 in Figure 1, is to prepare the available documents dataset to be used. This step is divided into two sub-steps: sampling and annotating. In the first sub-step, we filtered a smaller sample of the legal documents provided by MPSC, ensuring that all documents contained our study's analyzed variables. Then, to evaluate the accuracy of the experiments, the second sub-step was to manually annotated each selected document, mapping useful information including the variables to be evaluated. The annotations will be used to directly compare the model's responses to the constructed prompts, thereby assessing the accuracy of each extraction on every experiment.

4.2. Embeddings Creation

Embeddings were generated from the pieces of text parsed and chunked previously utilizing BERTimbau Large [Souza and et al. 2020], a BERT model pre-trained in Brazilian Portuguese, representing Step 2 in Figure 1. They were then stored in a vector database to manipulate and retrieve these embeddings. Chroma² was used as our option for vector database, a commonly chosen option for general RAG pipelines, highlighted for being open-source.

²<https://github.com/chroma-core/chroma>

4.3. Extracting variables

With the embeddings stored in the database, the next step was to embed the user’s query, retrieve the most similar embeddings through a similar search, and finally use them as context on the prompt final form, represented by the template illustrated in Figure 2. These steps correspond to the main RAG pipeline, identified by step 3 in Figure 1.

```
## Instructions
You are a helpful AI assistant and provide the response in Portuguese to the question based on the provided context.
Use the following chunks of context to answer the question at the end. If it is not possible to answer the question from the
context, just answer that you didn't find the answer.
## Context built with Top K relevant retrieved chunks
CONTEXT: [RETRIEVED CHUNKS]
>>>QUESTION<<<: [USER QUERY]
>>>ANSWER<<<:
```

Figure 2. English translation of prompt template used in every experiment.

The LLMs options used in our experiments were the Llama-2 family chat models [Touvron and et al. 2023] and Mistral-7B-Instruct [Jiang and et al. 2023], all of them loaded locally with a RTX 3090 as the main GPU. This setup ensured privacy to handle the legal document’s sensitive information, however, limited the involved models used in our study, making it impossible to handle bigger models on the current analysis.

4.4. Evaluation Metrics

While several aspects of evaluation around RAG can be measured [Gao and et al. 2024], our work primarily concentrates on direct accuracy assessment. We specifically examine whether the generated response by the model precisely matches the annotated value associated with a particular document. This evaluation occurs in step 4 in Figure 1, which will divide the quantity of successfully matched extracted values by its annotation value by the total amount of documents.

4.5. Evaluated extracted variables

As previously stated, our work focuses on extracting two variables: public procurement process identifiers and municipalities of irregularity. The public procurement process identifier is a string that identifies a certain public procurement process for a municipality, and it is consistently presented in the format X/YYYY, where 'X' represents any numerical sequence and 'YYYY' denotes a four-digit year. The municipality of irregularity refers to the name of the municipality where fraud was committed through public procurement processes. Both variables have 145 associated experiments, one for every unique configuration possible interchanging the parameters detailed earlier in this Section.

5. Results and Discussion

As our baseline, we built a regular expression that looked for matches using the mentioned formats. When comparing the mode of the matches, extracting the variable with a regular expression reached a maximum of 35% accuracy against the best accuracy of 88% using our workflow when extracting public process identifiers and a maximum accuracy of 82.5% compared with our best accuracy of 93% when extracting municipalities. This comparison is visible in Figure 3, where the best obtained accuracies through our proposed method overcomes expressively the result obtained by the regular expression, when extracting public procurement process identifiers. For extracting municipalities, our experiments still bests the regular expression results by 10.5%. These results underscore the effectiveness of our method, overcoming regular expressions by contextualizing the

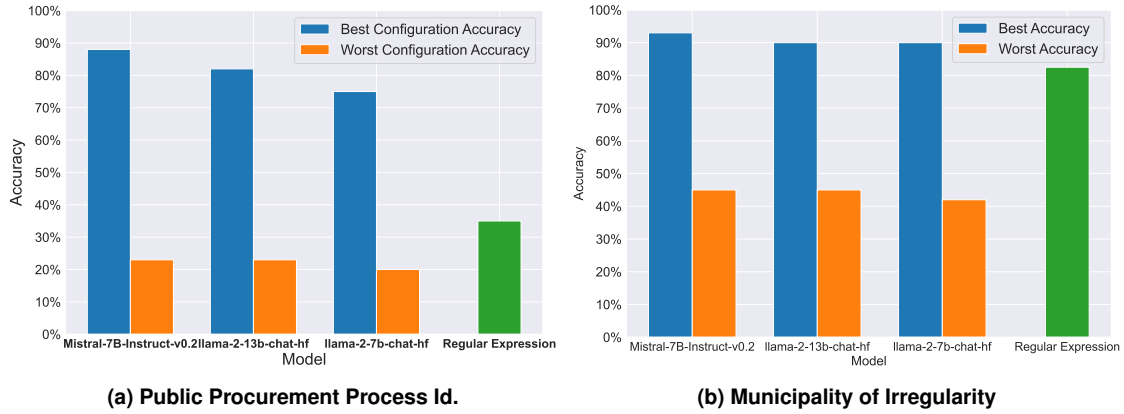


Figure 3. Best and Worst results per model vs Regular Expression

variables on the prompt fed to the LLMs. The built regular expressions for public procurement process identifier and municipality of irregularity are `\b\d+\/\d{4}\b` and `Município de ([A-Z][a-z]+(?:\s[A-Z][a-z]+)*)`, respectively.

