UNISINOS

Programa de Pós-Graduação em

Computação Aplicada

Mestrado/Doutorado Acadêmico

Luciano Zanuz

DEVELOPING EFFECTIVE AI AND LAW APPLICATIONS: A methodological proposal for developing natural language processing applications based on transformers, pre-trained models and transfer learning

São Leopoldo, 2025

Luciano Zanuz

DEVELOPING EFFECTIVE AI AND LAW APPLICATIONS:
A methodological proposal for developing natural language processing applications based on transformers, pre-trained models and transfer learning

Thesis presented as a partial requirement for obtaining the title of Doctor, by the Applied Computing Graduate Program at the University of Vale do Rio dos Sinos — UNISINOS

Advisor:
Prof. Dr. Sandro José Rigo

São Leopoldo
2025

Z34d Zanuz, Luciano.
Developing effective AI and law applications : a methodological proposal for developing natural language processing applications based on transformers, pre-trained models and transfer learning / by Luciano Zanuz. – 2025.
235 p. : il. ; 30 cm.

Thesis (doctor's degree) — University of Vale do Rio dos Sinos, Applied Computing Graduate Program, São Leopoldo, RS, 2025.
"Advisor: Dr. Sandro José Rigo".

1. Artificial intelligence (AI). 2. Law. 3. Natural language processing (NLP). 4. Transformers. 5. Transfer learning. I. Title.

UDC: 004.8:34

Cataloging in Publication (CIP):
Librarian Alessandro Dietrich - CRB 10/2338

(Esta folha serve somente para guardar o lugar da verdadeira folha de aprovação, que é obtida após a defesa do trabalho. Este item é obrigatório, exceto no caso de TCCs.)

ACKNOWLEDGEMENTS

This thesis is the result of a long journey that would not have been possible without the support, guidance, and encouragement of many people, to whom I am deeply grateful.

First and foremost, I would like to thank my beloved wife, Denise, and my children, Rafael and Gabriela, for their unwavering support, patience, and understanding during the many moments of absence and dedication throughout this journey.

I am also profoundly grateful to my parents, João and Elda, and to my parents-in-law, Mercino and Mara, whose constant support has always been a source of strength and motivation.

My sincere thanks go to my advisor, Professor Dr. Sandro José Rigo, for his continued support, trust, and insightful guidance. His ability to lead with clarity and lightness made the research process both enriching and enjoyable.

I am also deeply grateful to the members of my doctoral examination committee for their time, careful reading, and valuable feedback: Professor Dr. Renata Vieira, Professor Dr. Fausto Santos de Morais, and Professor Dr. Jorge Luis Victoria Barbosa. Their thoughtful comments and questions contributed significantly to the refinement and improvement of this work.

I am grateful to the Court of Justice of the State of Rio Grande do Sul for the institutional support that made this research possible. In particular, I would like to thank the following magistrates for their valuable contributions and encouragement: Desembargador Ricardo Torres Hermann, Desembargador Antonio Vinicius Amaro da Silveira, Dr. André Dal Soglio Coelho, Dr. André Luís De Aguiar Tesheiner, Dr. Daniel Pellegrino Kredens, and Dr. Diego Viegas Sato Barbosa. I am especially thankful to the latter three for their support in validating the experiments conducted during this research.

Special thanks to legal advisors Taynara Silva Arceno and Fernanda Vargas dos Santos for their collaboration and support in validating the experiments conducted during this research, and to Dr. Alsones Balestrin, advisor to the Presidency, for his commitment to fostering innovation and promoting the use of artificial intelligence at the Court of Justice of Rio Grande do Sul.

I also wish to thank Dr. Ana Luiza Treichel Vianna and Gabriel Tamujo Meyrer for their valuable contributions to the writing of one of the scientific articles developed during this research.

My appreciation extends to my colleagues at the Information and Communication Technology Directorate: Vanessa Barbisan Pires, Clairton Buligon, Antonio Braz da Silva Neto, Débora Pritsch, and Sílvia Maria Saggiorato, for their constant support and encouragement. A special acknowledgment goes to Rodrigo Schneider, a close collaborator in both real-world AI projects and academic research in AI applied to Law.

A very special thank you to Denis Andrei de Araujo and Juliano Pacheco, with whom I had the opportunity to work on the first AI projects at the Court of Justice of Rio Grande do Sul. I am especially grateful to Dr. Denis for his inspiring question: "When are you going to start your PhD?", a question that marked the beginning of this academic endeavor.

Finally, I am thankful to all the researchers, colleagues, and friends who, directly or indirectly, contributed to this work through conversations, collaboration, and encouragement. I also wish to acknowledge the anonymous reviewers of the scientific articles submitted during the course of this research for their constructive feedback, which helped improve the quality of this work.

To all of you, my deepest gratitude.

"*Education is the most powerful weapon which you can use to change the world.*"
Nelson Mandela

RESUMO

CONTEXTO: A interseção entre Inteligência Artificial (IA) e Direito vem sendo explorada desde os primórdios da pesquisa em IA. Nos últimos anos, tanto instituições judiciais quanto profissionais do direito têm adotado, de forma crescente, tecnologias baseadas em IA para otimizar processos, apoiar a tomada de decisões jurídicas, automatizar tarefas repetitivas e extrair informações estruturadas de textos legais. Essas aplicações são predominantemente impulsionadas por técnicas de Processamento de Linguagem Natural (PLN), dada a natureza textual dos procedimentos jurídicos. Avanços em aprendizado profundo, especialmente nas arquiteturas baseadas em transformadores e nos modelos de linguagem pré-treinados, transformaram significativamente o desenvolvimento de sistemas de PLN e alcançaram estado-da-arte em diversas tarefas. PROBLEMA: Apesar do potencial desses avanços, ainda não existe uma metodologia unificada e específica de domínio para o desenvolvimento de aplicações de IA no campo jurídico. Essa lacuna limita a adoção eficaz de soluções baseadas em PLN em contextos legais reais. Além disso, a maioria dos recursos de ponta está disponível principalmente em inglês, o que representa uma barreira para sistemas jurídicos de língua portuguesa. SOLUÇÃO: Esta tese propõe uma metodologia estruturada para o desenvolvimento de aplicações de IA aplicadas ao Direito, fundamentada em paradigmas modernos de PLN, como transformadores, aprendizado por transferência e adaptação ao domínio. A metodologia aborda tanto os desafios técnicos quanto os específicos do domínio jurídico e inclui componentes práticos, como conjuntos de dados, modelos ajustados e ferramentas de avaliação, todos voltados ao contexto jurídico em português. MÉTODO PROPOSTO: A metodologia proposta é composta por quatro etapas principais e enfatiza a colaboração interdisciplinar entre equipes jurídicas e técnicas. Ela define um fluxo de desenvolvimento claro, desde a definição do problema até a implantação da solução, incorporando validações iterativas, a criação de conjuntos de dados em nível de aplicação e o uso de mecanismos de IA Explicável, quando apropriado. RESULTADOS: A metodologia foi validada por meio de diversos experimentos, incluindo o desenvolvimento de modelos de Reconhecimento de Entidades Nomeadas (NER) jurídicas que alcançaram novos resultados de estado da arte no conjunto de dados LeNER-Br, além de experimentos com técnicas de ajuste fino específicas de contexto e ajustes eficientes em parâmetros, visando melhorar o desempenho e a adaptabilidade dos modelos. Também foi implementada uma aplicação real para geração automatizada de relatórios de sentença, apoiada por um novo framework de avaliação que combina métricas automatizadas com avaliação humana. Este trabalho ainda oferece recursos práticos em português e fornece insights sobre a correlação entre avaliações humanas e automáticas de saídas de IA no domínio jurídico, demonstrando a viabilidade e os benefícios de uma abordagem estruturada e adaptada ao domínio para o desenvolvimento de IA aplicada ao Direito.

Palavras-chave: Inteligência Artificial e Direito. Processamento de Linguagem Natural. Transformadores.

ABSTRACT

CONTEXT: The intersection between Artificial Intelligence (AI) and Law has been explored since the early days of AI research. In recent years, both judicial institutions and legal professionals have increasingly adopted AI technologies to streamline processes, support legal decision-making, automate repetitive tasks, and extract structured information from legal texts. These applications are predominantly powered by Natural Language Processing (NLP), given the textual nature of legal proceedings. Advances in deep learning, particularly transformer-based architectures and pre-trained language models, have significantly reshaped the development of NLP systems and achieved state-of-the-art results across many downstream tasks. PROBLEM: Despite the potential of these advances, there remains a lack of a unified, domain-specific methodology for developing AI applications in the legal field. This gap limits the effective adoption of NLP-based AI solutions in real-world legal contexts. Additionally, most cutting-edge resources are available primarily in English, creating a barrier for Portuguese-speaking legal systems. SOLUTION: This thesis proposes a structured methodology for developing AI and Law applications, grounded in modern NLP paradigms such as transformers, transfer learning, and domain adaptation. The methodology addresses both technical and domain-specific challenges and includes practical components, such as datasets, fine-tuned models, and evaluation tools, tailored to the Portuguese legal context. PROPOSED METHOD: The proposed methodology is composed of four main steps and emphasizes interdisciplinary collaboration between legal and technical teams. It defines a clear development flow, from problem definition to deployment, incorporating iterative validation, the creation of application-level datasets, and the use of Explainable AI mechanisms where applicable. RESULTS: The methodology was validated through multiple experiments, including the development of Legal Named Entity Recognition (NER) models that achieved new state-of-the-art results on the LeNER-Br dataset, alongside experiments with context-specific and parameter-efficient fine-tuning techniques to enhance model performance and adaptability. A real-world application for AI-generated judgment reports was implemented, supported by a novel evaluation framework combining automated metrics and human assessment. This work also provides practical resources in Portuguese and insights into the correlation between human and automated evaluations of AI outputs in the legal domain, demonstrating the feasibility and benefits of a structured, domain-adapted approach to AI development in legal applications.

Keywords: Artificial Intelligence and Law. Natural Language Processing. Transformers.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

Ex	Example
e.g.	*exempli gratia* in Latin, which means "for example" in English
i.e.	*id est* in Latin, which means "that is" in English
lbs	pound

LIST OF ACRONYMS

AI	Artificial Intelligence
ANN	Artificial Neural Network
API	Application Programming Interface
ASCII	American Standard Code for Information Interchange
ATF	Augmented Term Frequency
BiLM	Bidirectional Language Modeling
BPE	Byte-Pair Encoding
CAIL	Chinese AI and Law challenge
CART	Classification and Regression Trees
CNJ	National Council of Justice
CNN	Convolutional Neural Network
CPU	Central Processing Unit
CRF	Conditional Random Field
DNN	Deep Neural Network
DPCNN	Deep Pyramid Convolutional Neural Networks
DT	Decision Tree
ECtHR	European Court of Human Rights
ETE_MN	End-To-End Memory Network
GB	Gigabyte
GBM	Gradient Boosting Machine
GNN	Graph-based Neural Networks
GPLVM	Gaussian Process Latent Variable Models
GPU	Graphics Processing Unit
HAN	Hierarchical Attention Networks
HTML	Hypertext Markup Language
ICAIL	International Conference of Artificial Intelligence and Law
k-NN	k-Nearest Neighbors
LIME	Local Interpretable Model-agnostic Explanations
LKIF	Legal Knowledge Interchange Format
LLE	Locally-Linear Embedding
LLM	Large Language Model
LM	Language Modeling
LR	Logistic Regression

LSA	Latent Semantic Analysis
LSTM	Long Short-Term Memory
LTF	Logarithmic Term Frequency
MDS	Multidimensional Scaling
MI	Mutual Information
ML-KNN	Multi-Label k-Nearest Neighbor
ML-LOC	Multi-Label learning using LOcal Correlation
MLM	Masked Language Modeling
MLN	Markov Logic Network
MLP	Multilayer Perceptron
NB	Naïve Bayes
NER	Named Entity Recognition
NLP	Natural Language Processing
NLPS	Natural Language Processing (almost) from Scratch
NN	Neural Network
NPLM	Neural Probabilistic Language Model
PART	Projective Adaptive Resonance Theory
PCA	Principal Component Analysis
PEFT	Parameter-Efficient Fine-Tuning
PO	Product Owner
PLM	Pre-trained Language Model
QQP	Quora Question Pairs
RAG	Retrieval-Augmented Generation
RF	Random Forest
RFE	Recursive Feature Elimination
RLHF	Reinforcement Learning from Human Feedback
RNN	Recurrent Neural Networks
SHAP	SHapley Additive exPlanation
SOTA	State-Of-The-Art
SLR	Systematic Literature Reviews
SMS	Systematic Mapping Studies
SNE	Stochastic Neighbor Embedding
STF	Federal Supreme Court
STJ	Superior Court of Justice

SVM	Support Vector Machine
TAM	Technology Acceptance Model
TF	Term Frequency
TF-IDF	Term Frequency Inverse Document Frequency
TPU	Tensor Processing Unit
URL	Uniform Resource Locator
XAI	eXplainable Artificial Intelligence

LIST OF SYMBOLS

CO_2 Carbon Dioxide

CONTENTS

1 INTRODUCTION

"The reasonable man adapts himself to the world; the unreasonable one persists in trying to adapt the world to himself. Therefore all progress depends on the unreasonable man."

George Bernard Shaw

Since the Dartmouth Conference officially introduced the concept of "Artificial Intelligence" in 1956, Artificial Intelligence (AI) has profoundly transformed human society. In recent years, its impact has accelerated, driven by advances in deep learning, human-machine collaboration, open intelligence, and autonomous systems (ZHANG et al., 2019a). The application of Artificial Intelligence to the legal field has evolved over more than six decades, beginning with Lucien Mehl's 1958 article "Automation in The Legal World." In 1963, Lawlor envisioned a future in which computers would be capable of analyzing and predicting court decisions (LAWLOR, 1963). However, it was only with the establishment of the International Conference on Artificial Intelligence and Law (ICAIL) in 1987 that the field gained consistent academic visibility. The first ICAIL marks the inception of the AI and Law research community (BENCH-CAPON et al., 2012).

According to (RISSLAND; ASHLEY; LOUI, 2003), AI and Law is a significant and challenging subfield within Artificial Intelligence, requiring solutions to fundamental problems such as reasoning, knowledge representation, and learning. As (ROBALDO et al., 2019) highlights, Law has always been an attractive domain for linguistic and semantic technologies due to its central role in governance, which in turn motivates cutting-edge research in Natural Language Processing (NLP).

Initial efforts in AI and Law focused on knowledge representation and rule-based systems. Since the 2000s, however, the field has increasingly embraced data-driven approaches, particularly those based on Machine Learning. Currently, AI applications in Law are generally categorized into three groups (SURDEN, 2019): (i) support for law practitioners, (ii) support for law administrators, and (iii) tools aimed at law users. For law administrators, such as judges, legislators, and government entities, AI has the potential to assist in decision-making, sentence generation, and risk assessments for criminal defendants, among other tasks.

Decision support systems for judges, for instance, aim to predict trial outcomes, highlight critical elements of cases, suggest similar precedents, assess document completeness, and improve triage processes by estimating case duration or complexity (BRANTING et al., 2018; ZHANG et al., 2019a).

Beyond decision support, the full digitization of legal proceedings across Brazil's judiciary and the influx of electronically submitted cases has created an unprecedented volume of documents that can benefit from intelligent processing. This opens opportunities for automation in routine tasks such as document classification, information extraction, and legal triage. Tech-

niques such as Named Entity Recognition (NER), Sentiment Analysis, Text Summarization, Aspect Mining, and Topic Modeling are particularly relevant.

Recent advances in AI, especially in NLP and Deep Learning, have made it possible to automatically analyze legal texts and build predictive models based on the semantic content of case law. While earlier research in judicial outcome prediction focused heavily on structured, non-textual information (LIU; CHEN, 2017), later studies on U.S. Supreme Court decisions revealed that incorporating legal text significantly enhances predictive performance compared to models based solely on numerical data (KATZ; II; BLACKMAN, 2017; KAUFMAN; KRAFT; SEN, 2017; SIM; ROUTLEDGE; SMITH, 2014). These studies suggest that factual narratives in case documents strongly influence outcomes, and neural models can learn such associations from textual content (ALETRAS et al., 2016; LIU; HSIEH, 2006; SULEA et al., 2017). Recent work using deep learning and neural architectures has shown strong performance in legal text classification tasks (ZHANG et al., 2019a), and the use of legal ontologies (DAS; ROY; MAJUMDAR, 2020; EL GHOSH et al., 2017; FAWEI et al., 2018; KURCHEEVA et al., 2019; SPEROTTO; BELCHIOR; AGUIAR, 2019) has further contributed to building more interpretable and effective models.

NLP has evolved from a statistically driven discipline to one powered by machine learning and deep learning. Initial word embeddings such as Word2Vec (MIKOLOV et al., 2013) and GLoVE (PENNINGTON; SOCHER; MANNING, 2014) gave way to contextualized embeddings from pre-trained language models, such as ELMo (PETERS et al., 2018), ULMFit (HOWARD; RUDER, 2018), and more recently, transformer-based architectures (VASWANI et al., 2017; WOLF et al., 2020), including BERT (DEVLIN et al., 2019). These advances rapidly elevated the State-Of-The-Art (SOTA) across NLP tasks (VASWANI et al., 2017), bringing the field closer to achieving the equivalent of "ImageNet for language" (RUDER, 2018), a generalizable, foundational platform for language understanding.

Currently, transformer-based models form the backbone of most advanced NLP systems. Even more recently, models of significantly larger scale, known as Large Language Models (LLMs), have emerged, containing tens or hundreds of billions of parameters (ZHAO et al., 2023), offering substantial performance improvements and previously unseen capabilities such as in-context learning. Unlike earlier encoder-based models like BERT (DEVLIN et al., 2019) or decoder-based GPT-2 (RADFORD et al., 2019), modern LLMs (e.g., GPT-4 (OPENAI et al., 2024), PaLM (CHOWDHERY et al., 2024), LLaMA (TOUVRON et al., 2023)) are typically decoder-only and optimized for generative tasks, such as text generation. For this reason, these models are also broadly referred to as Generative AI.

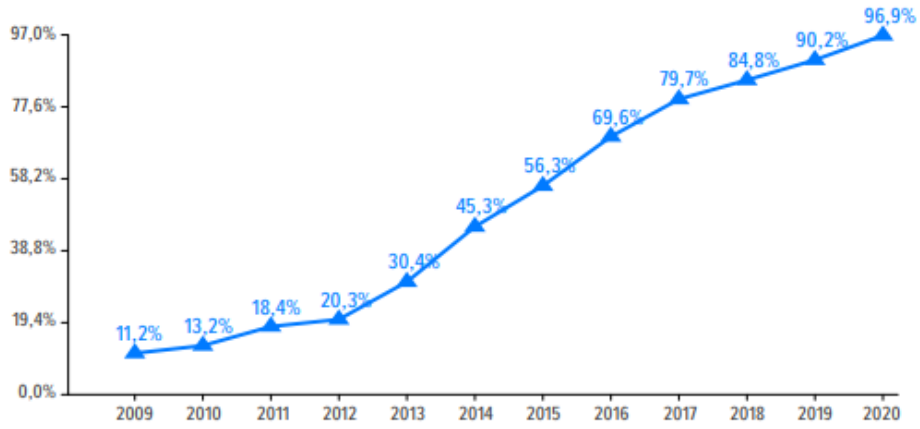
It is worth noting that the literature reviewed and the experiments reported in this thesis were mostly conducted in the early stages of this doctoral research. Consequently, some content may appear outdated when compared to the rapid technological advancements in AI and NLP observed in recent years. Nonetheless, these foundations remain relevant for understanding the methodological evolution and the challenges addressed by this work.

1.1 Motivation

As illustrated in Figure 1, recent years have seen a substantial increase in the percentage of lawsuits processed electronically in Brazil, replacing the traditional paper-based system (CNJ, 2021). According to CNJ (2024a), only in 2023, 35.1 million new cases were filed electronically in the Brazilian courts. This figure is part of a broader trend, totaling over 253.3 million cases received in digital format over the last 15 years.

This digital transformation presents a significant opportunity to leverage the textual content of judicial documents through artificial intelligence, particularly natural language processing (NLP) techniques. Potential applications include decision support for judges, prediction of case outcomes, document classification, information extraction, text summarization, and various other tasks that benefit from structured analysis of unstructured legal texts.

Figure 1: Evolution in the percentage of electronic lawsuits in Brazil



Source: (CNJ, 2021)

The recent advances in NLP, particularly with the advent of deep learning and pre-trained language models, have transformed the way text-based software solutions are developed. Although a wide array of techniques is currently available, there is still a lack of a consolidated and structured methodology to guide the development of NLP applications, especially within specialized domains such as Law.

In addition, most of the existing resources, including architectures, pre-trained models, and datasets, are predominantly available in English. Given the linguistic nature of NLP, state-of-the-art performance can only be achieved when domain-specific resources are also available in the target language. In this context, Portuguese remains an under-resourced language, particularly for legal applications.

Although NLP techniques can be applied to general texts, domain-specific applications often require tailored adaptations and specialized resources to reach optimal performance. This is particularly true for AI applications in the legal domain, where the technical language, stylistic conventions, and conceptual structures of legal texts demand customized approaches. Develop-

ing applications for the legal field, therefore, requires more than just the generic application of NLP techniques, it demands a methodological framework that incorporates linguistic, institutional, and functional specificities of the legal system.

Furthermore, while recent NLP models often achieve impressive results in academic benchmarks, these metrics do not necessarily translate into effective performance in real-world systems. Academic datasets are typically designed for research evaluation and may not capture the variability and complexity found in real use cases. In the legal context, these variations may include regional linguistic features, professional writing styles (e.g., texts authored by lawyers vs. judges), and the specificities of each legal task.

To be truly effective in practical settings, AI-based systems must also incorporate business logic and domain constraints, aligning their outputs with the actual needs and workflows of end users. Simply deploying an AI model, even one with excellent benchmark performance, is not sufficient. It is necessary to integrate preprocessing, postprocessing, and domain-specific rules to ensure that the system operates reliably within its intended environment.

Given this scenario, it becomes clear that enabling the development of efficient and effective AI applications in the legal domain requires the establishment of a robust methodological foundation. This includes a clearly defined development process and access to high-quality resources in both the Portuguese language and the legal context. This thesis aims to address this gap by proposing a methodological framework specifically designed for the development of AI and Law applications, grounded in the current state of the art in NLP and adapted to the unique challenges of legal practice in Brazil.

1.2 Research question

> What are the essential elements of a general methodology for developing Artificial Intelligence applications in the legal domain?

1.3 Goals

The main objective of this research is to propose a general methodology for the development of Artificial Intelligence applications applied to Law. To achieve this overarching goal, the following specific objectives are defined:

- Conduct an initial exploratory study to understand the interdisciplinary field of Artificial Intelligence and Law;
- Carry out a Systematic Mapping Study to identify recent works on decision support systems in the legal domain, analyzing techniques, evaluation metrics, development platforms, and results, in order to identify research gaps and opportunities;

- Survey the state-of-the-art in Natural Language Processing for the Portuguese language;
- Map current initiatives and research on AI applications in the Brazilian Judiciary;
- Propose a methodological framework for the development of AI applications tailored to the legal context;
- Implement and document the proposed methodology;
- Validate the proposed methodology through the application of standard metrics and a series of structured experiments;
- Make available resources in Portuguese for the development of legal AI applications, such as pre-trained models, datasets, and fine-tuning tools.

1.4 Contributions

This thesis makes several original scientific and practical contributions to the field of Artificial Intelligence and Law. Its central contribution is the formulation of a general and structured methodology for the development of AI applications tailored to the legal domain. This methodology leverages recent advances in Natural Language Processing (NLP), including transformer-based architectures, pre-trained language models, transfer learning, and domain-specific adaptation, and addresses both technical and legal-specific challenges in a unified, reproducible, and application-oriented framework.

Beyond the methodological proposal, this research delivers concrete resources and experimental validations that advance the development, evaluation, and practical deployment of AI systems in the legal context, particularly in Brazilian Portuguese. It also contributes to the scientific understanding of how legal tasks interact with state-of-the-art NLP techniques and evaluation methods.

The main contributions of this work are summarized as follows:

- A Systematic Mapping Study on AI-based decision support systems for judges, highlighting existing approaches, gaps, and research opportunities;
- The first BERT-based language models fine-tuned exclusively on Brazilian Portuguese for Legal Named Entity Recognition (NER), which reached new state-of-the-art results on the LeNER-Br dataset;
- A prototype application for judicial users, enabling the evaluation of model performance on real legal documents;
- An evaluation method specifically designed to assess the quality of AI-generated judgment reports;

- A domain-specific application-level dataset to support evaluation of generative legal text models;
- A web-based tool to facilitate human evaluation of automatically generated judgment reports;
- A set of LLM-as-a-Judge prompts for automatic assessment of generated legal texts;
- A correlation analysis between automated metrics and human judgment in legal AI evaluation;
- The identification of the most effective model and prompting configuration for generating judgment reports;
- A general methodology for developing AI applications in the legal domain, grounded in NLP best practices and adapted to legal language and tasks;
- Reusable components of the methodology (e.g., datasets, fine-tuned models, evaluation tools), enabling faster and higher-quality development of new applications;
- A demonstrable improvement in the quality of AI applications applied to Law, resulting from the structured and domain-specific approach proposed in this work.

Collectively, these contributions support the advancement of trustworthy, effective, and domain-sensitive AI systems in the legal field, and provide a structured foundation for future research and application development in Portuguese-language legal contexts.

1.5 Text organization

The remainder of this thesis is organized as follows:

- Chapter 2 presents the theoretical foundations that support this research, including the evolution of Artificial Intelligence, key concepts in Natural Language Processing, and recent advances in transformer-based models and transfer learning;
- Chapter 3 reviews related work on the application of AI in the legal domain, with a focus on Portuguese-language initiatives, including a systematic mapping study on AI-based decision support systems for judges and a broader survey of relevant efforts in Brazil, such as the development of legal datasets and language models;
- Chapter 4 details the proposed methodology for developing AI and Law applications, outlining its structure, components, and rationale, as well as optional extensions to address domain-specific needs;

- Chapter 5 presents the experiments conducted to validate the methodology, demonstrating its feasibility and effectiveness across multiple components and tasks, including a peer-reviewed study and additional experiments;
- Chapter 6 concludes the thesis, summarizing the main contributions, discussing implications, and proposing directions for future work.

2 THEORETICAL FOUNDATION

"Any sufficiently advanced technology is indistinguishable from magic."
Arthur C. Clarke

This chapter presents the key concepts relevant to the research, providing the necessary background for a comprehensive understanding of the proposed approach.

2.1 AI applied to Law

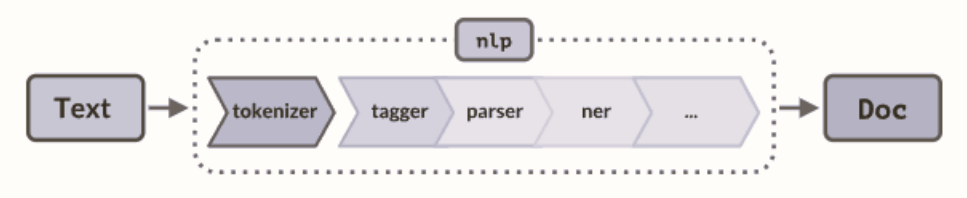
Considering the significant increase in legal documents produced and available in recent years, the digital processing of such materials is leading to the need to develop AI systems that support the automatic extraction of relevant information (BAPTISTA; COSTA, 2019). On the other hand, there are predictions for a not-so-distant future in which the legal profession may cease to exist in the current molds, considering the impact of AI in the area (SAUNDERS, 2016).

In Brazil, the use of AI in Law is much more recent, driven by the emergence of lawtechs and the incubation of startups in law firms, mainly using systems with machine learning and NLP to extract and process relevant information from legal documents (Lawgorithm, 2019). According to Marr (2018), AI applications in Law involve document review, legal research, help with due diligence, contract review and management, prediction of legal results and even divorce automation. AI systems such as "ROSS Intelligence" (SILLS, 2016) use NLP to help analyze documents, making legal research faster and easier.

The Judiciary is also aware of the benefits that AI can bring to itself, and the National Council of Justice (CNJ) is promoting initiatives in this sense for the Judiciary as a whole (CNJ, 2019). Among the projects already developed, we can highlight the Victor project (PEIXOTO, 2020), of the Federal Supreme Court (STF), which separates and classifies the pieces of the judicial process and identifies the main themes of general repercussion; and the Socrates project, of the Superior Court of Justice (STJ), which acts in the classification of cases and will also produce an automated examination of the appeal and the judgment appealed, providing similar cases to support the rapporteur.

Recently, the Courts of Brazil started to use electronic judicial process systems. In this scenario, where legal proceedings are being processed digitally, including the digitization of old processes that were processed physically, there is a demand for applications supported by Artificial Intelligence and Natural Language Processing to increase the automation of parts of the procedural flow and to decision support. The automations, in general, occur within the Judicial Secretariat, where the procedural flow takes place in a more operational way. As a result, cases are processed more agile at the Judicial Secretariat, returning more quickly to the Judge's Office, where the content and merits of the cases are analyzed and where decisions

Figure 2: NLP pipeline



Source: Elaborated by the author, adapted from https://spacy.io/.

are rendered, in order to promote the development of AI based Decision Support Systems for judges.

2.2 Natural Language Processing

Natural Language Processing (NLP) is an area of research and application that explores how computers can be used to understand and manipulate natural language text or speech to do useful things (CHOWDHURY, 2003). NLP is considered a branch of Artificial Intelligence (AI) and encompasses approaches that use computers to analyze, determine semantic similarity, and translate between languages. The area usually deals with written languages, but it could also be applied to speech (MARTINEZ, 2010).

The field of natural language processing is also known as computational linguistics and involves the engineering of computational models and processes to solve practical problems in understanding human languages. Work in NLP can be divided into two broad sub-areas: core areas and applications. The core areas address fundamental problems such as language modeling, which underscores quantifying associations among naturally occurring words; morphological processing, dealing with segmentation of meaningful components of words and identifying the true parts of speech of words as used; syntactic processing, or parsing, which builds sentence diagrams as possible precursors to semantic processing; and semantic processing, which attempts to distill meaning of words, phrases, and higher level components in text. The application areas involve topics such as extraction of useful information (e.g., named entities and relations), translation of text between and among languages, summarization of written works, automatic answering of questions by inferring answers, and classification and clustering of documents. Often one needs to handle one or more of the core issues successfully and apply those ideas and procedures to solve practical problems (OTTER; MEDINA; KALITA, 2021).

Tradition NLP pipeline includes text analysis capabilities as tokenization, parsing, lemmatization, stemming, part-of-speech tagging, language detection and identification of semantic relationships. As shown in Figure 2, typically, from an input text, several processing steps is executed in sequence: segmenting text into tokens (tokenization), assigning part-of-speech tags (tagging), assigning dependency labels (parsing), detecting and labeling named entities; assigning base forms of the words, etc.

Figure 3: Example of sentence processing with SpaCy library

TEXT	LEMMA	POS	TAG	DEP	SHAPE	ALPHA	STOP
Apple	apple	PROPN	NNP	nsubj	Xxxxx	True	False
is	be	AUX	VBZ	aux	xx	True	True
looking	look	VERB	VBG	ROOT	xxxx	True	False
at	at	ADP	IN	prep	xx	True	True
buying	buy	VERB	VBG	pcomp	xxxx	True	False
U.K.	u.k.	PROPN	NNP	compound	X.X.	False	False
startup	startup	NOUN	NN	dobj	xxxx	True	False
for	for	ADP	IN	prep	xxx	True	True
$	$	SYM	$	quantmod	$	False	False
1	1	NUM	CD	compound	d	False	False
billion	billion	NUM	CD	pobj	xxxx	True	False

Source: Elaborated by the author, adapted from https://spacy.io/.

Figure 3 presents an example of input sentence processed using the SpaCy library[1] pipeline. In the example, the sentence "Apple is looking at buying U.K. startup for $1 billion" is separated in word-based tokens (TEXT) and each token is presented in its base form (LEMMA), part-of-speech tag (POS), dependency label (DEP), word shape considering alphanumeric, numbers, symbols and capital letter information (SHAPE), if is alphanumeric (ALPHA), if is considered a stop-word for the library model (STOP).

Currently, NLP is primarily a data-driven field using statistical and probabilistic computations along with machine learning. In the past, machine learning approaches such as Naïve Bayes, K-nearest Neighbors, hidden Markov models, conditional random fields, decision trees, random forests, and support vector machines were widely used. However, during the past several years, there has been a wholesale transformation, and these approaches have been entirely replaced, or at least enhanced, by neural models (OTTER; MEDINA; KALITA, 2021). More recently, attention models have become ubiquitous in NLP applications. Attention can be applied to different input parts, different representations of the same data, or different features, to obtain a compact representation of the data as well as to highlight the relevant information (GALASSI; LIPPI; TORRONI, 2021).

[1] https://spacy.io/

2.3 Neural Networks and Deep Learning

Deep learning is an approach to AI. Specifically, it is a particular kind of machine learning that achieves great power and flexibility by learning to represent the world as a nested hierarchy of concepts, with each concept defined in relation to simpler concepts, and more abstract representations computed in terms of less abstract ones. On the other hand, machine learning is an AI technique that allows computer systems to improve with experience and data, or simply learn from data. The quintessential example of a deep learning model is the feedforward deep network or Multilayer Perceptron (MLP). A multilayer perceptron is a mathematical function mapping some set of input values to output values. The function is formed by composing many simpler functions, and each function can be considered as a new representation of the input (GOODFELLOW; BENGIO; COURVILLE, 2016).

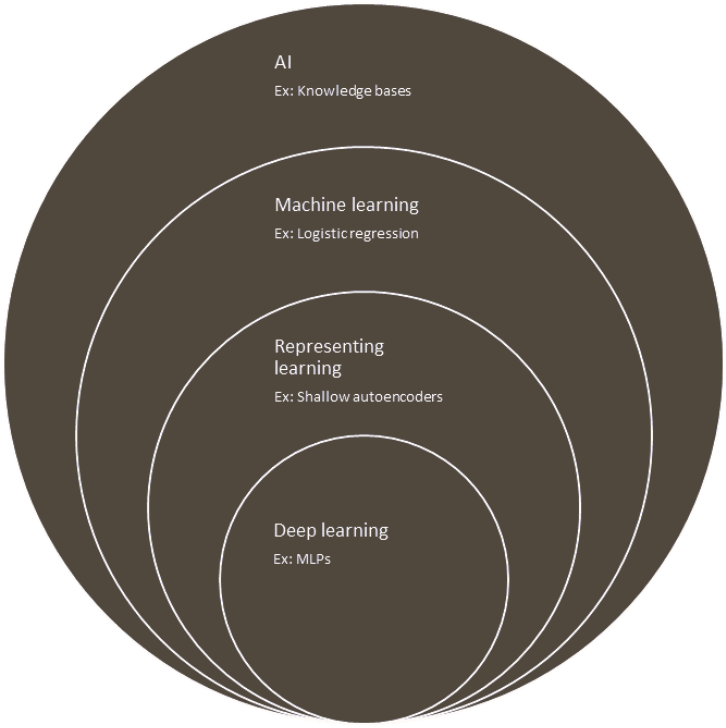
The idea of learning the right representation for the data provides one perspective on deep learning. Another perspective on deep learning is that depth allows the computer to learn a multi-step computer program. Each layer of the representation can be thought of as the state of the computer's memory after executing another set of instructions in parallel. Networks with greater depth can execute more instructions in sequence. Sequential instructions offer great power because later instructions can refer back to the results of earlier instructions (GOODFELLOW; BENGIO; COURVILLE, 2016).

There is no consensus about how much depth a model requires to qualify as "deep". However, deep learning can safely be regarded as the study of models that either involve a greater amount of composition of learned functions or learned concepts than traditional machine learning does (GOODFELLOW; BENGIO; COURVILLE, 2016). Figure 4 illustrates the relationship between different AI disciplines, showing how deep learning is a kind of representation learning, which is in turn a kind of machine learning, which is used for many but not all approaches to AI. The figure also presents an example of each AI discipline.

A high-level schematic of how each AI discipline works is illustrated in Figure 5, which shows how the different parts of an AI system relate to each other within different AI disciplines. Shaded boxes indicate components that are able to learn from data.

Some of the earliest learning algorithms were intended to be computational models of biological learning, i.e., models of how learning happens or could happen in the brain. As a result, one of the names that deep learning has gone by is Artificial Neural Networks (ANNs). The corresponding perspective on deep learning models is that they are engineered systems inspired by the biological brain. However, they are generally not designed to be realistic models of biological function. The neural perspective on deep learning is motivated by two main ideas. One idea is that the brain provides a proof by example that intelligent behavior is possible, and a conceptually straightforward path to building intelligence is to reverse engineer the computational principles behind the brain and duplicate its functionality. Another perspective is that it would be deeply interesting to understand the brain and the principles that underlie human

Figure 4: Relationship between AI disciplines



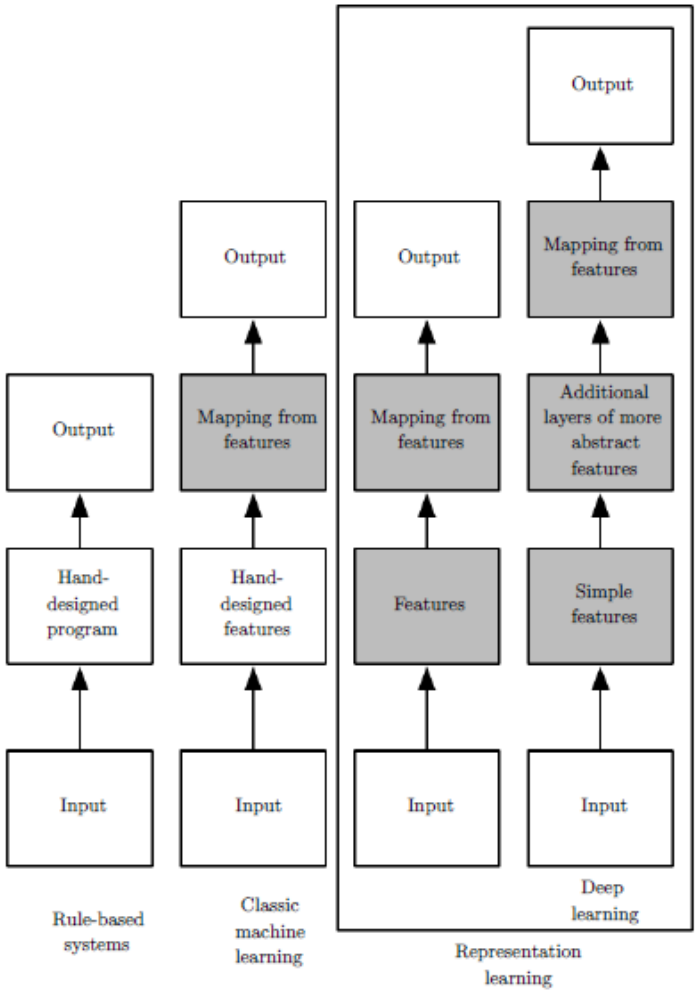
Source: Elaborated by the author, adapted from (GOODFELLOW; BENGIO; COURVILLE, 2016)

intelligence, so machine learning models that shed light on these basic scientific questions are useful apart from their ability to solve engineering applications (GOODFELLOW; BENGIO; COURVILLE, 2016).

According to (OTTER; MEDINA; KALITA, 2021), Neural Networks (NNs) are composed of interconnected nodes, or neurons, each receiving some number of inputs and supplying an output. Each of the nodes in the output layers perform weighted sum computation on the values they receive from the input nodes and then generate outputs using simple nonlinear transformation functions on these summations. While there is no clear consensus on exactly what defines a Deep Neural Network (DNN), generally networks with multiple hidden layers are considered deep and those with many layers are considered very deep (SCHMIDHUBER, 2015). Thereby, deep learning is an AI technique built from deep neural networks.

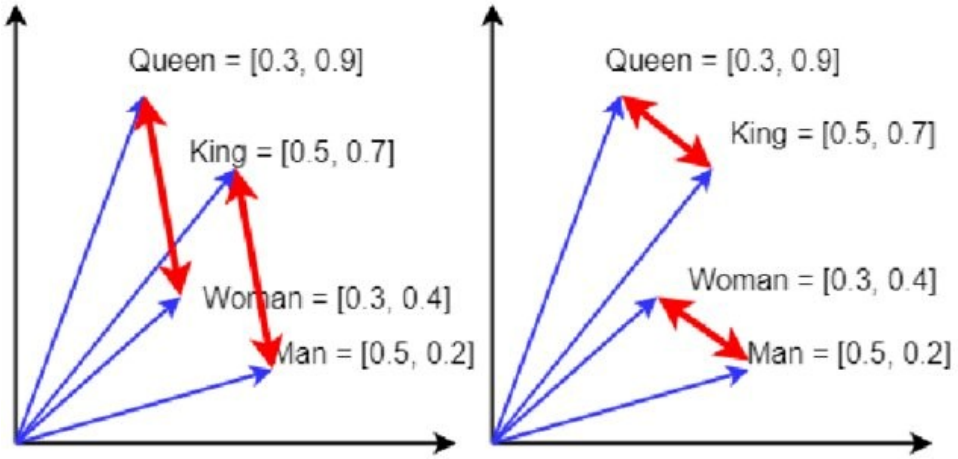
With the development of deep learning, various neural networks have been widely used to solve NLP tasks, such as Convolutional Neural Networks (CNNs) (GEHRING et al., 2017; KALCHBRENNER; GREFENSTETTE; BLUNSOM, 2014; KIM, 2014), Recurrent Neural Networks (RNNs) (LIU; QIU; HUANG, 2016; SUTSKEVER; VINYALS; LE, 2014), Graph-based Neural Networks (GNNs) (MARCHEGGIANI; BASTINGS; TITOV, 2018; SOCHER et al., 2013; TAI; SOCHER; MANNING, 2015) and attention mechanisms (BAHDANAU; CHO; BENGIO, 2016; VASWANI et al., 2017). One of the advantages of these neural models is their ability to alleviate the feature engineering problem. Non-neural NLP methods usually heavily rely on the discrete handcrafted features, while neural methods usually use low-

Figure 5: How each AI discipline works



Source: (GOODFELLOW; BENGIO; COURVILLE, 2016)

Figure 6: The classical word embeddings example $king + woman - man \approx queen$



Source: (SUTOR et al., 2019).

dimensional and dense vectors (aka. distributed representation) to implicitly represent the syntactic or semantic features of the language. These representations are learned in specific NLP tasks. Therefore, neural methods make it easy for people to develop various NLP systems (QIU et al., 2020).

2.4 Word embeddings

Word embeddings are vectors whose relative similarities correlate with semantic similarity. Such vectors are used both as an end in itself, for computing similarities between terms, for example, and as a representational basis for downstream NLP tasks. In computational linguistics, it is often called as distributional semantic model. There are also many other alternative terms in use, from the very general distributed representation to the more specific semantic vector space or simply word space.

Vector embeddings serve to embed semantic relationships into discrete points typically in high-dimensional spaces. Word embeddings are attractive due to their ability to measure similarity in the semantic sense between words and phrases by measures such as cosine similarity on the vectors and linear combinations of them. Words that are close to each other are related in a contextual sense. Word embedding models are interesting due to how well they model analogies between words, such as the famous $king + woman - man \approx queen$ example, shown in Figure 6. It must follow that $king + woman - man \approx queen$, and we can visually see based on the parallel red arrows and their direction. The similar direction of the red arrows indicates similar relational meaning (SUTOR et al., 2019).

Every feed-forward neural network that takes words from a vocabulary as input and embeds them as vectors into a lower dimensional space, which it then fine-tunes through backpropagation, necessarily yields word embeddings as the weights of the first layer, which is usually referred to as Embedding Layer (RUDER, 2016).

The main difference between such a network that produces word embeddings as a by-product and a method such as Word2Vec (MIKOLOV et al., 2013) whose explicit goal is the generation of word embeddings is its computational complexity. Generating word embeddings with a very deep architecture is simply too computationally expensive for a large vocabulary. This is the main reason why it took until 2013 for word embeddings to explode onto the NLP stage; computational complexity is a key trade-off for word embedding models (RUDER, 2016).

There are two kinds of word embeddings: non-contextual and contextual embeddings. The difference between them is whether the embedding for a word dynamically changes according to the context it appears in (QIU et al., 2020):

- **Non-contextual Embeddings**. The first step of representing language is to map discrete language symbols into a distributed embedding space. These embeddings are trained on task data along with other model parameters. There are two main limitations to this kind of embeddings. The first issue is that the embeddings are static. The embedding for a word does is always the same regardless of its context. Therefore, these non-contextual embeddings fail to model polysemous words. The second issue is the out-of-vocabulary problem. To tackle this problem, character-level word representations or sub-word representations are widely used in many NLP tasks, such as CharCNN (KIM et al., 2016), FastText (BOJANOWSKI et al., 2017) and Byte-Pair Encoding (BPE) (SENNRICH; HADDOW; BIRCH, 2016). Other examples of non-contextual embeddings are (LIU; KUSNER; BLUNSOM, 2020): (TURIAN; RATINOV; BENGIO, 2010), Word2vec (MIKOLOV et al., 2013) and Glove (PENNINGTON; SOCHER; MANNING, 2014).

- **Contextual Embeddings**. To address the issue of polysemous and the context-dependent nature of words, we need distinguish the semantics of words in different contexts. Given a text where each token is a word or sub-word, the contextual representation of each token depends on the whole text. This is also called as dynamical embedding. Are examples of contextual embeddings (LIU; KUSNER; BLUNSOM, 2020): ELMO (PETERS et al., 2018), build on a bidirectional Long Short-Term Memory (LSTM) architecture, FLAIR (AKBIK; BLYTHE; VOLLGRAF, 2018), and all the Transformer architecture models.

2.5 The Transformer

The Transformer (VASWANI et al., 2017) is a recently network architecture based solely on attention mechanisms, dispensing with recurrence and convolutions entirely, and was built with the goal of reducing sequential computation. Recurrent models typically factor computation along the symbol positions of the input and output sequences. Aligning the positions to steps in computation time, they generate a sequence of hidden states ht, as a function of the previous hidden state ht-1 and the input for position t. This inherently sequential nature precludes

parallelization within training examples, which becomes critical at longer sequence lengths, as memory constraints limit batching across examples.

Attention was first introduced in natural language processing for machine translation tasks by (BAHDANAU; CHO; BENGIO, 2016). However, the idea of glimpses had already been proposed in computer vision by (LAROCHELLE; HINTON, 2010), following the observation that biological retinas fixate on relevant parts of the optic array, while resolution falls off rapidly with eccentricity. The attention mechanism is a part of a neural architecture that enables to dynamically highlight relevant features of the input data, which, in NLP, is typically a sequence of textual elements. It can be applied directly to the raw input or to its higher-level representation. The core idea behind attention is to compute a weight distribution on the input sequence, assigning higher values to more relevant elements (GALASSI; LIPPI; TORRONI, 2021).

A key feature of Transformer models is that they are built with special layers called attention layers, which tell the model to pay specific attention to certain words in the sentence you passed it when dealing with the representation of each word. For example, consider the task of translating text from English to Portuguese. Given the input “You like this course”, a translation model will need to also attend to the adjacent word “You” to get the proper translation for the word “like”, because in Portuguese the verb “like” is conjugated differently depending on the subject. The rest of the sentence, however, is not useful for the translation of that word. In the same vein, when translating “this” the model will also need to pay attention to the word “course”, because “this” translates differently depending on whether the associated noun is masculine or feminine. Again, the other words in the sentence will not matter for the translation of “this”. With more complex sentences and more complex grammar rules, the model would need to pay special attention to words that might appear farther away in the sentence to properly translate each word. The same concept applies to any task associated with natural language: a word by itself has a meaning, but that meaning is deeply affected by the context, which can be any other word or words before or after the word being studied. The Figure 7 illustrates an example of attention visualization for an aspect-based sentiment analysis task, in which words are highlighted according to attention scores. Phrases in bold are the words considered relevant for the task or human rationales. We can see in the figure that the attention mechanism gives to each word different importance according to the context.

The Transformer architecture eschewing recurrence and instead relying entirely on an attention mechanism to draw global dependencies between input and output. The Transformer allows for significantly more parallelization and can reach a new state of the art in translation quality after being trained for as little as twelve hours on eight P100 GPUs (VASWANI et al., 2017). Figure 8 illustrates the Transformer architecture, which uses stacked self-attention and pointwise, fully connected layers for both the encoder and decoder, shown in the left and right halves, respectively.

The general functioning of the Transformer is as follows: The encoder (left half) receives an input and builds a representation of it (its features). This means that the model is optimized to

Figure 7: Example of attention visualization

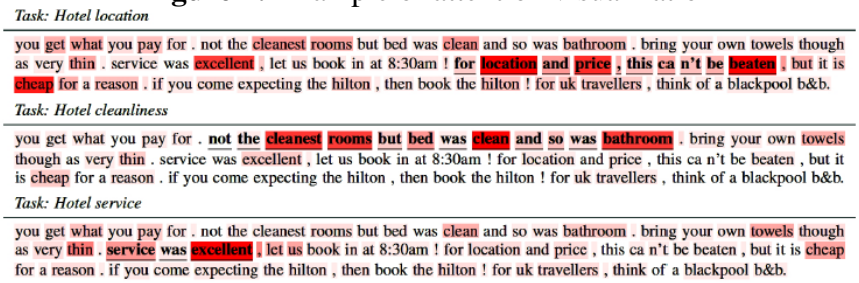


Fig. 1. Example of attention visualization for an aspect-based sentiment analysis task, from [1, Fig. 6]. Words are highlighted according to attention scores. Phrases in bold are the words considered relevant for the task or human rationales.

Source: (GALASSI; LIPPI; TORRONI, 2021)

acquire understanding from the input. The decoder (right half) uses the encoder's representation (features) along with other inputs to generate a target sequence. This means that the model is optimized for generating outputs.