Then, Figure 4 illustrates the Top K evolution and its impact on obtained accuracies on extracting both variables on fixed chunk sizes and chunk overlap. It suggests that the increase of Top k values directly impacts on the accuracy in pipelines with small chunks, increasing the probability of the retriever returning the accurate answer among the available embeddings.

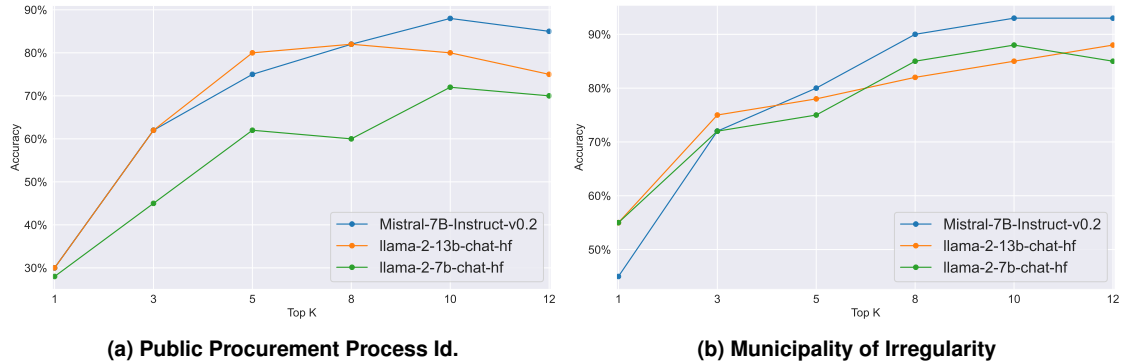


Figure 4. Top K evolution with fixed chunk size as 128 and chunk overlap as 20.

6. Conclusion and Future Works

In conclusion, legal documents are often extensive and irregularly structured, and extracting relevant and structured data from these documents still poses a significant challenge. In this paper, we presented and evaluated a promising approach utilizing retrieval-augmented generation to extract two different variables of interest, obtaining an average accuracy of 90%, which overcame pattern matching measured accuracies in both scenarios. Our work addresses a common bottleneck for traditional extraction techniques — contextualization, and is part of a new paradigm of zero-shot IE, not requiring training or finetuning any models, representing a step forward on IE in legal domains.

Finally, our future works will focus on overcoming dataset and hardware limitations, in order to evaluate more expressive samples and include more robust Large Language Models. It will also be centralized in formalizing the proposed method as a RAG parameter evaluator framework for any type RAG pipeline for any system.

References

- Bach and et al. (2019). Reference extraction from vietnamese legal documents. SoICT '19, page 486–493, New York, NY, USA. Association for Computing Machinery.
- Bhattacharya, P. and et al. (2019). Identification of rhetorical roles of sentences in indian legal judgments.
- Boisen, S. and et al. (2000). Annotating resources for information extraction. In *Proceedings of the Second International Conference on Language Resources and Evaluation (LREC'00)*, Athens, Greece. European Language Resources Association (ELRA).
- Cheng and et al. (2009). Information extraction from legal documents. In *2009 Eighth International Symposium on Natural Language Processing*.
- Doan, A. and et al. (2006). Managing information extraction: state of the art and research directions. In *Proceedings of the 2006 ACM SIGMOD International Conference on Management of Data*, SIGMOD '06', page 799–800, New York, NY, USA. Association for Computing Machinery.
- Gao, Y. and et al. (2024). Retrieval-augmented generation for large language models: A survey.
- Han, R. and et al. (2023). Is information extraction solved by chatgpt? an analysis of performance, evaluation criteria, robustness and errors.
- Huang, L. and et al. (2023). A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions.
- Jiang, A. Q. and et al. (2023). Mistral 7b.
- Kandpal, N. and et al. (2023). Large language models struggle to learn long-tail knowledge.
- Katz, D. M. and et al. (2023). Natural language processing in the legal domain.
- Kowsrihawatt and et al. (2015). An information extraction framework for legal documents: A case study of thai supreme court verdicts. In *2015 12th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, pages 275–280. IEEE.
- Liu, N. F. and et al. (2023). Lost in the middle: How language models use long contexts.
- Pereira, J. and et al. (2024). Inacia: Integrating large language models in brazilian audit courts: Opportunities and challenges. *Digit. Gov.: Res. Pract.*
- Sarkhel, R. and et al. (2021). Improving information extraction from visually rich documents using visual span representations. *Proc. VLDB Endow.*, 14(5):822–834.
- Souza, F. and et al. (2020). BERTimbau: pretrained BERT models for Brazilian Portuguese. In *9th Brazilian Conference on Intelligent Systems, BRACIS, Rio Grande do Sul, Brazil, October 20-23 (to appear)*.
- Touvron, H. and et al. (2023). Llama 2: Open foundation and fine-tuned chat models.
- Vianna and et al. (2022). Organizing portuguese legal documents through topic discovery. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '22, page 3388–3392, New York, NY, USA. Association for Computing Machinery.
- Wachsmuth, H. and et al. (2013). Information extraction as a filtering task. In *Proceedings of the 22nd ACM International Conference on Information & Knowledge Management*, CIKM '13, page 2049–2058, New York, NY, USA. Association for Computing Machinery.

Wei, X. and et al. (2024). Chatie: Zero-shot information extraction via chatting with chatgpt.

Zhu, W. and et al. (2012). Cross language information extraction for digitized textbooks of specific domains. In *2012 IEEE 12th International Conference on Computer and Information Technology*, pages 1114–1118.

APPENDIX B – CÓDIGO FONTE DO TCC

The code developed, including the workflow and the proposed interface, is made available at <https://github.com/isabellaaquino/ragify-app-portuguese>. Make sure to follow the step by step installation included in the README file before utilizing the framework.