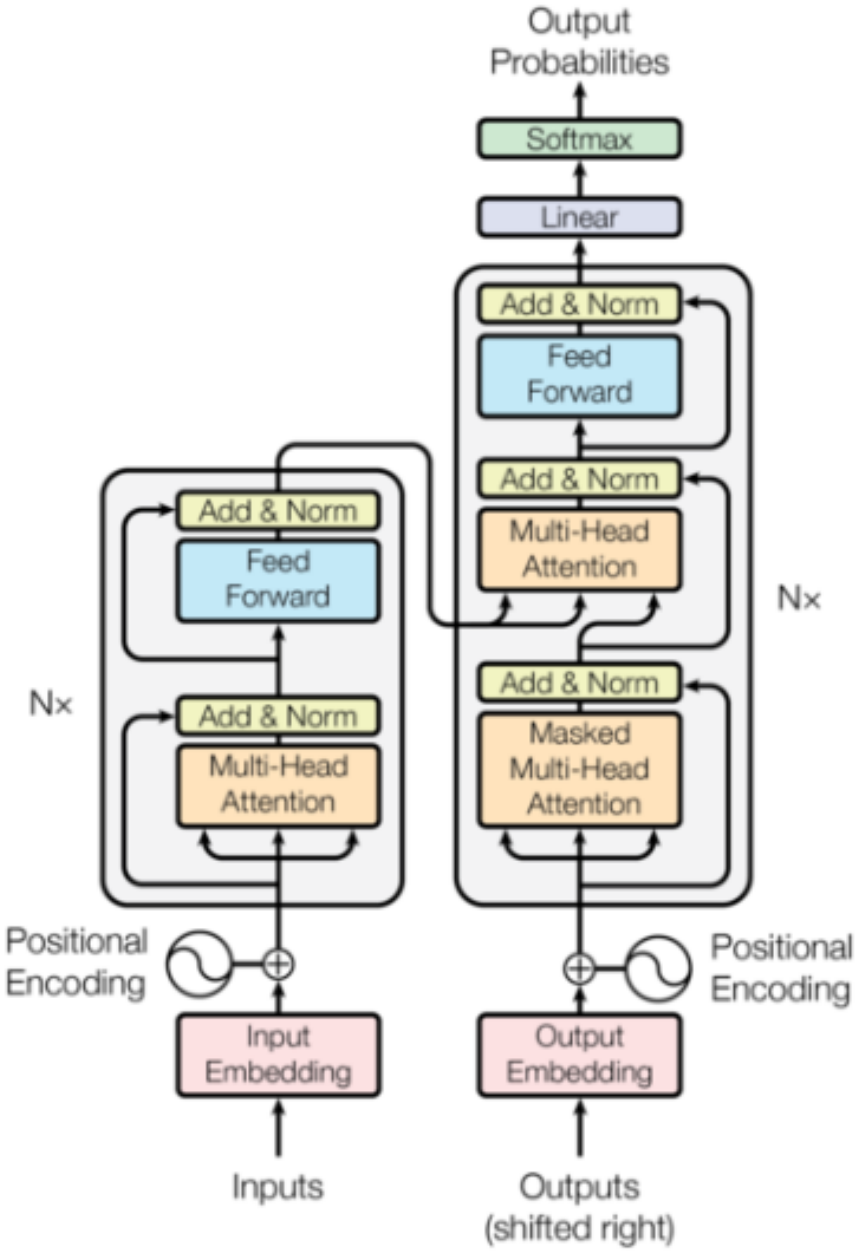
The Transformer was originally designed for translation. In that task, during the training, the encoder receives sentences in a certain language as inputs, while the decoder receives the same sentences in the desired target language. In the encoder, the attention layers can use all the words in a sentence, which is important once the translation of a given word can be dependent on what is after as well as before it in the sentence. The decoder, however, works sequentially and can only pay attention to the words in the sentence that it has already translated, i.e., only the words before the word currently being generated. For example, when we have predicted the first three words of the translated target, we give them to the decoder which then uses all the inputs of the encoder to try to predict the fourth word. The first attention layer in a decoder block pays attention to all past inputs to the decoder, but the second attention layer uses the output of the encoder. It can thus access the whole input sentence to best predict the current word. This is very useful as different languages can have grammatical rules that put the words in different orders, or some context provided later in the sentence may be helpful to determine the best translation of a given word (HUGGING Face, 2022).

Each of these half parts can be used independently, depending on the task:

- **Encoder-only models**. Good for tasks that require understanding of the input, such as sentence classification and named entity recognition.
- **Decoder-only models**. Good for generative tasks such as text generation.
- **Encoder-decoder models or sequence-to-sequence models**. Good for generative tasks that require an input, such as translation or summarization.

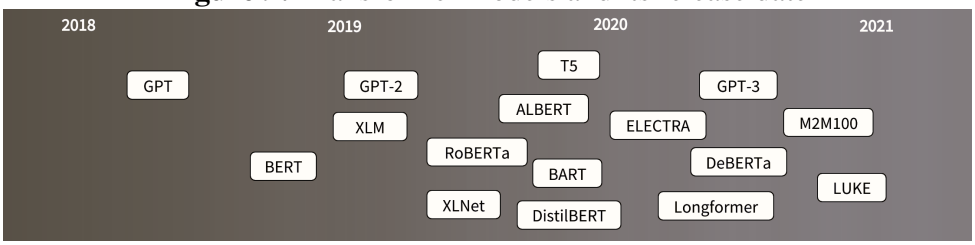
In the Transformer context, its common to see mentions of architectures, checkpoints and models. These terms all have slightly different meanings. Architecture is the skeleton of the model, the definition of each layer and each operation that happens within the model. Checkpoints are the weights that will be loaded in a given architecture. Model is an umbrella term that

Figure 8: The original Transformers architecture



Source: (VASWANI et al., 2017)

Figure 9: Transformer models and its release date



Source: Elaborated by the author, adapted from https://huggingface.co/.

isn't as precise as "architecture" or "checkpoint", it can mean both. For example, BERT is an architecture while bert-base-cased, a set of weights trained by the Google team for the first release of BERT, is a checkpoint. However, one can say "the BERT model" and "the bert-base-cased model".

Currently, a large number of Transformer models are available ("Hugging Face", 2022). A summarized list of them is briefly described below, along with their release date, also illustrated with some more models in Figure 9:

- June 2018. GPT (RADFORD et al., 2018), the first pretrained Transformer model, used for fine-tuning on various NLP tasks and obtained state-of-the-art results;
- October 2018. BERT (DEVLIN et al., 2019), large pretrained model designed to produce better summaries of sentences;
- February 2019. GPT-2 (RADFORD et al., 2019), an improved and bigger version of GPT that was not immediately publicly released due to ethical concerns;
- October 2019. DistilBERT (SANH et al., 2019), a distilled version of BERT that is 60
- October 2019. BART (LEWIS et al., 2019) and T5 (RAFFEL et al., 2019), the first two large pretrained models using the same architecture as the original Transformer model;
- May 2020. GPT-3 (BROWN et al., 2020), a bigger version of GPT-2 that is able to perform well on a variety of tasks without the need for fine-tuning (zero-shot learning).

This list is far from comprehensive and is just meant to highlight a few of the different kinds of Transformer models. Broadly, they can be grouped into three categories:

- **Autoregressive models** are pretrained on the classic language modeling task – guess the next token having read all the previous ones. They correspond to the decoder of the original transformer model, and a mask is used on top of the full sentence so that the attention heads can only see what was before in the text, and not what's after. Although those models can be fine-tuned and achieve great results on many tasks, the most natural application is text generation. A typical example of such models is GPT.

- **Autoencoding models** are pretrained by corrupting the input tokens in some way and trying to reconstruct the original sentence. They correspond to the encoder of the original transformer model in the sense that they get access to the full inputs without any mask. Those models usually build a bidirectional representation of the whole sentence. They can be fine-tuned and achieve great results on many tasks such as text generation, but their most natural application is sentence classification or token classification. A typical example of such models is BERT.

- **Sequence-to-sequence models** use both the encoder and the decoder of the original transformer, either for translation tasks or by transforming other tasks to sequence-to-sequence problems. They can be fine-tuned to many tasks, but their most natural applications are translation, summarization and question answering. The original transformer model is an example of such a model (only for translation), T5 and BART are examples that can be fine-tuned on other tasks.

2.6 Language Modeling

All the Transformer models mentioned in previous section have been trained as language models. This means they have been trained on large amounts of raw text in a self-supervised fashion. Self-supervised learning is a type of training in which the objective is automatically computed from the inputs of the model with no need of labeled data.

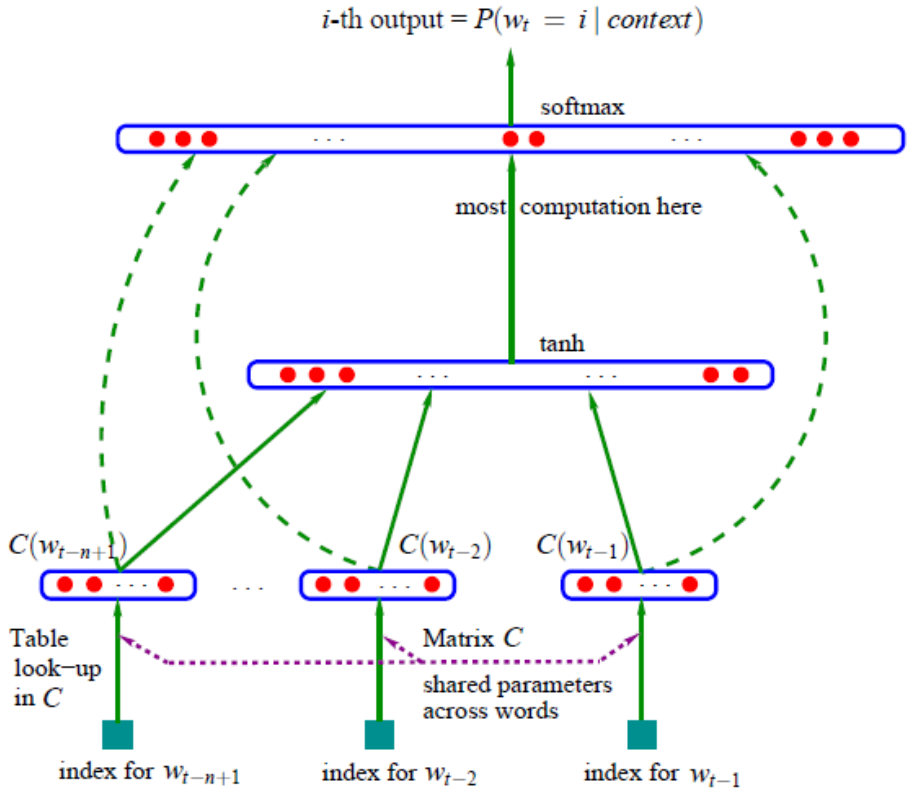
Language Modeling (LM) is an essential piece of almost any application of NLP. Language modeling is the process of creating a model to predict words or simple linguistic components given previous words or components (BENGIO et al., 2006). This is useful for applications in which a user types input, to provide predictive ability for fast text entry. However, its power and versatility emanate from the fact that it can implicitly capture syntactic and semantic relationships among words or components in a linear neighborhood, making it useful for tasks such as machine translation or text summarization. Using prediction, such programs are able to generate more relevant, human-sounding sentences (OTTER; MEDINA; KALITA, 2021).

The classic neural language model proposed by (BENGIO et al., 2003) consists of a one-hidden layer feed-forward neural network that predicts the next word in a sequence as in Figure 10. This work is one of the first to introduce what we refer to as a word embedding, a real-valued word feature vector in R. Their architecture forms very much the prototype upon which current approaches have gradually improved. The general building blocks of their model, however, are still found in all current neural language and word embedding models (RUDER, 2016):

- Embedding Layer. A layer that generates word embeddings by multiplying an index vector with a word embedding matrix;

- Intermediate Layer(s). One or more layers that produce an intermediate representation of

Figure 10: Classic neural language model



Source: (BENGIO et al., 2003)

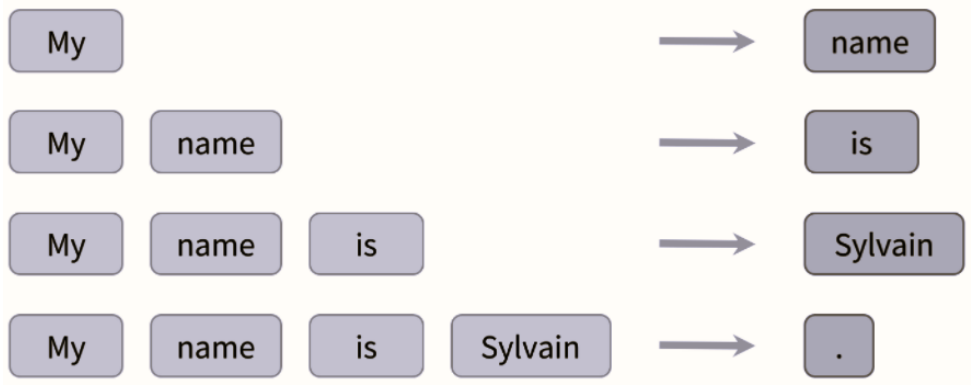
the input, e.g., a fully connected layer that applies a non-linearity to the concatenation of word embeddings of n previous words;

- Softmax Layer. The final layer that produces a probability distribution over words.

A statistical language model is a probability distribution over sequences of words. Given such a sequence of length *m*, it assigns a probability $P(w_1, \ldots, w_m)$ to the whole sequence. The language model also provides context to distinguish between words and phrases that sound phonetically similar. For example, in American English, the phrases "recognize speech" and "wreck a nice beach" sound similar but mean different things.

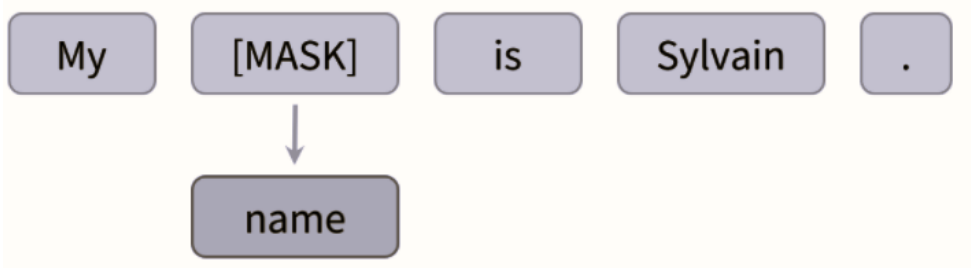
Probabilistic language modeling is the most common unsupervised task in NLP and is a classic probabilistic density estimation problem. Although LM is a general concept, in practice, it often refers in particular to auto-regressive LM or unidirectional LM. A drawback of unidirectional LM is that the representation of each token encodes only the leftward context tokens and itself. However, better contextual representations of text should encode contextual information from both directions. An improved solution is Bidirectional Language Modeling (BiLM), which consists of two unidirectional LMs: a forward left-to-right LM and a backward right-to-left LM (QIU et al., 2020). An example of a task is predicting the next word in a sentence having read the *n* previous words is presented in Figure 11. This is called *causal language modeling* because the output depends on the past and present inputs, but not the future ones.

Figure 11: Example of causal language modeling



Source: Elaborated by the author, adapted from https://huggingface.co/.

Figure 12: Example of masked language modeling



Source: Elaborated by the author, adapted from https://huggingface.co/.

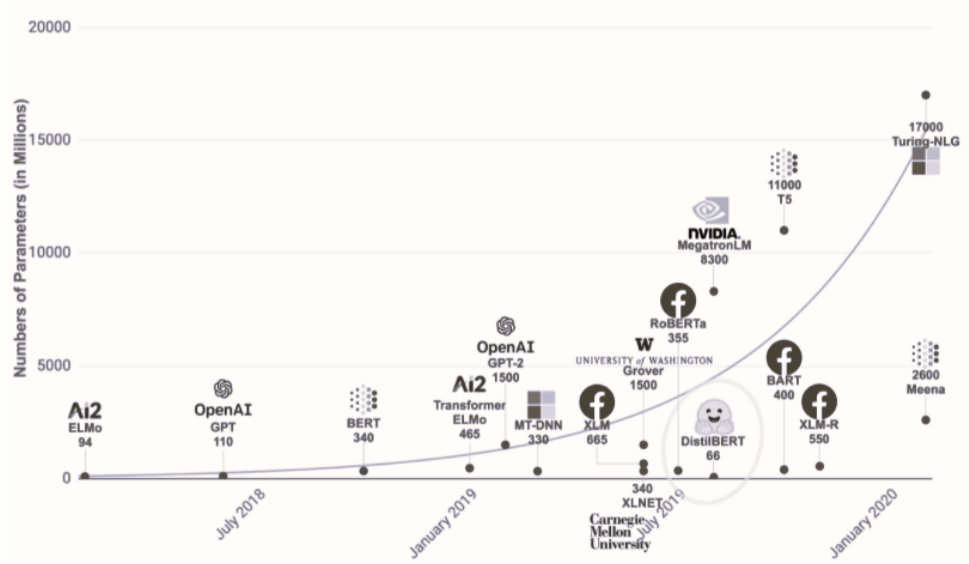
A pre-training task approach to overcome the drawback of the standard unidirectional LM is Masked Language Modeling (MLM) (DEVLIN et al., 2019), presented in Figure 12. MLM first masks out some tokens from the input sentences and then trains the model to predict the masked tokens by the rest of the tokens. However, this pre-training method will create a mismatch between the pre-training phase and the fine-tuning phase because the mask token does not appear during the fine-tuning phase. Empirically, to deal with this issue, (DEVLIN et al., 2019) used a special [MASK] token 80% of the time, a random token 10% of the time and the original token 10% of the time to perform masking (QIU et al., 2020).

Although a language model develops a statistical understanding of the language it has been trained on, it is not very useful for specific practical tasks. Because of this, the general pretrained model then goes through a process called transfer learning. During this process, the model is fine-tuned in a supervised way, with human-annotated labels, on a specific given NLP task.

Apart from a few outliers, like distilled models as DistilBERT (SANH et al., 2019), the general strategy to achieve better performance is by increasing the model's sizes as well as the amount of data they are pretrained on. Figure 13 presents a comparison of language model sizes, in which we can visualize the increase of the number of the parameters used to pretraining language models.

These accuracy improvements depend on the availability of exceptionally large computational resources that necessitate similarly substantial energy consumption. As a result, these

Figure 13: Comparison of language model sizes



Source: Elaborated by the author, adapted from https://huggingface.co/.

Table 1: Estimated CO_2 emissions from training common NLP models, compared to familiar consumption

Consumption	**CO_2e (lbs)**
Air travel, 1 passenger, roundtrip NY to SF	1.984
Human life, avg, 1 year	11.023
American life, avg, 1 year	36.156
Car, avg incl. fuel, 1 lifetime	126.000
Training one model (GPU)	
NLP pipeline (parsing, SRL)	39
w/ tuning & experimentation	78.468
Transformer (big)	192
w/ neural architecture search	626.155

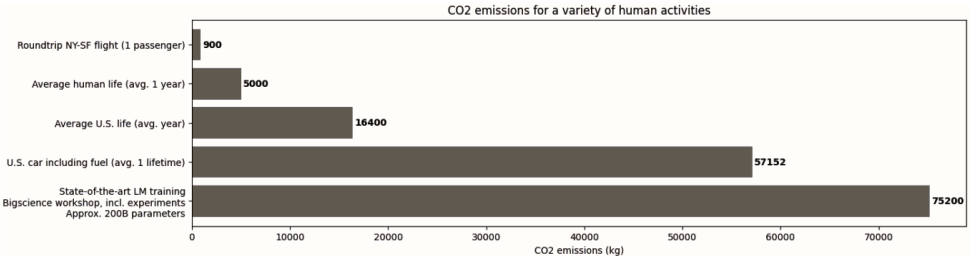
Source: Elaborated by the author, adapted from (STRUBELL; GANESH; MCCALLUM, 2019)

models are costly to train and develop, both financially, due to the cost of hardware and electricity or cloud compute time, and environmentally, due to the carbon footprint required to fuel modern tensor processing hardware (STRUBELL; GANESH; MCCALLUM, 2019). Table 1 compares the estimated CO_2 emissions from training common NLP models, compared to familiar consumption.

Similar comparison is illustrated in Figure 14, where we can clearly perceive the huge environment cost for training a 200 billion parameters language model. For comparison, the GPT-3 autoregressive language model (BROWN et al., 2020) was trained with 175 billion parameters, and Megatron-Turing NLG (SMITH; WELTY, 2001), the current world's largest and most powerful Generative Language Model was trained with 530 billion parameter.

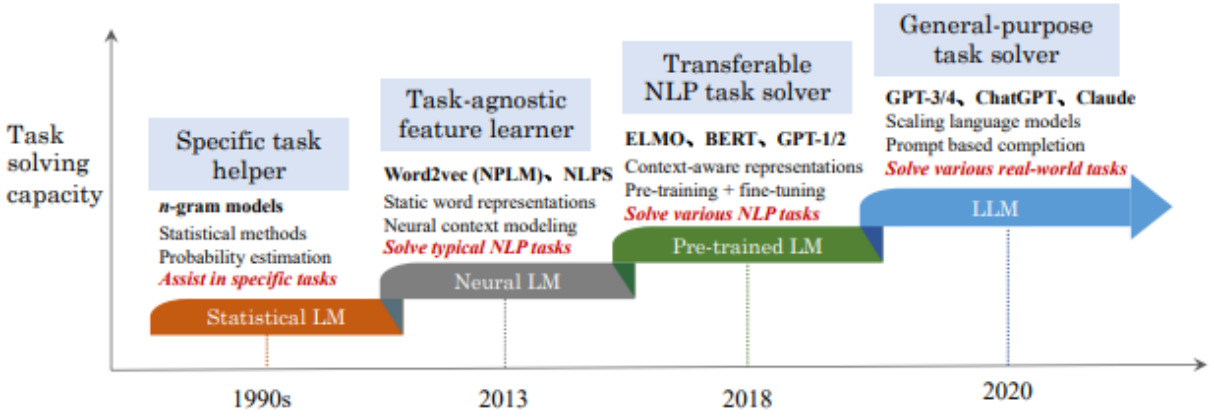
To avoid unnecessarily training language models from scratch and consequently increase the

Figure 14: Comparison of CO_2 emission with training a 200 billion parameters NLP model



Source: Elaborated by the author, adapted from https://huggingface.co/.

Figure 15: Evolution of language models



Source: (ZHAO et al., 2023).

carbon emissions, fine-tuning and transfer learning, which both reuse already-trained representations, rather than starting training of each NLP task's parameters from random initialization, are also considered as techniques to reduce the computational cost and therefore energy and carbon emissions of models (PATTERSON et al., 2021). If we consider running lots of trials to get the best hyperparameters the impact would be even higher. Thereby, sharing pretrained language models is paramount: sharing the trained weights and building on top of already trained weights reduces the overall compute cost and carbon footprint of the community.

Language modeling has been widely studied for language understanding and generation in the past two decades, evolving from statistical language models to neural language models. Figure 15 illustrates an evolution process of the four generations of language models (LM) from the perspective of task solving capacity (ZHAO et al., 2023)[2].

Recently, Pre-trained Language Models (PLMs) have been proposed by pretraining Transformer models over large-scale corpora, showing strong capabilities in solving various NLP tasks. Since the researchers have found that model scaling can lead to an improved model capacity, they further investigate the scaling effect by increasing the parameter scale to an even

[2]The time period for each stage was set mainly according to the publish date of the most representative studies at each stage. For neural language models, the authors abbreviate the paper titles of two representative studies to name the two approaches: NPLM (BENGIO et al., 2003) ("A neural probabilistic language model") and NLPS (COLLOBERT et al., 2011) ("Natural language processing (almost) from scratch").

larger size. The researchers identified that, when the parameter scale exceeds a certain level, these enlarged language models not only achieve a significant performance improvement, but also exhibit some special abilities (e.g., incontext learning) that are not present in small-scale language models (e.g., BERT (DEVLIN et al., 2019)). To discriminate the language models in different parameter scales, the research community has coined the term Large Language Models (LLMs) for the PLMs of significant size (e.g., containing tens or hundreds of billions of parameters) (ZHAO et al., 2023).

Even more recently, since researchers have found that model scaling can lead to an improved model capacity, the AI community further investigates the scaling effect by increasing the parameters of the language models to an even larger size. The researchers identified that when the parameter scale exceeds a certain level, these enlarged language models achieve a significant performance improvement and also some special abilities (e.g., in context learning) that are not present in small-scale language models (e.g., BERT (DEVLIN et al., 2019)). To discriminate the language models in different parameter scales, the research community has coined the term Large Language Models (LLMs) for the PLMs of significant size (e.g., containing tens or hundreds of billions of parameters) (ZHAO et al., 2023). Although the original Transformer architecture is composed of an encoder and a decoder (VASWANI et al., 2017) and each of these half parts can be used independently, all the new LLMs are typically decoder-only models, which are good for generative tasks such as text generation. Because of this, new AI models based on LLMs are also commonly known as Generative AI.

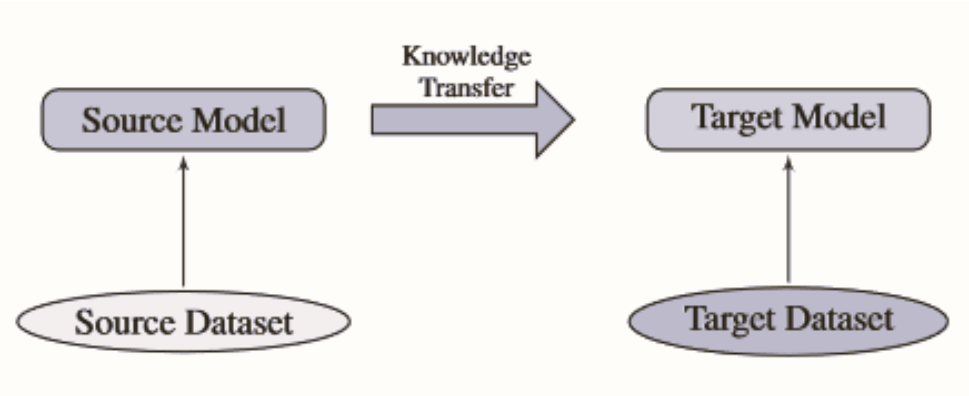
2.7 Transfer Learning

Although Pre-trained Language Models (PLMs) capture the general language knowledge from a large corpus, how effectively adapting their knowledge to the downstream task is still a key problem (QIU et al., 2020). Transfer learning (PAN; YANG, 2010) is to adapt the knowledge from a source task (or domain) to a target task (or domain). Figure 16 gives an illustration of transfer learning.

Transfer learning allows the domains, tasks, and distributions used in training and testing to be different. The study of transfer learning is motivated by the fact that people can intelligently apply knowledge learned previously to solve new problems faster or with better solutions. For example, we may find that learning to recognize apples might help to recognize pears. Similarly, learning to play the electronic organ may help facilitate learning the piano. Figure 17 shows the difference between the learning processes of traditional and transfer learning techniques, where traditional machine learning techniques try to learn each task from scratch, while transfer learning techniques try to transfer the knowledge from some previous tasks to a target task when the latter has fewer high-quality training data (PAN; YANG, 2010).

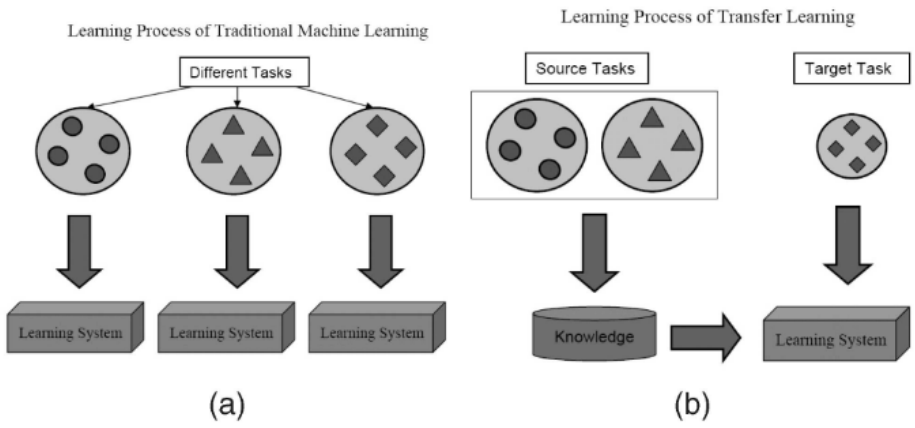
Pretraining is the act of training a model from scratch: the weights are randomly initialized, and the training starts without any prior knowledge. This pretraining is usually done on very

Figure 16: Transfer learning



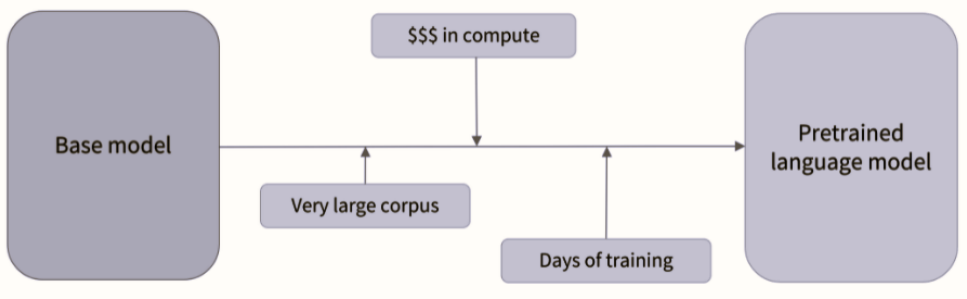
Source: Elaborated by the author, adapted from (QIU et al., 2020).

Figure 17: Different learning processes between (a) traditional machine learning and (b) transfer learning



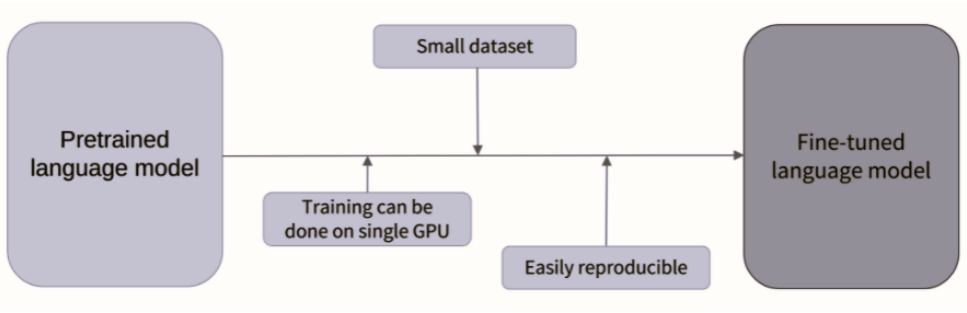
Source: (PAN; YANG, 2010).

Figure 18: Flow for training a language model



Source: Elaborated by the author, adapted from https://huggingface.co/.

Figure 19: Flow for fine-tuning a language model



Source: Elaborated by the author, adapted from https://huggingface.co/.

large amounts of data. Therefore, it requires a very large corpus of data, and training can take up from days to several weeks, as illustrated in Figure 18.

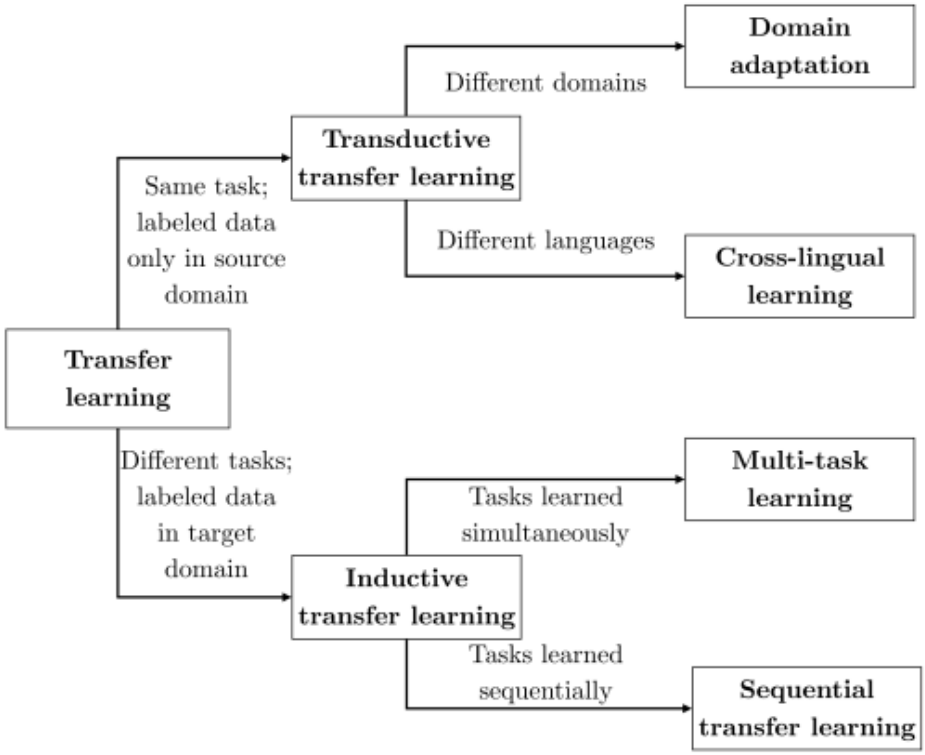
As shown in Figure 19, fine-tuning, on the other hand, is the training done after a model has been pretrained. Fine-tuning a model has lower time, data, financial, and environmental costs. It is also quicker and easier to iterate over different fine-tuning schemes, as the training is less constraining than a full pretraining. This process will also generally achieve better results than training from scratch. Thereby, we should always try to leverage a pretrained model, as close as possible to the downstream task, and fine-tune it.

There are many types of transfer learning in NLP, such as domain adaptation, cross-lingual learning, multi-task learning and sequential transfer-learning. As shown in Figure 20, (RUDER, 2019) proposes a taxonomy for transfer learning for NLP that adapts the taxonomy of (PAN; YANG, 2010) to contemporary settings in NLP, and gives more details about theses types of transfer learning.

Adapting PLMs to downstream tasks is **sequential transfer learning** task, in which tasks are learned sequentially and the target task has labeled data. According to (QIU et al., 2020), to transfer the knowledge of a PLM to the downstream NLP tasks, we need to consider the following issues:

a) **Choosing appropriate pre-training task, model architecture and corpus**. Different PLMs usually have different effects on the same downstream task, since these PLMs are

Figure 20: A taxonomy for transfer learning for NLP



Source: (RUDER, 2019).

trained with various pre-training tasks, model architecture, and corpora. For example, although BERT helps with most natural language understanding tasks, it is hard to generate language. The data distribution of the downstream task should be approximate to PLMs. Currently, there are a large number of off-the-shelf PLMs[3], which can just as conveniently be used for various domain-specific or language-specific downstream tasks.

b) **Choosing appropriate layers**. Given a pre-trained deep model, different layers should capture different kinds of information, such as POS tagging, parsing, long-term dependencies, semantic roles, coreference. For transformer-based PLMs, (TENNEY; DAS; PAVLICK, 2019) found BERT represents the steps of the traditional NLP pipeline: basic syntactic information appears earlier in the network, while high-level semantic information appears at higher layers. There are three ways to select the representation:

- *Embedding Only*. One approach is to choose only the pre-trained static embeddings, while the rest of the model still needs to be trained from scratch for a new target task. They fail to capture higher-level information that might be even more useful. Word embeddings are only useful in capturing semantic meanings of words, but we also need to understand higher-level concepts like word sense.
- *Top Layer*. The most simple and effective way is to feed the representation at the

[3]Hugging Face is an example of pre-trained models' repository (https://huggingface.co/models)

top layer into the task-specific model.

- *All Layers*. A more flexible way is to automatic choose the best layer in a soft version, like ELMo (PETERS et al., 2018).

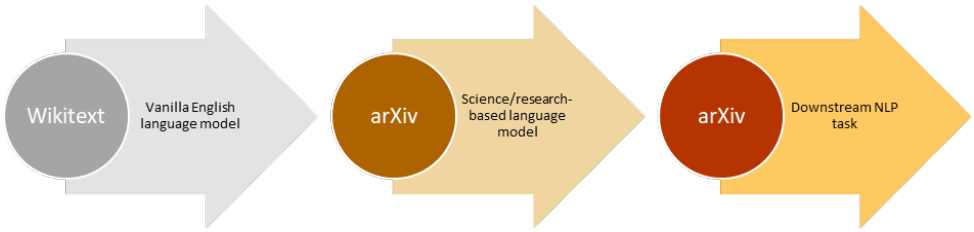
c) **Choosing to tune or not to tune the PLM**. Currently, there are two common ways of model transfer: feature extraction (where the pre-trained parameters are frozen), and fine-tuning (where the pre-trained parameters are unfrozen and fine-tuned). In feature extraction way, the pre-trained models are regarded as off-the-shelf feature extractors. Moreover, it is important to expose the internal layers as they typically encode the most transferable representations (DODGE et al., 2020). Although both these two ways can significantly benefit most of NLP tasks, feature extraction way requires more complex task-specific architecture. Therefore, the fine-tuning way is usually more general and convenient for many different downstream tasks than feature extraction way.

According to (QIU et al., 2020), with the increase of the depth of PLMs, the representation captured by them makes the downstream task easier. Therefore, the task-specific layer of the whole model is simple. Since ULMFit (HOWARD; RUDER, 2018) and BERT (DEVLIN et al., 2019), fine-tuning has become the main adaption method of PLMs. However, the process of fine-tuning is often brittle: even with the same hyper-parameter values, distinct random seeds can lead to substantially different results (DODGE et al., 2020).

Besides standard fine-tuning, some fine-tuning strategies are emerging. **Two-stage fine-tuning** introduces an intermediate stage between pretraining and fine-tuning. In the first stage, the PLM is transferred into a model fine-tuned by an intermediate task or corpus. In the second stage, the transferred model is fine-tuned to the target task. (SUN et al., 2019) showed that the "further pre-training" on the related-domain corpus can further improve the ability of BERT and achieved state-of-the-art performance on eight widely studied text classification datasets. (PHANG; FéVRY; BOWMAN, 2019) and (GARG; VU; MOSCHITTI, 2020) introduced the intermediate supervised task related to the target task, which brings a large improvement for BERT, GPT, and ELMo. (LI; DING; LIU, 2019) also used a two-stage transfer for the story ending prediction. The proposed TransBERT (transferable BERT) can transfer not only general language knowledge from large-scale unlabeled data but also specific kinds of knowledge from various semantically related supervised tasks.

In many NLP applications involving pre-trained language models, we can simply take a pre-trained model and fine-tune it directly on the data for the task at hand. Provided that the corpus used for pretraining is not too different from the corpus used for fine-tuning, transfer learning will usually produce good results. However, in some cases further pre-training will produce better results. For example, in case of using a dataset that contains legal contracts or scientific articles, a vanilla Transformer model like BERT will typically treat the domain-specific words in the corpus as rare tokens, and the resulting performance may be less than satisfactory. By fine-tuning the language model on intra-domain data, we can boost the performance of many

Figure 21: Example of domain adaptation



Source: Elaborated by the author.

downstream tasks, which means we usually only have to do this step once for the same domain. This process of fine-tuning a pretrained language model on in-domain data is usually called **domain adaptation**. For example, one could leverage a pretrained model trained on the English language and then fine-tune it on an arXiv corpus, resulting in a science/research-based model, and then use this domain adapted model to many downstream NLP tasks in the same domain, as shown in Figure 21.

2.8 Large Language Models

Typically, Large Language Models (LLMs) refer to Transformer language models that contain hundreds of billions (or more) of parameters[4], which are trained on massive text data (SHANAHAN, 2024), such as GPT-4 (OPENAI et al., 2024), PaLM (CHOWDHERY et al., 2024), Galactica (TAYLOR et al., 2022), and LLaMA (TOUVRON et al., 2023). LLMs exhibit strong capacities to understand natural language and solve complex tasks (via text generation) (ZHAO et al., 2023).

Recently, the research on LLMs has been largely advanced by both academia and industry, and a remarkable progress is the launch of ChatGPT[5] (a powerful AI chatbot developed based on LLMs), which has attracted widespread attention from society. The technical evolution of LLMs has been making an important impact on the entire AI community, which would revolutionize the way how we develop and use AI algorithms (ZHAO et al., 2023).

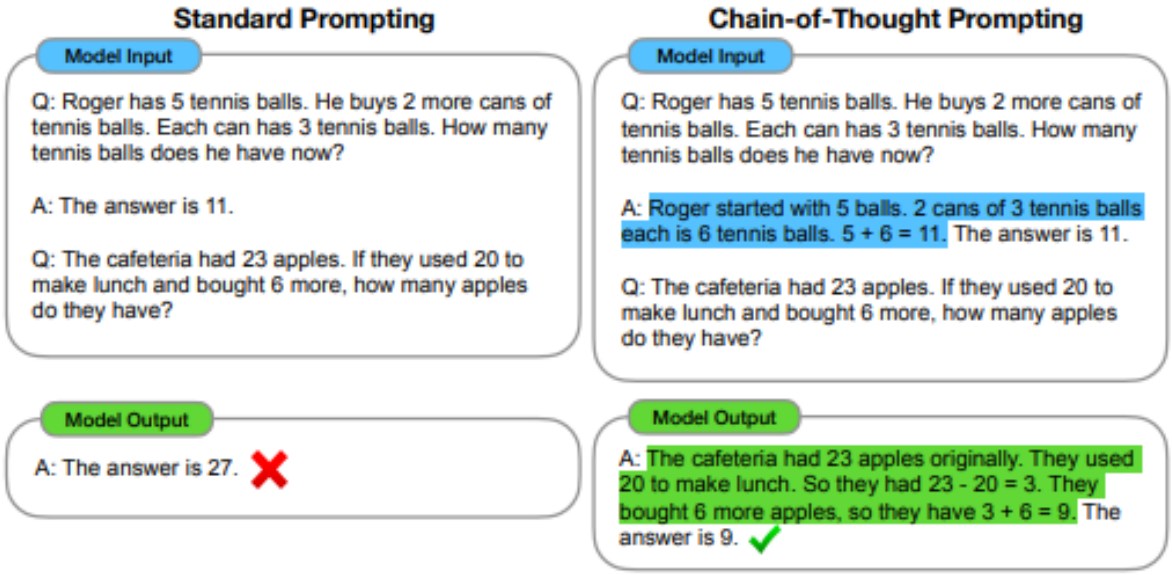
2.8.1 Prompt Engineering

The way a human interacts with LLMs is through prompts, which are natural language instructions sent to the LLM, from which the LLM sends its "response", in effect, completing the sent instruction. The same LLM can give responses in a more or less satisfactory way depending on the prompt that interacted with it. The technique of optimization of the language in a prompt to interact with an LLM in order to elicit the best possible performance is called prompt

[4] In existing literature, there is no formal consensus on the minimum parameter scale for LLMs, since the model capacity is also related to data size and total compute. (ZHAO et al., 2023)

[5] https://openai.com/index/chatgpt/

Figure 22: Example of Chain-of-Thought prompting



Source: (WEI et al., 2022).

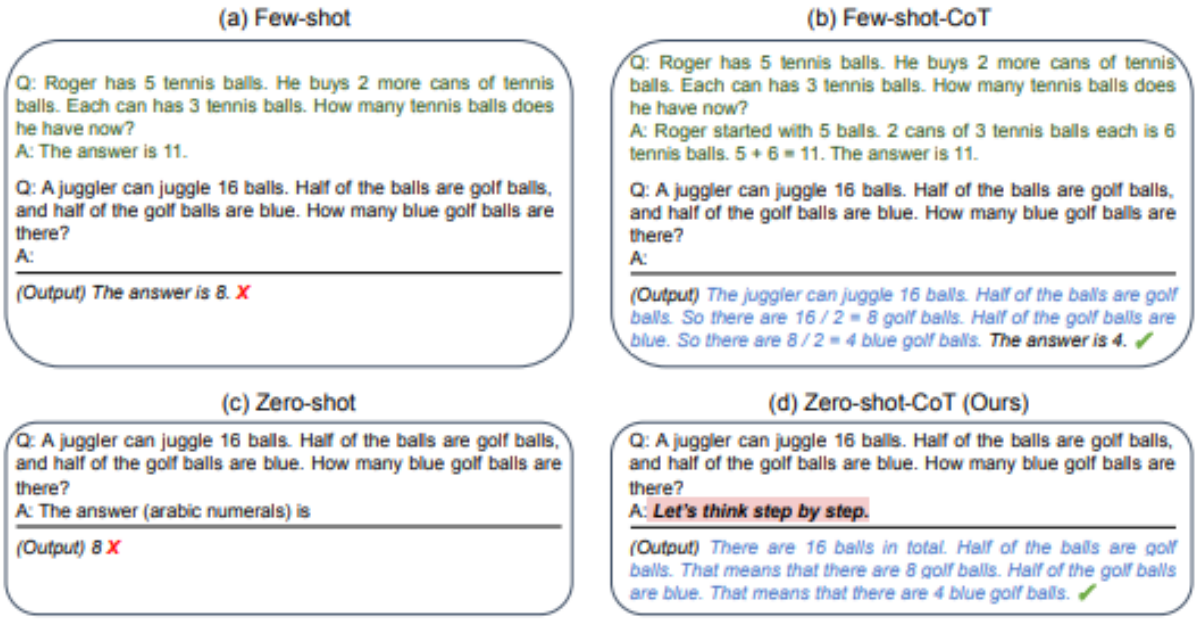
engineering. Therefore, following the release of LLMs, prompt engineering emerges as a new field of Artificial Intelligence research. We highlight some well-known prompt engineering techniques that have proven their effectiveness.

Chain-of-thought (CoT) (WEI et al., 2022) prompting enables LLMs to tackle complex arithmetic, commonsense, and symbolic reasoning tasks, by providing a simple series of intermediate reasoning steps. The authors explored how generating a chain of thought significantly improves the ability of large language models to perform complex reasoning, in particular, showing how such reasoning abilities emerge naturally in sufficiently large language models via a simple method called chain-of-thought prompting, where a few chain of thought demonstrations are provided as exemplars in prompting. In Figure 22, Chain-of-Thought reasoning processes are highlighted.

(KOJIMA et al., 2023) proposes Zero-shot-CoT, a zero-shot template-based prompting for chain of thought reasoning. It differs from the original chain of thought prompting (WEI et al., 2022) as it does not require step-by-step few-shot examples, and it differs from most of the prior template prompting (LIU et al., 2023a) as it is inherently task-agnostic and elicits multi-hop reasoning across a wide range of tasks with a single template. The core idea of the method is simple, as described in Figure 23: just adding Let's think step by step to extract step-by-step reasoning.

In practice, **Manual-CoT** (WEI et al., 2022) has obtained stronger performance than **Zero-Shot-CoT** (KOJIMA et al., 2023)]. However, this superior performance hinges on the hand-drafting of effective demonstrations. Specifically, the hand-drafting involves nontrivial efforts in designs of both questions and their reasoning chains for demonstrations (ZHANG et al., 2022).

Figure 23: Example of Zero-shot-CoT prompting



Source: (KOJIMA et al., 2023).

To eliminate such manual designs, (ZHANG et al., 2022) proposes **Auto-CoT** paradigm to automatically construct demonstrations with questions and reasoning chains. Specifically, Auto-CoT leverages LLMs with the "Let's think step by step" prompt to generate reasoning chains for demonstrations one by one, i.e., *let's think not just step by step, but also one by one*.

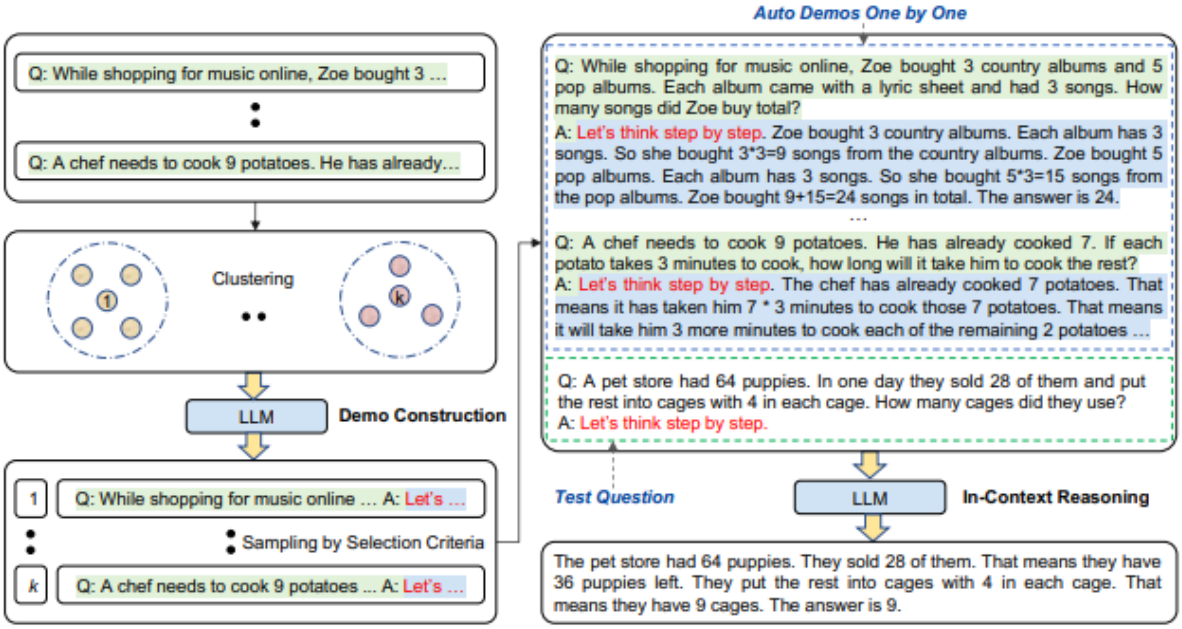
Auto-CoT consists of two main stages: (i) question clustering: partition questions of a given dataset into a few clusters; (ii) demonstration sampling: select a representative question from each cluster and generate its reasoning chain using Zero-Shot-CoT with simple heuristics. The overall procedure is illustrated in Figure 24, where demonstrations (on the right) are automatically constructed one by one (total: k) using an LLM with the "Let's think step by step" prompt.

Other automatic prompt engineering approach was proposed by (ZHOU et al., 2023). Inspired by classical program synthesis and the human approach to prompt engineering, the authors propose Automatic Prompt Engineer (APE) for automatic instruction generation and selection The method treat the instruction as the "program," optimized by searching over a pool of instruction candidates proposed by an LLM in order to maximize a chosen score function. To evaluate the quality of the selected instruction, the zero-shot performance of another LLM following the selected instruction was evaluated.

2.8.2 Retrieval-Augmented Generation

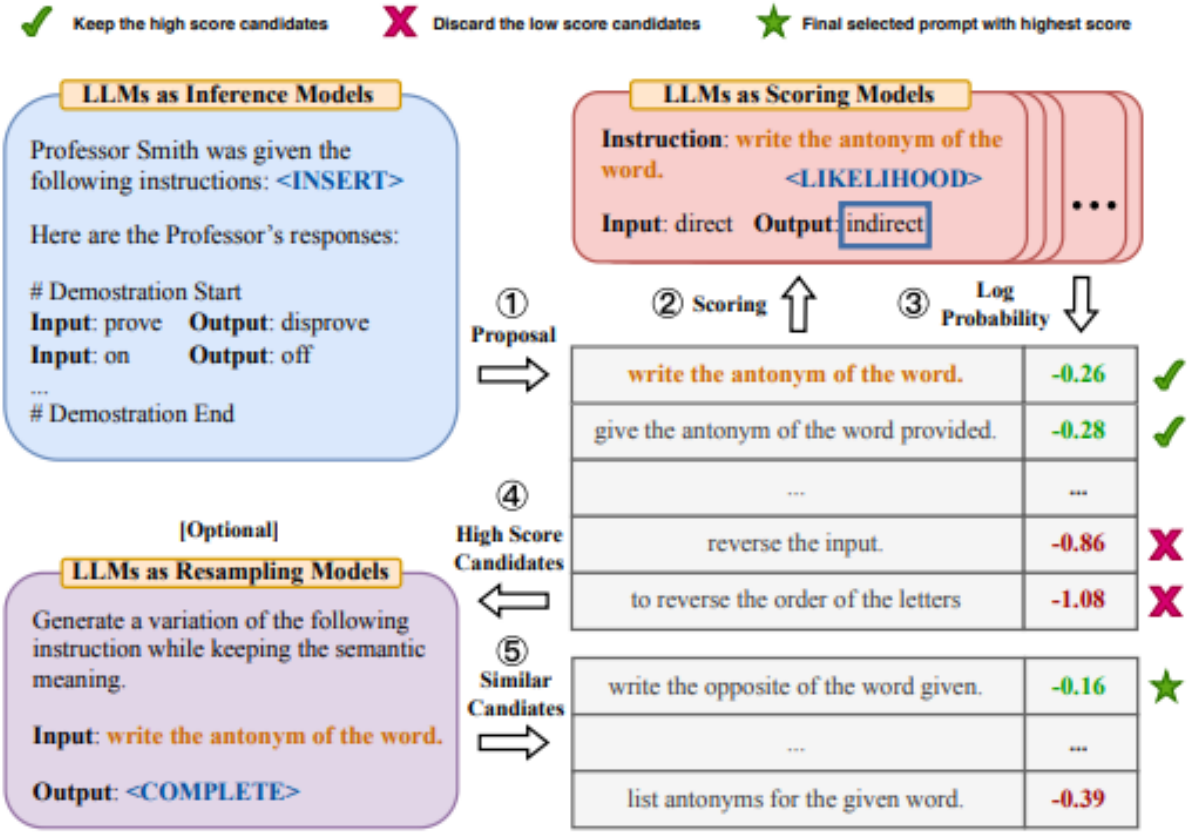
Although LLMs have demonstrated impressive capabilities, they have some challenges like hallucination, outdated knowledge, and non-transparent, untraceable reasoning processes. Retrieval-Augmented Generation (RAG) has emerged as a promising solution by incorporating

Figure 24: Overview of the Automatic Chain-of-Thought Prompting method



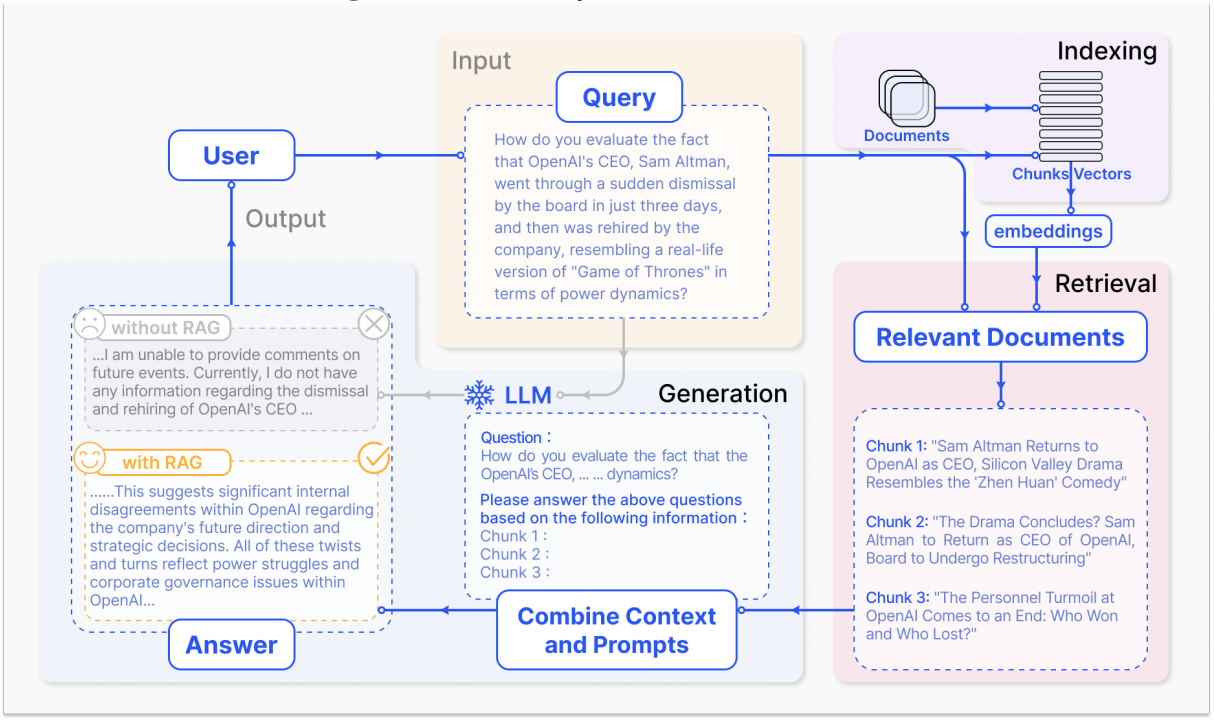
Source: (ZHANG et al., 2022).

Figure 25: Automatic Prompt Engineer (APE) workflow: APE automatically generates several instruction candidates for a task that is specified via output demonstrations, either via direct inference or a recursive process based on semantic similarity, executes them using the target model, and selects the most appropriate instruction based on computed evaluation scores.



Source: (ZHOU et al., 2023).

Figure 26: Summary of how the RAG works



Source: (GAO et al., 2024).

knowledge from external data sources. This enhances the accuracy and credibility of the generation, particularly for knowledge-intensive tasks, and allows for continuous knowledge updates and integration of domain-specific information (GAO et al., 2024).

Figure 26 illustrates the summary of how the RAG works. Here, a user poses a question to ChatGPT about a recent, widely discussed news. Given ChatGPT's reliance on pre-training data, it initially lacks the capacity to provide updates on recent developments. RAG bridges this information gap by sourcing and incorporating knowledge from external databases. In this case, it gathers relevant news articles related to the user's query. These articles, combined with the original question, form a comprehensive prompt that empowers LLMs to generate a well-informed answer (GAO et al., 2024).

2.9 Parameter Efficient Fine-Tuning

By fine-tuning a Pre-trained Language Model (PLM), we can transfer learning from a foundation model to a specific context or a specific downstream NLP task. However, the enormous size and computational demands of Large Language Models (LLMs) pose significant challenges for adapting them to specific downstream tasks, especially in environments with limited computational resources. Parameter Efficient Fine-Tuning (PEFT) offers an effective solution by reducing the number of fine-tuning parameters and memory usage while achieving comparable performance to full fine-tuning.

Full fine-tuning of transformer-based PLMs involves training the entire model, including

all layers and parameters, on a specific downstream task using task-specific data. During full fine-tuning, the PLM is initialized with pretrained weights and subsequently trained on context-specific or task-specific data, where all model parameters, including pretrained weights, are updated. Notably, full fine-tuning necessitates substantial computational resources and labeled data, as the model is trained from scratch for the specific target context or task. Moreover, as PLMs grow in size and with the advent of LLMs containing billions of parameters, full fine-tuning places even greater demands on computational resources. In contrast, PEFT methods aim to alleviate these requirements by selectively updating or modifying specific parts of the PLMs while still achieving performance comparable to full fine-tuning (XU et al., 2023).

There are several strategies of parameter efficient fine-tuning. Some of these methods are called as soft prompt-based fine-tuning (XU et al., 2023). A prompt can describe a task or provide an example of a task you want the model to learn. Instead of manually creating these prompts, soft prompting methods add learnable parameters to the input embeddings that can be optimized for a specific task while keeping the pretrained model's parameters frozen. This makes it both faster and easier to finetune large language models (LLMs) for new downstream tasks. The following are examples of prompt-based methods: Prompt-tuning (LESTER; ALRFOU; CONSTANT, 2021), Prefix-tuning (LI; LIANG, 2021) and P-tuning (LIU et al., 2023b).

Another examples are the methods $(IA)^3$ (LIU et al., 2022), low-rank decomposition methods like LoRA (HU et al., 2021) and QLoRA (DETTMERS et al., 2023), among others.

The letter "Q" from QLoRA comes from quantization. Quantization represents data with fewer bits, making it a useful technique for reducing memory-usage and accelerating inference.QLoRA is a method that quantizes a model to 4-bits and then trains it with LoRA.

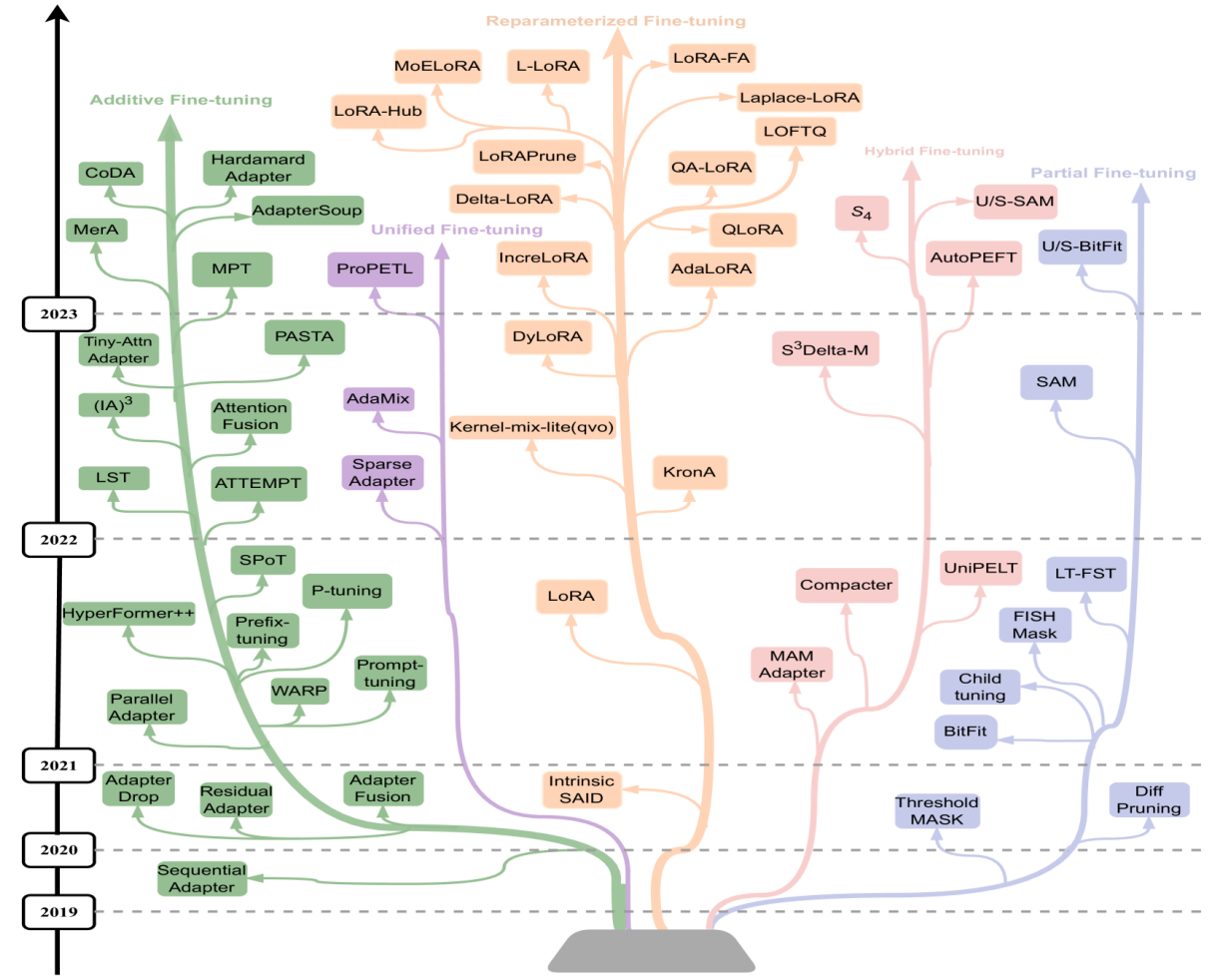
Other PEFT methods to be highlighted are the different types of adapters (XU et al., 2023) and model merging techniques (YU et al., 2024; YADAV et al., 2023), which offers a solution by combining multiple pretrained models into one model, giving it the combined abilities of each individual model without any additional training.

A more comprehensive list of PEFT methods can be found in Xu et al. (2023) and is illustrated by Figure 27.

2.10 LLM as a Judge

As traditional evaluation metrics for generative models, such as BLEU (PAPINENI et al., 2002) and ROUGE (LIN, 2004), only capture limited aspects of a model's performance (MATHUR; BALDWIN; COHN, 2020), recently, there has been a growing trend of utilizing Large Language Model (LLM) to evaluate the quality of other LLMs. Also, while human evaluation is the gold standard for assessing human preferences, but it is exceptionally slow and costly, because these models are often trained with Reinforcement Learning from Human Feedback (RLHF), they already exhibit strong human alignment, some research has proposed to be a surrogate for humans in a human evaluation. These initiatives have been called as LLM-as-a-Judge

Figure 27: The evolutionary development of PEFT methods in recent years



Source: (XU et al., 2023).

Figure 28: The default prompt for pairwise comparison (ZHENG et al., 2023)

```
[System]
Please act as an impartial judge and evaluate the quality of the responses provided by two
AI assistants to the user question displayed below. You should choose the assistant that
follows the user's instructions and answers the user's question better. Your evaluation
should consider factors such as the helpfulness, relevance, accuracy, depth, creativity,
and level of detail of their responses. Begin your evaluation by comparing the two
responses and provide a short explanation. Avoid any position biases and ensure that the
order in which the responses were presented does not influence your decision. Do not allow
the length of the responses to influence your evaluation. Do not favor certain names of
the assistants. Be as objective as possible. After providing your explanation, output your
final verdict by strictly following this format: "[[A]]" if assistant A is better, "[[B]]"
if assistant B is better, and "[[C]]" for a tie.

[User Question]
{question}

[The Start of Assistant A's Answer]
{answer_a}
[The End of Assistant A's Answer]

[The Start of Assistant B's Answer]
{answer_b}
[The End of Assistant B's Answer]
```

Source: (ZHENG et al., 2023).

(LI et al., 2024; ZHENG et al., 2023; OPENAI, 2024).

By defining evaluation schemes in the prompt template, LLMs can leverage their instruction-following ability to provide reliable evaluation. For example, (LI et al., 2024) constructed a test set containing 805 questions and used the win rate compared with text-davinci-003 as the evaluation result, which is determined by GPT4. (ZHENG et al., 2023) developed 80 multi-round test questions covering eight common areas, and then automatically scored the model's answers using GPT4. Their results reveal that strong LLM-based evaluators can achieve a high agreement rate among human experts, establishing a foundation for LLM-as-a-Judge (HUANG et al., 2024).

Zheng et al. (2023) propose 3 LLM-as-a-judge variations:

- **Pairwise comparison**. An LLM judge is presented with a question and two answers, and tasked to determine which one is better or declare a tie. The prompt used is given in Figure 28.

- **Single answer grading**. Alternatively, an LLM judge is asked to directly assign a score to a single answer. The prompt used for this scenario is in Figure 29.

- **Reference-guided grading**. In certain cases, it may be beneficial to provide a reference solution if applicable. An example prompt we use for grading math problems is in Figure 30.

Figure 29: The default prompt for single answer grading (ZHENG et al., 2023)

```
[System]
Please act as an impartial judge and evaluate the quality of the response provided by an
AI assistant to the user question displayed below. Your evaluation should consider factors
such as the helpfulness, relevance, accuracy, depth, creativity, and level of detail of
the response. Begin your evaluation by providing a short explanation. Be as objective as
possible. After providing your explanation, please rate the response on a scale of 1 to 10
by strictly following this format: "[[rating]]", for example: "Rating: [[5]]".

[Question]
{question}

[The Start of Assistant's Answer]
{answer}
[The End of Assistant's Answer]
```

Source: (ZHENG et al., 2023).

Figure 30: The prompt for reference-guided pairwise comparison (ZHENG et al., 2023)

```
[System]
Please act as an impartial judge and evaluate the quality of the responses provided by two
AI assistants to the user question displayed below. Your evaluation should consider
correctness and helpfulness. You will be given a reference answer, assistant A's answer,
and assistant B's answer. Your job is to evaluate which assistant's answer is better.
Begin your evaluation by comparing both assistants' answers with the reference answer.
Identify and correct any mistakes. Avoid any position biases and ensure that the order in
which the responses were presented does not influence your decision. Do not allow the
length of the responses to influence your evaluation. Do not favor certain names of the
assistants. Be as objective as possible. After providing your explanation, output your
final verdict by strictly following this format: "[[A]]" if assistant A is better, "[[B]]"
if assistant B is better, and "[[C]]" for a tie.

[User Question]
{question}

[The Start of Reference Answer]
{answer_ref}
[The End of Reference Answer]

[The Start of Assistant A's Answer]
{answer_a}
[The End of Assistant A's Answer]

[The Start of Assistant B's Answer]
{answer_b}
[The End of Assistant B's Answer]
```

Source: (ZHENG et al., 2023).

These methods can be implemented independently or in combination and have different pros and cons. For example, the pairwise comparison may lack scalability when the number of players increases, given that the number of possible pairs grows quadratically; single answer grading may be unable to discern subtle differences between specific pairs, and its results may become unstable, as absolute scores are likely to fluctuate more than relative pairwise results if the judge model changes.

The authors also identifies that LLM-as-a-judge offers two key benefits: scalability and explainability. It reduces the need for human involvement, enabling scalable benchmarks and fast iterations. Additionally, LLM judges provide not only scores but also explanations, making their outputs interpretable. Although they also identify certain biases and limitations of LLM judge like position bias, verbosity bias and self-enhancement bias, the authors present solutions and show the agreement between LLM judges and humans is high despite these limitations.

2.11 Explainable AI

Recent successes in machine learning have led to a new wave of artificial intelligence applications that offer extensive benefits to a diverse range of fields. However, many of these systems are not able to explain their autonomous decisions and actions to human users. Explanations may not be essential for certain AI applications, and some AI researchers argue that the emphasis on explanation is misplaced, too difficult to achieve, and perhaps unnecessary. However, for many critical applications in defense, medicine, finance, and law, explanations are essential for users to understand, trust, and effectively manage these new, artificially intelligent partners (GUNNING et al., 2019). According to (ADADI; BERRADA, 2018), in life-changing decisions such as disease diagnosis, it is important to know the reasons behind such a critical decision. Here, the crucial need for explaining AI outcomes becomes fully apparent.

These recent AI successes are largely attributed to machine learning algorithms that construct models in their internal representations. Although these models exhibit high performance in terms of results and predictions, AI algorithms suffer from opacity, that it is difficult to get insight into their internal mechanism of work, especially machine learning algorithms, i.e., they are opaque in terms of explainability. Which further compound the problem, because entrusting important decisions to a system that cannot explain itself presents obvious dangers. Furthermore, there may be inherent conflict between machine learning performance and explainability. Often, the highest performing methods (e.g., deep learning) are the least explainable, and the most explainable (e.g., decision trees) are the least accurate (ADADI; BERRADA, 2018; GUNNING et al., 2019).

To address this issue, eXplainable Artificial Intelligence (XAI) proposes to make a shift towards more transparent AI. It aims to create a suite of techniques that produce more explainable models whilst maintaining high performance levels (ADADI; BERRADA, 2018). The purpose of a XAI system is to make the behavior of AI more intelligible to humans by providing expla-

nations. According to (GUNNING et al., 2019), there are some general principles to help create effective, more human-understandable AI systems: the XAI system should be able to explain its capabilities and understandings; explain what it has done, what it is doing now, and what will happen next; and disclose the salient information that it is acting on.

However, every explanation is set within a context that depends on the task, abilities, and expectations of the user of the AI system. The definitions of interpretability and ex- plainability are, thus, domain dependent and may not be defined independently from a domain. Explanations can be full or partial. Models that are fully interpretable give full and completely transparent explanations. Models that are partially interpretable reveal important pieces of their reasoning process. Partial explanations may include variable importance measures, local models that approximate global models at specific points and saliency maps (GUNNING et al., 2019).

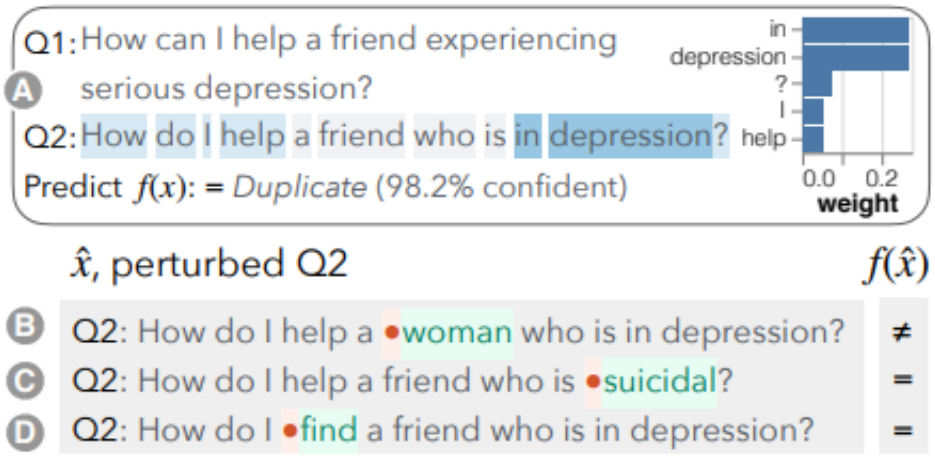
Explainability is essential for users to effectively understand, trust, and manage powerful artificial intelligence applications. This concern is beginning to be provided in legislation, as is the case in Brazil, with Resolution No. 615/2025 of the National Council of Justice (CNJ, 2025), which "disposes about ethics, transparency and governance in production and use of Artificial Intelligence in the Judiciary". The resolution specifically cites rules on the construction of artificial intelligence systems for the Brazilian judiciary. Regarding explainability, the resolution requires systems to "provide a satisfactory explanation that can be audited by human authority regarding any proposed decision presented by the Artificial Intelligence model".

A popular way of explaining NLP models is to attribute importance weights to the input tokens, either using attention scores (WIEGREFFE; PINTER, 2019) or by summarizing the model behavior on perturbed instances, e.g., LIME (RIBEIRO; SINGH; GUESTRIN, 2016) and SHAP (LUNDBERG; LEE, 2017).

Though ubiquitous, token scores may not always reflect their real importance (PRUTHI et al., 2021). Popular packages like LIME (Local Interpretable Model-agnostic Explanations) or SHAP (SHapley Additive exPlanation) estimate scores by masking words, and therefore may not reflect model behavior on natural counterfactual cases. For example, the Figure 31 presents an instance of duplicate question detection in Quora Question Pairs (QQP) dataset. In the figure, (A) is an instance in QQP where the model prediction f(x) is Duplicate (=) at 98.2% confidence, with SHAP importance weights for tokens in Q2. Counterfactual explanations complement SHAP with concrete examples and surprising behaviors, e.g., (B) shows that changing friend for woman surprisingly flips the prediction to NonDuplicate, despite the low weight on "friend" (WU et al., 2021).

Complementing the explanation presented in the example from Figure 31, the token "friend" in (A) is not considered important even though a natural substitution in (B) flips the prediction. The opposite happens to "in depression", where a significant change makes no difference to the model's prediction (C). Even perfect importance scores may be too abstract for users to gain real understanding, e.g., users may not grasp the significance of a low importance score for the

Figure 31: Instance of duplicate question detection in QQP



Source: Elaborated by the author, adapted from (WU et al., 2021).

token "help" without concrete examples such as the one in (D).

To improve understanding, Polyjuice (WU et al., 2021) and works like (MADAAN et al., 2021) and (ROSS; MARASOVIć; PETERS, 2020) produces counterfactual explanations, providing significant insight on top of state-of-the-art explanation techniques, automatically generating examples that change predictions for explanation or analysis, with no constraints on semantics. On the other hand, adversarial generators, which aim to maintain semantics while changing model predictions, were proposed by (RIBEIRO; SINGH; GUESTRIN, 2018), (IYYER et al., 2018) and (LI et al., 2021).

As examples of eXplainable AI frameworks there are the well-known SHAP (LUNDBERG; LEE, 2017) and LIME (RIBEIRO; SINGH; GUESTRIN, 2016). Other exames are ELI5[6], a Python package which helps to debug machine learning classifiers and explain their predictions; and Captum[7], a model interpretability and understanding library for PyTorch. containing general purpose implementations of integrated gradients, saliency maps, smoothgrad, vargrad and others for PyTorch models.

In the context of transformer-based architectures, there are exBERT (HOOVER; STROBELT; GEHRMANN, 2020), a visual analysis tool to explore learned representations in transformer models; and Transformers Interpret[8], a model explainability tool designed to work exclusively with the Huggingface transformers package, allowing any transformers model to be explained in just two lines, supporting visualizations in both notebooks and as savable html files.

[6]https://github.com/TeamHG-Memex/eli5
[7] https://github.com/pytorch/captum
[8]https://github.com/cdpierse/transformers-interpret

2.12 Ontologies

According to (NOY; MCGUINNESS, 2001), an ontology is a formal explicit description of concepts in a domain of discourse (classes; sometimes called concepts), properties of each concept describing various features and attributes of the concept (slots; sometimes called roles or properties), and restrictions on slots (facets; sometimes called role restrictions). An ontology together with a set of individual instances of classes constitutes a knowledge base.

Ontologies try to structure human knowledge, how concepts are created and related, presenting a formal view of knowledge. In Computer Science this is materialized in studies where knowledge base structures are developed focusing on structured knowledge sharing (SMITH; WELTY, 2001).

An ontology defines a common vocabulary for sharing information in a domain. It includes machine-interpretable definitions of basic concepts in the domain and relations among them. Some of the reasons to develop an ontology are (NOY; MCGUINNESS, 2001):

- To share common understanding of the structure of information among people or software agents;
- To enable reuse of domain knowledge;
- To make domain assumptions explicit;
- To separate domain knowledge from the operational knowledge;
- To analyze domain knowledge.

The domain of Law contains numerous related concepts, so ontologies can play an important role to describe these concepts in a machine-understandable way, once ontologies are descriptions of the world as human beings relate things. In the area of Computer Science, there are efforts to model human knowledge and distribute this information in ontology so that they can be interpreted in a computational environment. Legal ontologies are a specific type to deal with modeling of norms, regulations, functions, and roles, coming from the legal area (SPEROTTO; BELCHIOR; AGUIAR, 2019).

(HOEKSTRA et al., 2007) described a legal core ontology that is part of a generic architecture for legal knowledge systems, which would enable the interchange of knowledge between existing legal knowledge systems. The proposed Legal Knowledge Interchange Format (LKIF) has two main roles: 1) the translation of legal knowledge bases written in different representation formats and formalisms and 2) a knowledge representation formalism that is part of a larger architecture for developing legal knowledge systems.

Works in legal ontologies in Portuguese are being conducted for some time. (SAIAS; QUARESMA, 2002) proposed a methodology to semi-automatically transform a traditional web information retrieval system into a semantic aware one, also describing an application of

this methodology to the legal web information retrieval system of the Portuguese Attorney General's Office. (SAIAS; QUARESMA, 2005) proposed a methodology for applying NLP techniques to automatically create a legal ontology in a logic programming information retrieval system. Ontology applied in the judicial sentences was worked by (CASTRO et al., 2017), who introduced a semantic methodology using ontology in order to improve results of data mining in judicial decisions database, with an intelligent and automatic method to search for sentences in lawsuits related to the one in trial. (ARAUJO; RIGO; BARBOSA, 2017) presented a system for ontology-based information extraction from natural language texts, able to identify a set of legal events, based on domain ontology of legal events and a set of linguistic rules. (SPEROTTO; BELCHIOR; AGUIAR, 2019) developed a model of legal ontology based of Brazilian law and middleware that can assist the various platforms of agents in recognizing and interpreting the legal information of these ontologies.

Likewise, there are already works relating the use of ontologies with the transformer-based architectures, more specifically BERT. For example, (LIU; PERL; GELLER, 2020) proposed a method to automatically predict the presence of IS-A relationships between a new concept and pre-existing concepts based on the language representation model BERT; (HE et al., 2021) proposed a novel ontology alignment[9] system based on BERT, aiming to sufficiently utilize the text semantics implied by ontologies; (NEUTEL; BOER, 2021) examined the automatic alignment of two occupation ontologies using NLP methods using BERT; BERTMap (HE et al., 2022) is a BERT-based ontology alignment system.

[9]Ontology alignment refers to matching semantically related entities from different ontologies, with the vision of integrating data from heterogeneous resources (HE et al., 2021).

3 RELATED WORK

"If I have seen further, it is by standing on the shoulders of giants."
Isaac Newton

This chapter presents related work on the application of Artificial Intelligence to the legal domain in Portuguese. Section 3.1 describes a systematic mapping study on AI-based decision support systems for judges, conducted with the objective of identifying research gaps and opportunities in the intersection between Artificial Intelligence and Law, with a particular focus on decision support systems within the Judiciary.

In addition to the systematic review, a broader survey of related initiatives in the Brazilian context was carried out, encompassing the use of AI in the Judiciary, as well as the development of datasets, language models, and other relevant resources. These contributions are described in the following sections.

It is important to highlight that these investigations were carried out during the early stages of the doctoral research. Consequently, some parts may reflect the technological landscape of that period, which has evolved considerably with the rapid advancements in natural language processing, especially with the emergence and widespread adoption of large language models and generative AI systems. Nevertheless, the analyses remain valid in capturing the state of the art at the time and in identifying foundational trends and challenges in the field.

3.1 AI based Decision Support Systems for judges: a systematic mapping study

This section presents a Systematic Mapping Study on decision support systems to support judges, intending to identify the most relevant studies developed in the area in the last five years, seeking state of the art. This work also focuses on the main techniques, metrics, development platforms used, and results obtained by these works, in order to map the research area, allowing to identify research gaps and opportunities.

3.1.1 Introduction

Since the Dartmouth Conference officially proposed the concept of "Artificial Intelligence" in 1956, Artificial Intelligence (AI) has been transforming human society. In recent years, AI has accelerated this development, bringing new features such as deep learning, human-machine collaboration, open intelligence, and autonomous control (ZHANG et al., 2019a). The application of Artificial Intelligence in the legal field went through more than 60 years of development, having emerged from the publication of the article "*Automation in The Legal World*", by Lucien Mehl, in 1958. In 1963, Lawlor assumed that computers one day would become able to analyze and predict the results of court decisions (LAWLOR, 1963). However, only with the emergence

of the *International Conference of Artificial Intelligence and Law* (ICAIL) in 1987, there are regular conferences on the topic, and the first ICAIL can be considered as the birth of the IA Community and Law (BENCH-CAPON et al., 2012).

According to (RISSLAND; ASHLEY; LOUI, 2003), the union of AI and Law is a relevant field for research in Artificial Intelligence. Challenging problems are posed to AI and Law, demanding work on central issues of AI such as reasoning, representation, and learning. Further expanding the research horizon, according to (ROBALDO et al., 2019), Law has always been an attractive domain for linguistic and semantic technologies, since it is essential for governance, therefore fostering state of the art in Natural Language Processing (NLP).

The first AI projects in Law focused on knowledge representation and legal systems based on rules. However, since the 2000s, the use of AI in Law has shifted towards approaches based on machine learning. Currently, the use of AI in Law is conceptually divided into three categories (SURDEN, 2019): Law practitioners, Law administrators, and Law users. For law administrators (government, judges, legislators, police, etc.), AI can be used to support decision making, in making sentences, and to guide decisions for criminal defendants, for example.

Support for decision making for judges refers to the development of models for the prediction of trial results, the identification of most relevant case aspects regarding its outcome, the recommendation of similar cases, verification of legal documents completeness, better routing and triage of cases based on likely results and duration or apparent complexity (BRANTING et al., 2018; ZHANG et al., 2019b).

Recently, advances in Artificial Intelligence, especially in Natural Language Processing and Deep Learning, have provided tools to automatically analyze legal texts in order to create predictive models for judicial results based on the semantics of Law and cases texts. On the other hand, previous work on court decisions' prediction has focused primarily on non-textual information (LIU; CHEN, 2017). Research on the prediction of US Supreme Court justices' votes has revealed that the incorporation of information from legal texts can surpass predictive models that use only quantitative legal data (KATZ; II; BLACKMAN, 2017; KAUFMAN; KRAFT; SEN, 2017; SIM; ROUTLEDGE; SMITH, 2014). Approaches to unravel determinant patterns for judicial decisions can rely on the information extracted from the text of the cases. Some researchers investigate the correlation between the cases' results and the facts of the cases based only on the textual content, suggesting that judicial decision-making is significantly affected by the aspects of related facts (ALETRAS et al., 2016; LIU; HSIEH, 2006; SULEA et al., 2017). More recently, deep learning models based on neural networks have achieved excellent text classification results (ZHANG et al., 2019b). Besides, the use of ontologies (DAS; ROY; MAJUMDAR, 2020; EL GHOSH et al., 2017; FAWEI et al., 2018; KURCHEEVA et al., 2019; SPEROTTO; BELCHIOR; AGUIAR, 2019) is also being used to improve the construction of models based on legal texts.

Therefore, based on these considerations, it is possible to identify the importance of conducting a comprehensive study to map possible gaps and opportunities for research in the area

of Artificial Intelligence and Law, more specifically in decision support systems for judges. In this respect, an option that is receiving attention recently is the systematic mapping of the literature. Systematic Mapping Studies (SMS) are designed to provide a broad overview of a research area, to establish whether research evidence exists on a topic and to provide an indication of the amount of evidence available in literature. The results of a mapping study can identify areas suitable for conducting Systematic Literature Reviews (SLR) and also areas where a primary study is most appropriate (KITCHENHAM; CHARTERS, 2007). As a consequence, both methods can and should be used in a complementary way. A systematic mapping can be conducted first to obtain an overview of the area, and then the state of evidence on specific topics can be investigated using a systematic review (PETERSEN et al., 2008).

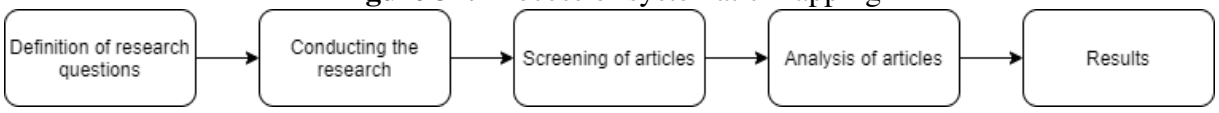
This paper presents a Systematic Mapping Study on decision support systems to support judges, intending to identify the most relevant studies developed in the area in the last five years, seeking state of the art. This work also focuses on the main techniques, metrics, development platforms used, and results obtained by these works, in order to map the research area, allowing to identify research gaps and opportunities.

This article is structured as follows: Section 2 presents the methodology used to carry out the systematic mapping study and the details of its use for this review; section 3 presents the individual analysis of the objectives, results, and conclusions of the selected articles; section 4 presents the results and discussion of the systematic mapping study, considering the research questions that originated the study; finally, in section 5, the conclusions obtained with this work are presented, identifying gaps and research opportunities in the area of AI and Law, more specifically of decision support systems for judges.

3.1.2 Methodology

A thorough review of the literature on a topic provides a basis for advancing knowledge and identifies existing research and areas where research is needed. Two important methodologies for conducting a literature review are Systematic Literature Review (SLR) and Systematic Mapping Study (SMS). Both methodologies introduce significant benefits over traditional methods of literature review, based on a rigorous and repeatable process. They provide improved accuracy, fairness, reliability and auditability of method and review results. In general, both are three-phase review methods that include planning, execution, and synthesis. They differ mainly on the following points: how a research question is formed, how the publication corpus is explored, what is the review style, and what is the main purpose of the review result. For example, the SMS methodology focuses on broader research questions, analyzes a large number of publications, adopts a style that is not as thorough as SLR, and aims at classifying the publication leading to a high level of understanding. On the other hand, the SLR methodology focuses on precise research questions leading to accurate results by reviewing a relatively small number of publications (BARN; BARAT; CLARK, 2017).

Figure 32: Process of systematic mapping



Source: Elaborated by the author.

An SLR is a means of identifying, evaluating and interpreting all available research relevant to a specific research question, topic area, or phenomenon of interest. Individual studies that contribute to a systematic review are called primary studies; a systematic review is a form of secondary study (KITCHENHAM; CHARTERS, 2007). An SMS allows evidence in a domain to be plotted at a high level of granularity. This allows identifying evidence groups and evidence gaps to direct the focus of future systematic reviews and identify areas for conducting primary studies. In summary, SMS are used to structure a research area, while SLR are focused on the collection and synthesis of evidence (PETERSEN; VAKKALANKA; KUZNIARZ, 2015).

This section aims to present the methodology used to carry out the Systematic Mapping Study in the context of Artificial Intelligence applied to Law and Decision Support Systems to support judges. The SMS carried out in this work was adapted from the steps proposed by (PETERSEN et al., 2008), as shown in Figure 32,. The following subsections detail the steps for defining research questions, the result of which is the scope of the research; conducting the research, the result of which is all articles from primary studies; and the screening of articles, the result of which is the relevant selected articles. The stages of analysis of the articles and results of the systematic mapping are presented in specific sections.

3.1.2.1 Defining research questions

Specifying questions is the most important part of any systematic review. The review questions guide the entire systematic review methodology. The search process must identify primary studies that address research questions. The data extraction process must extract the data items necessary to answer the questions. The data analysis process must synthesize the data so that the questions can be answered (KITCHENHAM; CHARTERS, 2007). Table 2 presents the research questions for this work.

3.1.2.2 Conducting research on primary studies

After defining research questions, the next step in systematic mapping is conducting research. Primary studies are identified using search strings in scientific databases or by manually browsing through conference proceedings or publications of relevant journals.

Considering the research questions to be answered, the following search strings was used to perform the searches in the databases:

Table 2: Research questions

ID	Research questions	Motivation
RQ1	What are the main current research on Decision Support applications to support judges?	Map the state of the art of research on Artificial Intelligence-based decision support systems to support judges.
RQ2	What datasets are used?	Identify the main data sets currently used in decision support systems experiments.
RQ3	What are the main techniques used?	Identify the main algorithms and techniques applied.
RQ4	What development platforms are being used?	Identify the main development platforms used to be more assertive when choosing which one will be used.
RQ5	Are articles of the Law considered for prediction results?	Verify if and how the predictions of the results consider the articles of the Law.
RQ6	What are the metrics to perform the evaluation?	Verify which metrics are being used to evaluate the experiments.
RQ7	What are the results obtained?	Evaluate the results obtained by the revised surveys, to identify promising research paths.

Source: Elaborated by the author.

Table 3: Objectives of the terms used in the research chain

Term used	Objective
"artificial intelligence" OR "machine learning" OR "deep learning"	Select works that use some Artificial Intelligence technique
"NLP" OR "natural language processing"	Select works that use Natural Language Processing techniques
"law" OR "legal" OR "judicial" OR "judiciary"	Select specific works for the area of Law
"decision making" OR "predic* decision*" OR "predic* jud* decision*"	Select works related to decision support systems

Source: Elaborated by the author.

("artificial intelligence" OR "machine learning" OR "deep learning") AND ("NLP" OR "natural language processing") AND ("law" OR "legal" OR "judicial" OR "judiciary") AND ("decision making" OR "predic decision*" OR "predic* jud* decision*")*

Table 3 shows a relationship between the terms used in defining the research chain and its objective.

The digital libraries chosen to select the primary articles for this systematic mapping study were ACM Digital Library, IEEE Xplore, Science Direct and Springer Link, since they are the most used for publishing studies on this area (ZHANG; BABAR; TELL, 2011). In addition to these, it was also decided to use Google Scholar to reach relevant publications published in other digital libraries. Table 4 presents the scientific databases used in this work and the respective electronic addresses.

In addition to the search string, to limit the number of articles returned, the year or start

Table 4: Scientific databases used

Digital library	Electronic address
ACM Digital Library	https://dl.acm.org
IEEE Xplore	https://ieeexplore.ieee.org
Science Direct	https://www.sciencedirect.com
Springer Link	https://link.springer.com/
Google Scholar	https://scholar.google.com/

Source: Elaborated by the author.

date of the articles was configured to consider only works published from 2015. The results of the research were manipulated with the support of the Zotero reference manager software (https://www.zotero.org/).

The search at ACM was possible to be performed with the original search string and the results were saved to Zotero by the web browser connector. This is the obtained result:

251 Results for: [[All: "artificial intelligence"] OR [All: "machine learning"] OR [All: "deep learning"]] AND [[All: "nlp"] OR [All: "natural language processing"]] AND [[All: "law"] OR [All: "legal"] OR [All: "judicial"] OR [All: "judiciary"]] AND [[All: "decision making"] OR [All: "predic decision*"] OR [All: "predic* jud* decision*"]] AND [Publication Date: (01/01/2015 TO *)]*

Although the survey returned 251 records, only 236 were entered in Zotero, which disregarded 15 records that referred to conference proceedings indexes. The publications that have been removed are described in Table 5.

In the IEEE survey it was also possible to use the string originally defined search query, which returned 5 records, which were saved to Zotero by the web browser connector. This is the obtained result:

Showing 1-5 of 5 for ("artificial intelligence" OR "machine learning" OR "deep learning") AND ("NLP" OR "natural language processing") AND ("law" OR "legal" OR "judicial" OR "judiciary") AND ("decision making" OR "predic decision*" OR "predic* jud* decision*")*

Filters Applied: 2015 - 2019 (2020)

The search on ScienceDirect needed to be adjusted considering search engine restrictions: no support for wildcards and more than 8 boolean connectors per field. The execution of the research considering the search string below returned 684 results, which were exported in RIS (Research Information Systems) format and then imported into Zotero. This is the string used and result:

("artificial intelligence" OR "machine learning" OR "deep learning") AND ("NLP" OR "natural language processing") AND ("law" OR "legal" OR "judicial") AND ("decision making")

(restrições da busca: Sorry - wildcards are not supported yet; Search does not support more than 8 Boolean connectors per field)

Year: 2015-2020

The research conducted at SpringerLink also used the original search string and returned

Table 5: Publications removed from ACM

Publications removed	**Electronic address**
WWW '19: Companion Proceedings of The 2019 World Wide Web Conference	https://doi.org/10.1145/3308560
SA '17: SIGGRAPH Asia 2017 Courses	https://doi.org/10.1145/3134472
CSCW '17: Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing	https://doi.org/10.1145/2998181
SA '16: SIGGRAPH ASIA 2016 Courses	https://doi.org/10.1145/2988458
SA '16: SIGGRAPH ASIA 2016 Symposium on Visualization	https://doi.org/10.1145/3002151
FSE 2016: Proceedings of the 2016 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering	https://doi.org/10.1145/2950290
ICMI '16: Proceedings of the 18th ACM International Conference on Multimodal Interaction	https://doi.org/10.1145/2993148
Onward! 2016: Proceedings of the 2016 ACM International Symposium on New Ideas, New Paradigms, and Reflections on Programming and Software	https://doi.org/10.1145/2986012
ASE 2016: Proceedings of the 31st IEEE/ACM International Conference on Automated Software Engineering	https://doi.org/10.1145/2970276
SIGGRAPH '16: ACM SIGGRAPH 2016 Courses	https://doi.org/10.1145/2897826
POPL '16: Proceedings of the 43rd Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages	https://doi.org/10.1145/2837614
MIG '15: Proceedings of the 8th ACM SIGGRAPH Conference on Motion in Games	https://doi.org/10.1145/2822013
SA '15: SIGGRAPH Asia 2015 Visualization in High Performance Computing	https://doi.org/10.1145/2818517
ITiCSE '15: Proceedings of the 2015 ACM Conference on Innovation and Technology in Computer Science Education	https://doi.org/10.1145/2729094
SIGMOD '15: Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data	https://doi.org/10.1145/2723372

Source: Elaborated by the author.

Table 6: Number of articles selected by digital library

Digital library	Quantity
ACM Digital Library	236
IEEE Xplore	5
Science Direct	684
Springer Link	1261
Google Scholar	100
Total articles	**2286**

Source: Elaborated by the author.

1,261 records, which were saved in Zotero through the web browser connector. This is the obtained result:

1,261 Result(s) for '("artificial intelligence" OR "machine learning" OR "deep learning") AND ("NLP" OR "natural language processing") AND ("law" OR "legal" OR "judicial" OR "judiciary") AND ("decision making" OR "predic decision*" OR "predic* jud* decision*")'*

within 2015 - 2020

Finishing the research in the scientific databases, a Google Scholar search using the original search string in the search field. However, due to limitations of the search engine, the returned result did not exactly consider the logical expressions of the original search string. However, it was possible to configure the year of publication filter (2015-2020). The search resulted in "Approximately 15,900 results", and we chose to use the first 100 results by relevance, which were saved in Zotero by the web browser connector.

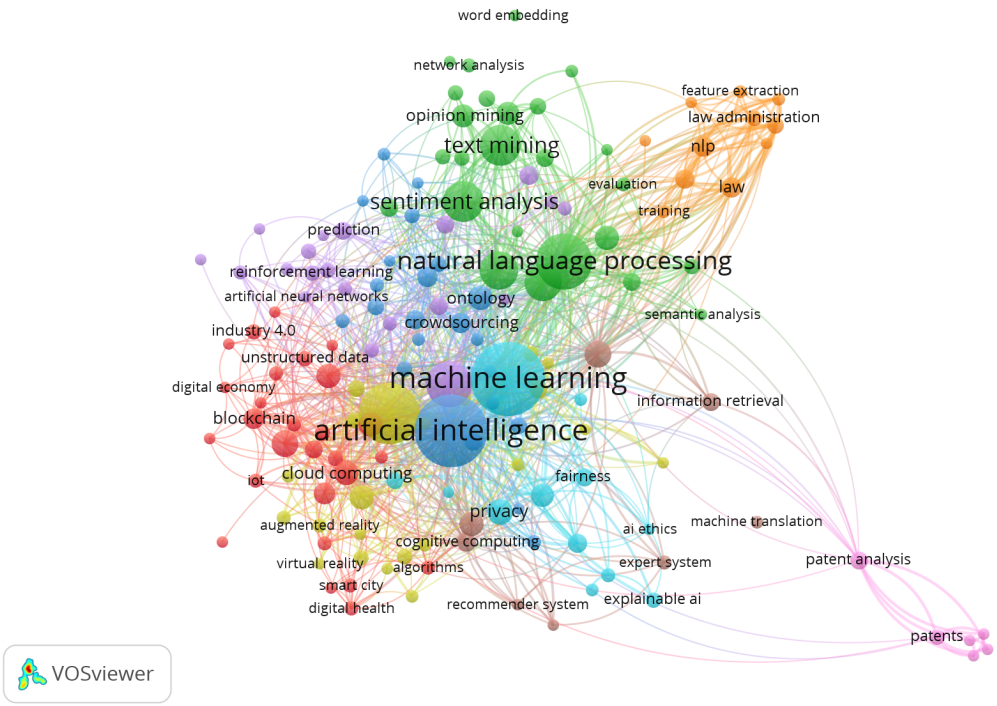
The total number of articles selected in this phase, therefore, was 2,286, which were exported in Zotero CSV format to a Google Docs spreadsheet. Table 6 shows the number of articles selected per digital library. In turn, Figure 33 illustrates a view of the keywords of the selected articles, generated by VOSviewer (https://www.vosviewer.com/).

3.1.2.3 Screening of articles

Inclusion and exclusion criteria are used to exclude studies that are not relevant to answer the research questions (BAILEY et al., 2007; MARSHALL; BRERETON, 2013; PETERSEN et al., 2008). Based on the use of inclusion and exclusion criteria applied to the set of selected primary studies, the articles to be studied in this systematic mapping were screened.

The method used for the screening of articles made, from the complete list of articles selected in the scientific databases, a sequence of exclusions, detailed in Table 7, until only the articles to be considered in this study remained. The total number of articles included (IC-1) for detailed analysis in this study was therefore 21 publications. Considering that the original research in the scientific databases used as a filter the period of publication between the years 2015 and 2020, an inclusion and exclusion criterion was not created specifically regarding the year of publication.

Figure 33: Keywords of the selected articles



Source: Elaborated by the author.

Table 7: Exclusion and inclusion criteria

Exclusion and inclusion criteria	**Quantity**
EC-1 - Works not published in English.	10
EC-2 - Duplicate publications (Publication Year, Author, Title, Publication Title).	36
EC-3 - Publications without authors or abstracts that, in general, represent pages of summaries, indexes and annals of events, journals and books.	191
EC-4 - Title and abstract do not contain at least one of the following terms (law|legal|justice|court|decision|predict|survey|deep learning|natural language processing)	965
EC-5 - Publications considered not relevant to answer the research questions, after reading the title, publication name, abstract and keywords.	1035
EC-6 - Classification of the remaining articles in the categories Legal analysis on AI in Law, Decision support, Text classification, Text extraction, Explainable AI, Ontology and Review, excluding all categories, except for Decision support.	28
IC-1 - Publications considered relevant to answer research questions, after reading the title, publication name, abstract and keywords.	21
Total articles	**2286**

Source: Elaborated by the author.

To apply the EC-4 exclusion criteria, a regular expression was used in the Title and Summary columns of the Google Docs spreadsheet. The function containing the regular expression used was as follows:

=NOT(OR(REGEXMATCH(H2;"(?i)(law|legal|justice|court|decision|predict|survey|deep learning|natural language processing)");REGEXMATCH(N2;"(?i)(law|legal|justice|court|decision|predict|survey|deep learning|natural language processing)"))) The following steps detail the complete procedure performed for sorting articles:

- Step 1. Original search in the digital libraries and storage of the results in Zotero.
- Step 2. Export the Zotero data in CSV format and then join and import the data in a Google Docs spreadsheet.
- Step 3. Ordering of articles by Title and Year.
- Step 4. Exclusion of articles that have not been published in English (EC-1).
- Step 5. Exclusion of duplicate articles considering Year of Publication, Authors, Title and Title of Publication (EC-2). Necessary since articles from Google Scholar were also used.
- Step 6. Exclusion of publications without authors or abstracts that, in general, represent pages of summaries, indexes and annals of events, magazines and books (EC-3).
- Step 7. Exclusion of publications whose title and abstract does not contain at least one of the following terms: *law, legal, justice, court, decision, predict, survey, deep learning, natural language processing* (EC-4).
- Step 8. Exclusion of publications considered not relevant to answer the research questions, after reading the title, publication name, abstract and keywords (EC-5).
- Step 9. Classification of the remaining articles, after reading the title, publication name, abstract and keywords, in the categories Legal analysis on AI in Law, Decision support, Text classification, Text extraction, Explainable AI, Ontology and Review, excluding all publications not classified as Decision Support (EC-6).
- Step 10. Inclusion of publications classified as Decision Support (IC-1).

3.1.3 Analysis of articles

This section presents an individual analysis of the articles from the selected primary studies, listed in Table 8. Objectives, results and conclusions obtained from reading the full text of the 21 articles selected in this systematic mapping study are presented.

In this study, considering that the number of articles selected was not very large, it was possible to carry out an analysis of the full text of the articles and not just the reading of the title, abstract and keywords, as is usually done, to classify the studies and generate systematic mapping. In a way, it can be considered as a literature review complementary to traditional systematic mapping.

(CHEN et al., 2019) used the CAIL (Chinese AI and Law challenge) 2018 dataset (XIAO et al., 2018) to analyze the basic case description and apply deep learning to predict the trial's outcome under the aspects: penalty charge and legal provisions. Three prediction models were built, all using Chinese word segmentation with Jieba (JUNYI, 2020): a penalty prediction model based on TF-IDF, FastText and TextCNN, a legal prediction model based on TextCNN and a charge model based on FastText and TextCNN. The constructed models can predict the judgment results according to the new facts of the case and provide a reference for obtaining results of the judgment. In general, the models performed well: the penalty prediction model reached an accuracy of 88.75% with FastText and 88.76% with TextCNN; the prediction model for legal provisions obtained an F1-score of 86.27% with TextCNN, better than the ML-KNN (Multi-Label k-Nearest Neighbor) (ZHANG; ZHOU, 2007) comparison methods with 74.38% and ML-LOC (Multi-label learning using local correlation) (HUANG; ZHOU, 2012) with 66.49%; the prosecution prediction model reached "Precision", "Recall" and "F1-score" respectively of 60.16%, 54.45% and 57.16% for FastText and 64.65%, 56.56% and 60.33% for TextCNN. The trial's expected results take into account large numbers of similar cases and provide new evidence to discuss with the prosecutor. Simultaneously, they also allow for supervision in the sentencing process and can avoid the unfairness of the trial results.

(LIU; CHEN, 2017) provided an empirical investigation comparing the predictive performance of five machine learning models: SVM (Support Vector Machine), Logistic Regression, Random Forests, Bagging (Bootstrap aggregating) and k-Nearest Neighbors (k-NN). The dataset used consisted text extracted from judgment sections of the cases of the European Court of Human Rights (European Court of Human Rights) obtained from the HUDOC ECHR database (http://hudoc.echr.coe.int), contemplating cases of violations of political and civil rights over the European Convention on Human Rights. Three sets of experiments were carried out. In all experiments, SVM outperformed the other models. As reasons for SVM's good performance, we can highlight its ability to work with small datasets, respectively 250, 80 and 254 examples for each set of experiments. Also, its better ability to work with a reasonably large number of features (2000) obtained via N-gram. This indicates that extraction techniques features can help to improve classification performance by reducing the amount of redundant or unimportant features. The experiments also demonstrated that the choice of features based on semantic information obtained by natural language processing of the text positively affects the models' performance. Besides, models built on high-level semantic features are more interpretable for humans, which is important for applicability in the real world.

(ZHANG et al., 2019b) proposed a prediction model for law articles based on Deep Pyramid

Table 8: Selected articles

ID	Reference	Title
1	(CHEN et al., 2019)	A Deep Learning Method for Judicial Decision Support
2	(LIU; CHEN, 2017)	A predictive performance comparison of machine learning models for judicial cases
3	(ZHANG et al., 2019b)	Applying Data Discretization to DPCNN for Law Article Prediction
4	(LEI et al., 2017)	Automatically Classify Chinese Judgment Documents Utilizing Machine Learning Algorithms
5	(MOK; MOK, 2019)	Classification of Breach of Contract Court Decision Sentences
6	(ZHANG et al., 2019a)	Evaluation of Judicial Imprisonment Term Prediction Model Based on Text Mutation
7	(MAHFOUZ; KANDIL; DAVLYATOV, 2018)	Identification of latent legal knowledge in differing site condition (DSC) litigations
8	(BRANTING et al., 2018)	Inducing Predictive Models for Decision Support in Administrative Adjudication
9	(SHEN et al., 2018)	Legal Article-Aware End-To-End Memory Network for Charge Prediction
10	(MARQUES et al., 2019)	Machine learning for explaining and ranking the most influential matters of law
11	(METSKER; TROFIMOV; GRECHISHCHEVA, 2020)	Natural Language Processing of Russian Court Decisions for Digital Indicators Mapping for Oversight Process Control Efficiency: Disobeying a Police Officer Case
12	(FANG; ZHAO, 2018)	Nonlinear Dimensionality Reduction with Judicial Document Learning
13	(VIRTUCIO et al., 2018)	Predicting Decisions of the Philippine Supreme Court Using Natural Language Processing and Machine Learning
14	(ALETRAS et al., 2016)	Predicting judicial decisions of the European Court of Human Rights: a Natural Language Processing perspective
15	(VISENTIN; NARDOTTO; O'SULLIVAN, 2019)	Predicting Judicial Decisions: A Statistically Rigorous Approach and a New Ensemble Classifier
16	(SHAIKH; SAHU; ANAND, 2020)	Predicting Outcomes of Legal Cases based on Legal Factors using Classifiers
17	(HE et al., 2018)	PTM: A Topic Model for the Inferring of the Penalty
18	(LI et al., 2019)	Research and Design on Cognitive Computing Framework for Predicting Judicial Decisions
19	(WALTL et al., 2019)	Semantic types of legal norms in German laws: classification and analysis using local linear explanations
20	(WESTERMANN et al., 2019)	Using Factors to Predict and Analyze Landlord-Tenant Decisions to Increase Access to Justice
21	(MEDVEDEVA; VOLS; WIELING, 2019)	Using machine learning to predict decisions of the European Court of Human Rights

Source: Elaborated by the author.

Convolutional Neural Networks (DPCNN) using a data discretization standard pre-processing. The discretization was performed with a pre-processing of the numerical data, such as money and age, which appear a lot in criminal cases. The case facts of the trial were also discretized, later introducing the data processed in the DPCNN model, improving the relationship capacity of the "high dependencies" distance between the features of the judgments. Three datasets for predicting articles of the law based on CAIL2018 with different perspectives. CAIL2018-L consists of all accusations and articles of the law, so it has a great imbalance between categories, containing 1,710,856 cases and 183 classified articles. CAIL2018-H consists of CAIL2018-L as removal of articles from the law with low frequency, totaling 1,477,184 cases and 62 articles. CAIL2018-S includes 196,231 cases randomly selected from CAIL2018-L to test the effect of learning on small datasets, totaling 183 articles. The experiments' results were compared with six text classification models: SVM, FastText, HAN, TextCNN, TexRNN and DPCNN and two more models with model fusion and limit filtering for TextCNN and DPCNN. The DPCNN model with model fusion and threshold filtering achieved the best results. The experiments on the CAIL2018-H data set were significantly better than the others, demonstrating that the data set's imbalance has an important effect on the proposed model. In addition, the proposed model is not well integrated into the manual decision process and cannot reason in legal judgment.

(LEI et al., 2017) proposed a machine learning approach to automatically classify Chinese judgment documents into predefined categories using the Naive Bayes (NB), Decision Tree (DT), Random Forest (RF) and Support Vector Machine (SVM) algorithms. The trial documents were represented as vectors using TF-IDF after the word segmentation. An automatic method was also built to identify judicial stop words, based on terms that frequently occur through court documents (low IDF - Inverse Document Frequency) and less frequent terms in each document. The removal of stop words included numbers, letters and special characters, judicial stop words, place names, and proper names. Three methods of dimension reduction were used: Minimum Document Frequency (DF), Principal Component Analysis (PCA), Truncated Singular Value Decomposition (SVD). The dataset used was based on 6735 Chinese judgment documents, related to product quality liabilities, manually labeled in 13 categories, using 70% to build the classification model and 30% to assess its performance. The experiments were carried out without and with pre-text processing, and experiments with pre-text processing obtained better results with all classifiers, with SVM achieving the best performance, with an average F1-score of 87%, followed in order performance by RFC, DT, and NB. On the other hand, by analyzing the classifiers' performance individually within the categories, RF can have more stable results in small datasets.

(MOK; MOK, 2019) has developed a methodology that uses machine learning to classify new cases of breach of contract based on old case sentences. The project used the ontology of these court decisions to extract and classify sentences according to their type. The project was developed in Python and used its Natural Language Processing and spaCy libraries. Three datasets for breach of contract rulings were used, with 529 sentences, with 70% (370) for train-

ing and 30% (159) for model testing obtained from Supreme Court of Alabama, Court of Civil Appeals of Alabama and United States District Court, on which a list of 1861 keywords was generated. The implementation was based on a Logistic Regression algorithm with Adaptive Linear Neuron modified by the authors. The results obtained were better than the implementation of the scikit-learn logistic regression algorithm, reaching 87% of correct predictions.

(ZHANG et al., 2019a) proposed an evaluation method based on text mutation in the set of tests, to assess the robustness of the prediction model of judicial cases. The evaluation methods were: classification preference test, word order variation test, and noise variation test. The multi-layered fastText, TextCNN, and LSTM models were used. The legal dataset used was CAIL [Xiao et al. 2018], which contains 155,000 training data, 17,000 validation data, and 33,000 test data. The characteristics of each data include charges, fines, related laws, factual descriptions, and sentences. The factual description is a complete text that describes the entire process of the case. The sentence prediction was based on a factual description of the judicial process, divided into six categories: 0-4 years, 4-7 years, 7-15 years, 15-30 years, life imprisonment, death penalty. The model was trained using fastText, TextCNN, and Multi-layer LSTM, obtaining the following Accuracy, respectively: 88.67%, 88.75%, 89.14%. In the classification preference test, all three models rated most nonsensical texts as 0-4 years old, mainly due to data imbalance, so that the three trained models are still not sufficient in terms of classification preferences. The word order variation test showed that the fastText model is less sensitive to word order changes than the other two models. The noise variation test demonstrated that the three models are robust to noise data, and the Accuracy of the prediction is little affected by noise.

(MAHFOUZ; KANDIL; DAVLYATOV, 2018) proposed a model to address construction disputes, more specifically of the Different Site Conditions (DSC) type. The dataset used was based on 600 cases from the Federal Court of New York, on which they were marked with the significant legal factors considered by the judges to base their verdicts, based on the factual extraction process. 15 statistically significant legal concepts were adopted for DSC litigation, implemented in four mechanisms for data representation: Term Frequency (TF), Logarithmic Term Frequency (LTF), Augmented Term Frequency (ATF), and Term Frequency Inverse Document Frequency (TF-IDF). The machine learning classifiers used were: Naïve Bayes (NB), Decision Tree (DT) and Projective Adaptive Resonance Theory (PART). In this way, it resulted in 12 prediction models (a combination of data representation mechanism and machine learning classifiers), which were compared with a prediction model based on Support Vector Machines (SVM), previously developed by the authors. As a result, the machine learning classifiers proved to be an adequate solution for the analyzed task, with the Naïve Bayes classifiers achieving the best result, with 88% accuracy.

(BRANTING et al., 2018) developed models for predicting administrative processes, which are the majority of cases in many countries. The experiments used three datasets: Motion Rulings, which consists of 6,866 pairs of motions/orders extracted from the United States Federal

District Court bulletin; Board of Veterans Appeals (BVA) Decisions, which consists of cases with clear sections (Issues, Introduction, Findings, Conclusions, and Reasons), containing 3,844 instances of 4 classes ((1) the requirements for benefits have been met, (2) the requirements have not been met, (3) the case must be remanded for additional hearings, and (4) the case must be reopened) or 1,605 instances of 2 classes (met or unmet). World Intellectual Property Organization (WIPO) Domain Name Dispute Decisions consists of 5,587 instances with an inclination of approximately 10 to 1 in favor of "transferred". The prediction techniques used were: Hierarchical Attention Networks (HAN), extended from the hierarchical neural network presented by (YANG et al., 2016) for predicting the outcome of legal cases from free text sections of your records; Support Vector Machines (SVM) and Maximum Entropy Classification (Maxent) (BERGER; PIETRA; PIETRA, 1996), often called logistic regression. The prediction results (Frequency-weighted mean F1) obtained were as follows: Motion-Rulings Maxent = 0.742 and SVM = 0.757; SVM BVA = 0.731 and HAN = 0.738; WIPO SVM = 0.950 with balanced data set and HAN = 0.944 with whole data set (inclined). The results indicate that judgments and routine requests are predictable, to a certain extent, from models trained from the text that represents the facts of the case (in the WIPO and BVA datasets) or the motion text (for the whole of data Motion-Rulings).

(SHEN et al., 2018) developed a neural network that has an end-to-end trained external memory structure, similar to (SUKHBAATAR et al., 2015). The proposed network architecture may incorporate statutory law support articles for large external memory, which can be trained without requiring significantly more supervision, which makes it more universally applicable to the criminal charge prediction task's problem. The charge prediction job is to determine the appropriate charge, such as theft and violence, for a criminal case, by analyzing the textual description of the facts. In this work, the charge prediction task was treated as a classification problem. Given a case and relevant articles of law, the type of charge is determined. A dataset from the China Judicial Artificial Intelligence Challenge was used to assess the proposed method's performance. The dataset contains 13,000 cases, with 10,000 training instances and 3,000 test instances. Each case contains part of the description of the fact and the result of the penalty. For the descriptive part of the case, the main factual part was maintained. Time and other contents that are not relevant to the judgment of the crime were removed. There are 16 different types of charges, each with a corresponding legal provision. The proposed solution End-To-End Memory Network (ETE-MN) obtained in the experiments F1-score of 85.1%, surpassing SVM, CNN and LSTM models and the attention-based neural network models proposed by (LUO et al., 2017) and (HU et al., 2018), showing that neural networks using Position Embeddings (pF), Part-of-speech tag Embeddings (POSF), WordNet (WN) and bigram information as input can achieve better results than other methods.

(MARQUES et al., 2019) proposed a method for ranking citations to the most relevant legal principles in legal proceedings to support a given motion. The dataset used was composed of two types of documents: legal jurisprudence, with 2 million documents obtained from the

French Cours d'appel, containing the decision and the related motions, being extracted by regular expression pairs of citation of legal article and motion, resulting in a highly unbalanced dataset; legal principles text, obtained from website the French government's legal principles database called LEGIFRANCE (https://www.legifrance.gouv.fr/), which contains more than 1.5 million French legal articles of 75 codes different legal requirements. The techniques used were: XGBoost classifiers, SHAP framework (LUNDBERG; LEE, 2017)(LUNDBERG; LEE, 2017), word embeddings with FastText skip-gram, Bag-of-words. To validate the model, the following methods were used for comparison: Mutual Information (MI) (VERGARA; ESTéVEZ, 2014), Random forest, XGBoost, Recursive Feature Elimination (RFE), Boruta Feature Selection (KURSA; RUDNICKI, 2010). To evaluate the model, the metrics Normalized Discounted Cumulative Gain (NDCG) (WANG et al., 2013) and a final metric k@RR, obtained by averaging the results of the same motion. The experiments proved that it is possible to automatically detect the most important legal articles for a motion using techniques for interpreting machine learning models. The proposed method is also compatible with highly unbalanced datasets with many features, as is the case with motions argued with few versus many legal articles. It has also been shown that word embeddings trained in legal corpus improve results.

(METSKER; TROFIMOV; GRECHISHCHEVA, 2020) developed a Natural Language Processing method for semi-structured court decisions to improve the quality of knowledge extraction that describes the legal process, identifying combinations of offense facts and procedural facts that can affect court decision-making. The applicability of the results was shown in the development of a Decision Tree (DT) model for the appointment of a prison or fine for disobeying a police officer. One of the main tasks was the development of the method for the automatic layout of court decision texts based on artificial intelligence technologies. The algorithm's task is to extract key entities (indicators) from semi-structured data for further creation of prognostic and descriptive models to analyze the effectiveness of control and surveillance activities. For processing judgments, a combination of TF-IDF and Latent Semantic Analysis (LSA) methods was used, which is a method for analyzing dependencies between features, providing a set of words that behave similarly, and the results were validated and interpreted by specialists. The dataset used was the database of administrative offenses of judicial decisions of the Russian government's automated system, which contains 4,444,562 cases, more specifically cases referring to Article 19.3, which deals with disobedience to police officers, which includes 13.5% of the dataset. The developed method includes a NLP pipeline of court decisions, involving parsing, text segmentation, tokenization, classification and mapping. The result is texts of court decisions marked with the essence of the administrative process, on which machine learning models can be trained. A Decision Tree model was developed to classify cases in prison or fines, reaching the good result of 90% of the ROC curve. As a result, the developed solution can be used as follows: 1. Structuring and automatic markup of legal texts; 2. Analysis of the effectiveness of norms improvement of law-making and law enforcement; 3. Development of knowledge base of intellectual administrative-legal and criminal control systems; 4. Develop-

ment of descriptive and predictive models of law enforcement and lawmaking.

(FANG; ZHAO, 2018) compares several approaches to reducing linear and non-linear dimensionality for classifying court documents and presents some routing procedures for these approaches. The datasets used were collected from (ALETRAS et al., 2016), which include features extracted from textual materials from European Court of Human Rights (ECtHR) legal proceedings with *n-gram* and topic templates. The label indicates whether a case violates a special article of the European Convention on Human Rights. The dataset was divided into three: Article 3 (Prohibits torture, inhuman and degrading treatment) contains 250 records; Article 6 (Protects the right to a fair trial) contains 80 records; Article 8 (Provides the right to respect "someone's private and family life, home, and correspondence") contains 254 records. Each case in the dataset is divided into the following sections: Procedure; Facts, subdivided into Circumstances of the case and Relevant Law; Law (merits of the case with the use of legal arguments). Manifold learning algorithms were used to extract the low dimensional representation of data incorporated in a feature space high dimensional. The n-gram features used were extracted in four different ways: Circumstances, Relevant Law, Facts, Complete (Procedure, Facts, and Law). In order to compare the performance of 14 manifold learning algorithms in n-gram features extracted from lawsuits and five machine learning classifiers, several experiments were carried out. Each of the four extractions of n-gram features was tested using a manifold learning algorithm and a machine learning classifier. The following manifold learning algorithms were used: Autoencoder, Factor Analysis, GPLVM (Gaussian Process Latent Variable Models), Isomap, KernelPCA, Landmark Isomap, Locally-Linear Embedding (LLE), Multidimensional Scaling (MDS), Principal Component Analysis (PCA), Probabilistic PCA, Sammon Mapping, Stochastic Neighbor Embedding (SNE), SymSNE, tSNE. The following machine learning classifiers were used: Bagging, k-NN, Logistic Regression, Random Forest, Support Vector Machine. Thus, 70 (14 x 5) experiments were carried out for one of the 4 extractions of n-gram features and for each of the 3 datasets of the ECtHR articles, totaling 840 experiments. The metric used to analyze the results was the Accuracy of the prediction. Considering the number of experiments, the results obtained varied considerably from the point of view of the best and worst manifold learning algorithms and machine learning classifiers, being described for one of the 12 sets of experiments (4 extractions of n-gram features and 3 articles ECtHR).

(VIRTUCIO et al., 2018) proposed an approach based on Natural Language Processing and Machine Learning to predict the outcome of legal proceedings based on the framework used by (ALETRAS et al., 2016). The dataset used was the Philippine Supreme Court's decisions, being the first systematic study in the prediction of decisions of the Supreme Court of the Philippines based purely on textual content, containing 27,492 cases. The cases were identified as civil and criminal, and criminal cases, which totaled 8,132 cases, were chosen for the job. Of these, 6,483 cases were chosen that refer to appeals in a case decided at first instance. A base of metadata about the cases was created from these cases, including information on whether the decision confirmed or reversed the first instance decision, obtained from the extraction of information

from the original database. The study followed the one proposed by (ALETRAS et al., 2016), which divided the data set into three articles with different aspects of human rights, dividing the data set into four subsets: People, Property, Public Order and Drugs. The subdivision was performed by mapping the articles extracted from the cases with the Revised Penal Code. Natural Language Processing used n-grams and topics. The metric used to compare the results was Accuracy. Initially, the SVM, Logistic Regression and Naïve Bayes machine learning classifiers were used. 64 different tests were carried out for the SVM, Logistic Regression, Naïve Bayes classifiers: 4 data subsets x 8 (n-gram 1 to 4, with and without stemming) x 2 (n-grams and topics). However, the best results were obtained by the SVM, reaching an average of only 45% in the dataset n-grams, worse than 75% of Accuracy achieved by (ALETRAS et al., 2016) and reaching a maximum of 49% in a subset of data. In the topic dataset, the best results were also achieved by the SVM, better than in the dataset n-gram, reaching an average of 55% and a maximum accuracy of 66% in a subset of data. This improvement was also observed by (ALETRAS et al., 2016). From this, another experiment was carried out only on the topic dataset, using the Random Forest classifier, reaching an average accuracy of 59% and a maximum of 68% in a subset of data, slightly better than that observed with the SVM. As a conclusion, it was observed that, apparently, there is no correlation between the use of stemming and the size of n-gram, and the best results were achieved with 2-grams.

(ALETRAS et al., 2016) presents the first systematic study on predicting the outcome of cases judged by the European Court of Human Rights (ECtHR) based only on textual content, whose task is to predict whether a particular article of the Convention has been violated, due to textual evidence extracted from a case, which comprises specific parts referring to facts, the relevant applicable law and the arguments presented by the parties involved. The main hypotheses are that (1) the textual content and (2) the different parts of a case are important factors that influence the outcome reached by the Court. The results corroborate these assumptions. The dataset used were European Court of Human Rights (ECtHR) cases obtained from the HUDOC ECHR database (http://hudoc.echr.coe.int), more specifically cases involving articles 3 (Prohibits torture and inhuman treatment) and degrading), 6 (Protects the right to a fair trial) and 8 (Grants the right to respect "private and family life, home and correspondence"), containing, respectively, 250, 80 and 254 cases. The analyzed sections of the judgments were: Procedures, which contain procedures before ECtHR; Facts, which contains the legal arguments, subdivided into Circumstances, which contains the facts of the case in the domestic court and Relevant Law, which contains references to domestic laws before the case reaches ECtHR; Law, which contains the merits of the case, through the use of legal arguments; Operational Provisions, which contains the result of the case. The texts were extracted from each section using regular expressions, with the exception of the Operational Provisions section, ensuring that the models do not use information pertaining to the outcome of the case. The problem was defined as a binary classification task (violation or non-violation of an article), using N-grams features and Topics to train an SVM machine learning classifier. Topics were created for each article group-

ing N-grams semantically similar, taking advantage of the distributive hypothesis, suggesting that similar words appear in similar contexts. A linear function was applied to identify features important that are indicative of each class, observing the weight learned for each feature (CHANG; LIN, 2008). All violation cases are labeled +1, while non-violation cases are labeled -1. Therefore, features assigned with positive weights are more indicative of violation, while features with negative weights are more indicative of no violation. The metric used to evaluate the results was Accuracy. The result reached an average accuracy of 79%, that is, the models can reliably predict ECtHR decisions with high precision. In general, the best results were obtained with the subsections Circumstances and Facts, indicating that a case's facts are the best sources for predicting decisions.

(VISENTIN; NARDOTTO; O'SULLIVAN, 2019) replicated the experiments carried out (ALETRAS et al., 2016) in the same dataset of the European Court of Human Rights (ECtHR), using a statistically more reliable methodology and analyzed the results using state-of-the-art Bayesian techniques to compare classifiers. First, the results reported in (ALETRAS et al., 2016) were replicated, which were analyzed using a variety of statistical techniques. Second, a new way of making judicial decisions based on documents with different subsections was introduced, motivated by ensemble classifiers' work. Third, the set classifier technique has been extended to other classifiers. The following machine learning classifiers were used in the experiments: SVM, Gradient Boosting Machine (GBM), k-Nearest Neighbor (KNN) and Ensemble Classifier (VISENTIN; NARDOTTO; O'SULLIVAN, 2019) (proposed in this article). The metric used to evaluate the results was Accuracy. The platform used was R. The results of the comparison with (ALETRAS et al., 2016) showed that the experiment carried out, with more statistical rigor, obtained better results than the original. In addition, the new ensemble classifier proposed by the authors had a promising result and surpassed the state of the art.

(SHAIKH; SAHU; ANAND, 2020) developed a model to predict the results of murder-related cases using sentences from the Delhi District Court, applying a binary classification "acquittal" or "conviction" on the accused, being the first research work on a set of Indian legal data for prediction of results of legal cases, according to the authors' knowledge. The work methodology involved the following steps: identification of important factors for the prediction of results; manual extraction of these features through reading and analyzing documents; normalization of features considering minimum and maximum values of the obtained dataset; application of machine learning algorithms. The dataset consists of 86 criminal cases related to the murder of the Delhi District Court, with trials between 2017 and 2018, balanced with 43 cases for the "acquittal" and "conviction" classes. The algorithms used were: Logistic Regression (LR), k-Nearest Neighbors (kNN), Classification and Regression Trees (CART), Naïve Bayes (NB), Support Vector Machines (SVM), Bagging, Random Forest (RF), Boosting. To evaluate the results, the following metrics were used: Accuracy, Precision, Recall and F1-Score. CART achieved the best result in terms of Accuracy, while Bagging and RF came in second. Bagging is the most accurate, with CART and RF second. As for the identification of the majority of

conviction cases, SVM came in first, followed by kNN, CART and NB, with Recall rates equal for conviction cases. LR is consistent for all metrics. Regarding the identification of the majority of conviction cases and being precise at the same time, CART came in first, according to the metric F1 Score. In short, CART had the best results in Accuracy and F1 Score. In addition, all the algorithms had good results in terms of Accuracy and F1 Score obtaining, respectively, results between 85% and 92% and 86% and 92%. In general, therefore, the results are promising, demonstrating the relevance of the factors considered and the success in forecasting results. However, there appears to be a weakness, that is, poor performance in predicting outcomes, for cases with more than one defendant and where the outcome is different for different defendants.

(HE et al., 2018) developed penalty inference techniques for judicial study, which provides efficient utility to assist judges in deciding the final penalty or fine amount. To the authors' knowledge, this is the first work to deeply and completely analyze the problem of inferring penalties through machine learning methods. The authors proposed a generative probabilistic model called Topic Model with Penalties (PTM), to infer the topic of legal cases and the temporal and spatial patterns of topics incorporated in the sentence. Then, the knowledge learned is used to automatically group all cases in a unified way. The authors analyzed a set of data from legal cases and observed that, under the same cause of action, the fine tends to converge, in terms of location and time, that is, those cases that are geographically close and close to time tend to reveal penalty or amount of fine within a certain range. This corresponds to common sense regarding the justice of the legal right. From this analysis, four features different judicial, including timestamp, location, cause of the action and the textual description of the fact of the case, were extracted from the gross cases of the law and inserted in the PTM model. Then, the topic of the test law case is learned, along with the topics learned in the training process, and a voting mechanism can then deduct the law case penalty. The legal case data set was obtained by a crawler, totaling 145,000 Chinese court cases. Each case is originally a textual description, from which, using natural language processing methods, the cause of the action, the temporal information and the spatial information of each case, which are the inputs of the PTM model, were extracted. The next step was to filter the cases with vague or undefined temporal and spatial information. It was defined a threshold of 30 feature words to remove cases with insufficient topical words. After this work, the final dataset used was 67,060 court cases. The developed PTM model was compared with the following methods: SVM (CORTES; VAPNIK, 1995), fastText (JOULIN et al., 2017), TextCNN (KIM, 2014). To evaluate the results, the following metrics were used: Accuracy, Precision, Recall and F1-Score. The results of the experiments showed that the proposed PTM model surpassed the other methods in all metrics.

(LI et al., 2019) developed a framework of cognitive computing for semantic understanding, learning of knowledge and legal reasoning in the Chinese legal field. The framework proposed has three layers: 1 - Legal Semantic Understanding Layer, where the legal factors are first represented in a formal way; 2 - Legal Knowledge Learning Layer, where legal factors are extracted and concepts and their relationships are enhanced with a combination of deep and rule-based

learning methods; 3 - Legal Knowledge Reasoning Layer, where a prediction model is generated and trained to make judicial decisions. When a description of fact is brought into the model, the likelihood of court decisions is provided automatically. The data set was obtained from China Judgments Online (http://wenshu.court.gov.cn), totaling 695,418 court documents of divorce cases. Of these, 50,000 court documents were randomly selected, 80% of which were used for training, and 20% for model validation. In layer 1, first, the court documents and the description of the facts are pre-processed. The semantics of legal factors by extraction rules were written in the TML programming language (JIAJING; XIAOMING; TAO, 2015). In layer 2, the legal factors' values are extracted, and the concepts and relationships are increased through deep learning. In this layer, both rule-based methods and deep learning were used, using Google's word2vec (MIKOLOV et al., 2013), Bi-LSTM and CRF (MA; HOVY, 2016; ZHAI et al., 2017), Naïve Bayes and Convolution Network (CNN) (ZENG et al., 2014). At layer 3, a Markov Logic Network (MLN) model (SINGLA; DOMINGOS, 2005) is generated, and then court documents are used to train the weights of the rules. The results of the predictions include the textual descriptions along with the probabilities of each outcome factor. For comparison of results, an SVM model was implemented, with features TF-IDF as input and chi-square was used to select 2,000 features, with a linear core function being applied to classify the features. The metrics used to evaluate the results were Precision, Recall, and F1-Score. The results showed that the proposed model surpassed the SVM in all metrics by around 8%, demonstrating that the model can effectively predict decisions for divorce cases with different styles of expression and offers better performance than traditional methods, in addition to predictive machine learning results can be easily understood by the general public, according to the applied induction rules.

(WALTL et al., 2019) developed two experiments for automatic classification of legal norms in relation to their semantic type. Based on legal and linguistic aspects, a taxonomy of nine different semantic types was created, focusing on the functional aspects of the rules, such as duty, permission, prohibition, etc. Experiment 1 used a rules-based approach using manual standard definitions. Experiment 2 used an approach based on machine learning comparing different classifiers and parameter settings. The data set used comprises 601 sentences that constitute the lease law of the German Civil Code (§535 - §597) in its consolidated version with effect from 21 February 2017. In a first pre-processing stage, the raw text from these articles was segmented into sentences. In the next step, the 601 phrases were classified manually by a single domain specialist, according to their taxonomy. For experiment 1, classification of legal norms based on rules, the UIMA Ruta language (KLüGL et al., 2014) was used. Four iterations were performed, with the rules being adjusted based on the previous iteration results, which were evaluated by the metrics Precision, Recall and F1-Score, individually for the nine semantic types. The best results were obtained in the last iteration, with the exception of category IX "Reference". Although the general quality of the rules is acceptable, categories IV "Prohibition" and VIII "Definition" performed poorly. For these categories and in general, a

better overall score of the rules could be obtained if the phrase segmentation module created not only whole sentences, but also auxiliary phrases. As German legal texts make heavy use of long, nested sentences, this information would be beneficial for the implementation of the rules. For experiment 2, classification of legal norms based on supervised machine learning, the following classifiers were used: Multinomial Naive Bayes (MNB), Logistic Regression (LR), Support Vector Machines (SVM) with linear kernel function, Random Forest (RF) and Multilayer Perceptrons (MLP). For each classifier, three combinations of features were performed: without removing stopwords, removing stopwords, and using TF-IDF to calculate tokens. The platform used for the development of the experiments was Python with the scikit-learn library. Local linear explanations were extracted for SVM using the Local Interpretable Model-agnostic Explanations (LIME) library (RIBEIRO; SINGH; GUESTRIN, 2016). The metrics used were the same as in experiment 1: Precision, Recall and F1-Score. Results varied widely between classifiers and classes. The general average showed a minimum F1-Score of 0.57 (MNB + SW + TF-IDF) and a maximum of 0.83 (SVM). Compared with the results of experiment 1, most classifiers had a worse result regarding F1-Score, except SVM and LR. Also notable that all classifiers had better results when no treaties were made for the input data (without removing stopwords and without TF-IDF calculation). In short, the best F1-Score results obtained were 0.78 in experiment 1 (based on rules) and 0.83 in experiment 2 (based on machine learning). In general, the authors identified that the classifiers' decisions are highly related to the knowledge engineering approach, as tokens similar are highly relevant during the classification task.

(WESTERMANN et al., 2019) reported results from the JusticeBot Project, in which two sets of data extracted from 1 million written decisions of the Régie du logement du Québec (RDL) were analyzed. Using an empirical methodology, 44 factors were identified that occur in disputes in which the tenant seeks a solution due to problems with the rented apartment, such as the existence of bed bugs, high noise levels or insulation problems. In the first dataset, the factors were used to record 149 cases and a correlation was found between how many factors are found in a case and how likely it is that the judge will grant rent reduction to a tenant; the amount of reduction was also greater in cases with more factors. For the second dataset (39 cases with bedbugs, extracted from the first dataset), detailed factors were developed and used to note the cases and a series of plausible correlations were found, with the average damage premium being higher in the cases with high-intensity infestations. The methodology used first organized and represented the legal rules and problems involved in the cases. To represent legal rules as a set of propositions, Default Logic Framework (WALKER, 2007) was used, where one of which is the conclusion and the remaining propositions are the conditions of the rule (that is, the conditions under which the conclusion is true) , so that the rules were represented as a rule tree (WALKER et al., 2017), formulated from the Code Civil du Québec. The use of the conditions of the rules tree as a method to classify the cases decided by the RDL allowed to organize the cases by the legal issues decided and to classify the parts of the text in these decisions by legal question, making it possible to associate past cases with future cases, being the unit of analysis (or topic)

the condition of the legal rule. The next step was to identify the relevant factors for deciding a legal condition in the rule tree, in which a method was developed to extract and analyze which factors contribute to a judge's decision, in order to build a model of what "good habitable condition" means in terms of real-world facts. To develop a case classification system, the Grounded Theory Method (WEBLEY, 2010) was used. In summary, the methodology involved the following phases: 1. Identification of factors, establishing a taxonomy on the recurrent factors in the cases, with 44 factors being found; 2. Refinement of the factors in which it was decided to focus on the factor of the existence of bedbugs; 3. Annotation of the factors in the cases; 4. Construction of machine learning models, one for classification and one for regression. The classification model was created to estimate the prediction of the value of the factors, based on the hypothesis that treating the factors as features binary input (presence or absence of a particular feature) it is possible to create a model capable of predicting whether a judge would grant a rent reduction or not. The algorithm used was Random Forest, with the metric Precision to evaluate the results. The regression model was developed to predict the number of months of rent reduction. The algorithm used was Random Forest Regressor. The metric used to evaluate the results was Mean Squared Error, which takes into account the average difference between the prediction and the real values and raises it to the square. The results of the experiments were similar to those of the baseline or slightly above and show the challenges of using computers to help carry out legal analyzes and one of the main reasons identified is that when extracting categorical information from decision texts, a large amount of information is lost.

(MEDVEDEVA; VOLS; WIELING, 2019) developed three experiments for predicting court decisions based on machine learning. The data set used is the one made publicly available by the European Court of Human Rights (ECtHR), with the documents being obtained from the HUDOC ECHR database (http://hudoc.echr.coe.int). In the experiments, five parts of a judicial decision were used: Procedures, Circumstances, Relevant Law, Facts (Circumstances and Relevant Law), Complete (Procedures and Facts). In experiment 1, the possibilities of using words and phrases in the text of cases were investigated to predict court decisions, classifying cases of all ECtHR articles in violation and non-violation. In experiment 2, the first experiment approaches were used to estimate the potential to predict future cases. In experiment 3, the ability to predict court sentences using only the surnames of the judges involved was tested, and therefore the Introduction part of an ECtHR court decision was also used. The techniques used in the experiments were TF-IDF and SVM. To identify the set of features to be included, (for example, only unigrams, only bigrams, only trigrams, a combination of these, larger n-grams?), a program was created to test all combinations, so that each article used a different different combination. For example, for Article 2 the parts Procedures and Facts, n-grams 3-4, were used, with capital letters removed and without removed stopwords. For Article 11, the Procedures part was used, without removing capital letters and removing stopwords. The metrics used to evaluate the results were Accuracy, Precision, Recall and F1-Score. The result of experiment 1 showed an average accuracy of 75% in predicting the violation of 9 ECtHR

Table 9: Publications per year

Year	Relevant	All
2015	0	160
2016	1	213
2017	2	287
2018	6	463
2019	10	687
2020	2	476
Total	**21**	**2286**

Source: Elaborated by the author.

articles. However, the result of experiment 2 showed that the prediction of decisions for future cases based on past cases negatively affects performance (average precision range 58 to 68%). In turn, experiment 3 demonstrated that predicting results based only on the judges' surnames can achieve a relatively high ranking performance (average Accuracy of 65%).

3.1.4 Results and discussion

This section presents the results and discussion of the systematic mapping study, considering the research questions that originated the study. The analysis of the results focuses on the presentation of the publication frequencies for each category. This makes it possible to verify which categories have been emphasized in previous research and, thus, to identify gaps and possibilities for future research (PETERSEN et al., 2008).

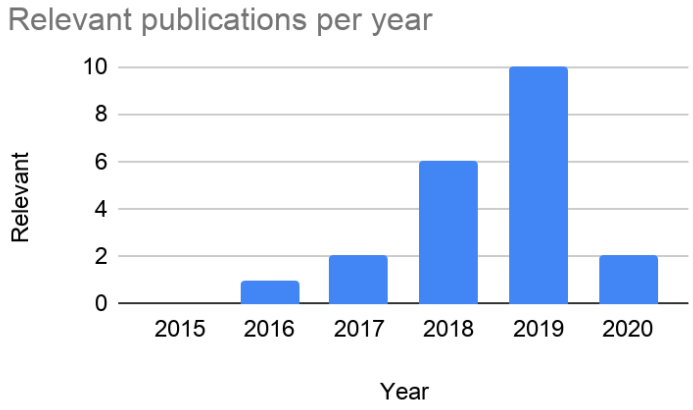
RQ1 - What are the main current researches on Decision Support applications to support judges?

The main objective of a systematic mapping is to provide an overview of a research area and identify the quantity, type of research, and the results available. In addition, it can be used to map publication frequencies over time to identify trends. A secondary objective may be to identify the forums in which research in the area has been published (PETERSEN et al., 2008). These objectives are reflected in the first research question (QP1).

Table 9 shows the number of publications per year, both selected relevant publications and all publications initially obtained from research in scientific databases. As Figure 34 illustrates, it is possible to notice that there is an increase in publications in the area over the years, disregarding the current year of 2020, since the survey was conducted in May 2020 and there is a natural delay in publications, due to the normal process of making them available. In addition, no relevant publication was selected for the year 2015 and only 1 for 2016, so that filtering to consider publications only from 2015 onwards was agreed.

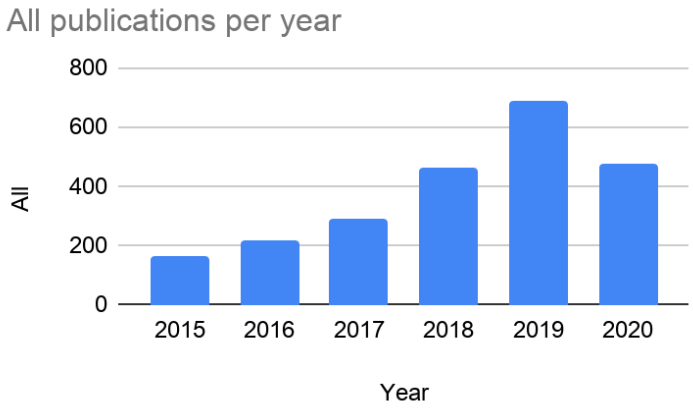
Likewise, as shown in Figure 35, considering all publications, a gradual increase in the

Figure 34: Relevant publications per year



Source: Elaborated by the author.

Figure 35: All publications per year



Source: Elaborated by the author.

number of publications in the area over the years is also noticed, disregarding the current year, in a way to validate the selection of relevant articles.

Table 10 shows the number of relevant publications selected by digital library. All four originally selected libraries have selected articles. However, as Google Scholar was used to complement the research, ordered by relevance to its large number of results already filtered by the year of publication, some other digital libraries have also had results. As the Google Scholar search engine did not consider the entire search string originally defined, being less restrictive, it brought additional results from the original four libraries, as can be seen by the amount of selected publications from IEEE Xplore, greater than the amount originally selected of the digital library itself.

As shown in Figure 36, the main scientific databases in which we identified publications to Artificial Intelligence-based decision support systems to support judges are Springer Link and IEEE Xplore.

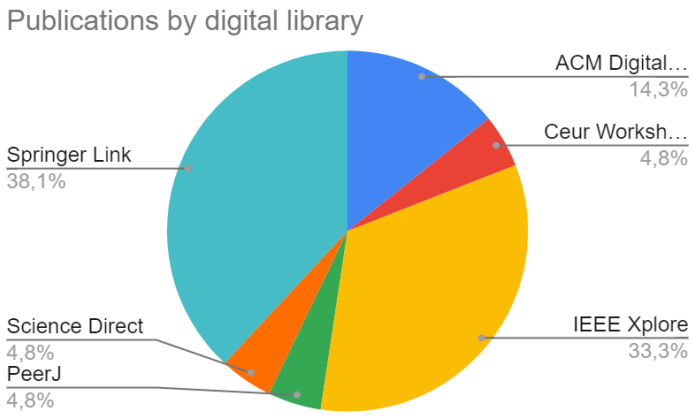
An important analysis of the related works refers to the type of decision support developed,

Table 10: Publications by digital library

Digital library	Quantity
ACM Digital Library	3
Ceur Workshop Proceedings	1
IEEE Xplore	7
PeerJ	1
Science Direct	1
Springer Link	8
Total	**21**

Source: Elaborated by the author.

Figure 36: Publications by digital library



Source: Elaborated by the author.

Table 11: Types of decision support

Type of decision support	IDs	Quantity
Prediction of the outcome of court cases	[2], [7], [8], [11], [12], [13], [14], [15], [16], [18], [20], [21]	12
Penalty prediction	[1], [6], [17]	3
Charge prediction	[1], [9]	2
Law article prediction	[1], [3]	2
Classification of judgment documents	[4]	1
Classification of legal norms	[19]	1
Classification of sentences	[5]	1
Ranking of most relevant legal principle citations	[10]	1

Source: Elaborated by the author.

since the term decision support is broad and includes several implementation possibilities. In this regard, as shown in Table 11, the selected publications describe experiments with eight types of decision support. Although some publications had more than one objective, from the point of view of the type of decision support developed, with the exception of the publication [1], which experimented with three types of decision support, all other publications reported a single type.

The vast majority of publications reported work on predicting the outcome of court cases, which is therefore the main form of decision support in the context of AI applied to law. The type of prediction, in turn, can vary and depends on each case. For example, the outcome prediction developed by (ALETRAS et al., 2016) aims to predict whether a particular article of the European Convention on Human Rights has been violated or not.

The second most used type of decision support was the penalty prediction for defendants in criminal cases. In this case, sentences are defined in categories, for example, from 1 to 4 years in prison. It is also possible to highlight, in the sequence and with two publications each, the types of decision support charge prediction and law article prediction. Both have potential for use in real applications, since they direct the judge to a specific crime or a specific article of the Law to which the process is related.

It was also analyzed whether published works were reporting real systems or just experiments. In this regard, it is clear that there is still no massive use of AI in the construction of real decision support systems related to Law, with the majority of publications reporting experiments based on possible usage scenarios. Only one of the selected works, (MOK; MOK, 2019), reported that the work presented concerns a part of a development project. According to the authors, "classifying sentences is an important first step in our long-term research objective: to develop a machine learning system that analyzes hundreds or thousands of previous violations of court decisions similar to the case in question, thus providing a powerful tool for legal professionals when faced with new cases and issues".

Analyzing the publications referenced in the selected works to identify possible baselines

Table 12: Publications cited

Publication cited	IDs	Quantity
(ALETRAS et al., 2016)	[2], [3], [8], [11], [12], [13], [15], [16], [18], [20], [21]	11
(LIU; CHEN, 2017)	[15]	1
(MEDVEDEVA; VOLS; WIELING, 2019)	[15]	1
(VIRTUCIO et al., 2018)	[15]	1

Source: Elaborated by the author.

and benchmarking, it was found that (ALETRAS et al., 2016) stands out, being referenced by 11 of the other 20 selected articles, and in some of the works, it was even used as a comparison of the validation of the experiments. As shown in Table 12, in addition to (ALETRAS et al., 2016), only three other articles were referenced by the works selected in this mapping.

Although they were not selected in this mapping, it was noticed that two other publications were highlighted among the selected works. (KATZ; II; BLACKMAN, 2017) was referenced by 8 papers ([2], [3], [9], [13], [15], [16], [18], [21]) and (SIM; ROUTLEDGE; SMITH, 2014) for 4 works ([2], [12], [14], [18]).

RQ2 - What datasets are used?

Table 13 presents the datasets used by the publications selected in this study. Although the use of datasets can vary widely, the European Court of Human Rights (ECtHR) and CAIL2018 (Chinese AI and Law challenge dataset) (XIAO et al., 2018) datasets can be highlighted.

ECtHR is an international court that decides on requests alleging violations by any State Party of the civil and political rights established in the European Convention on Human Rights. The Court's judgments contain a distinct structure, which makes them particularly suitable for text-based analysis. A judgment contains (among others) a description of the procedure followed at the national level, the facts of the case, a summary of the parties' observations, which comprise their main legal arguments, the reasons in question articulated by the Court and the operational provisions. The judgments are clearly divided into different sections, covering this content, which allows for a direct standardization of the text and, consequently, makes text-based analyzes possible.

In turn, O CAIL2018 is the first large-scale Chinese legal dataset for judgment prediction, containing more than 2.6 million criminal cases published by the Supreme People's Court of China, several times larger than other datasets about predicting the outcome of court cases, in addition to having more detailed and richer annotations of the trial results. The dataset consists of law articles, charges and applicable prison sentences, which are expected to be deducted according to the case descriptions.

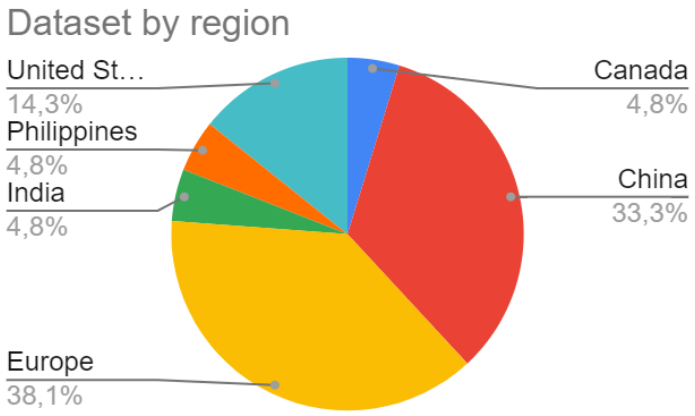
Analyzing the datasets according to the regions of origin, adding to the ECtHR the datasets of specific European countries, we note the predominance of works based on data from Europe

Table 13: Datasets used

ID	Publication	Dataset	Region of data
1	(CHEN et al., 2019)	CAIL2018 (Supreme People's Court of China)	China
2	(LIU; CHEN, 2017)	European Court of Human Rights (ECtHR)	Europe
3	(ZHANG et al., 2019b)	CAIL2018 (Supreme People's Court of China)	China
4	(LEI et al., 2017)	Unspecified Chinese	China
5	(MOK; MOK, 2019)	Supreme Court of Alabama Court of Civil Appeals of Alabama United States District Court	United States
6	(ZHANG et al., 2019a)	CAIL2018 (Supreme People's Court of China)	China
7	(MAHFOUZ; KANDIL; DAVLYATOV, 2018)	Federal Court of New York	United States
8	(BRANTING et al., 2018)	Motion Rulings (United States federal district court) Board of Veterans Appeals (BVA) World Intellectual Property Organization (WIPO)	United States
9	(SHEN et al., 2018)	China Judicial Artificial Intelligence Challenge	China
10	(MARQUES et al., 2019)	French Cours d'appel LEGIFRANCE	France
11	(METSKER; TROFIMOV; GRECHISHCHEVA, 2020)	Administrative offenses of the Russian government's automated system	Russia
12	(FANG; ZHAO, 2018)	European Court of Human Rights (ECtHR)	Europe
13	(VIRTUCIO et al., 2018)	Philippine Supreme Court	Philippines
14	(ALETRAS et al., 2016)	European Court of Human Rights (ECtHR)	Europe
15	(VISENTIN; NARDOTTO; O'SULLIVAN, 2019)	European Court of Human Rights (ECtHR)	Europe
16	(SHAIKH; SAHU; ANAND, 2020)	Delhi District Court	India
17	(HE et al., 2018)	Unspecified Chinese Judgments	China
18	(LI et al., 2019)	China Judgments Online	China
19	(WALTL et al., 2019)	German Civil Code	Germany
20	(WESTERMANN et al., 2019)	Régie du logement du Québec (RDL)	Canada
21	(MEDVEDEVA; VOLS; WIELING, 2019)	European Court of Human Rights (ECtHR)	Europe

Source: Elaborated by the author.

Figure 37: Datasets by region



Source: Elaborated by the author.

and China, with more than 71% of publications. The United States comes next with 14.3% of the publications, as shown in Figure 37.

In this respect, the highlight of China stems from the fact that many authors are from China and, of course, the focus of his study tends to be their country of origin. The use of datasets from Europe derives both from the aspect of the naturalness of their authors and from the availability of public data in an appropriate format for experiments in decision support systems related to law. In addition, the work of (ALETRAS et al., 2016), which used data from ECtHR (Europe), motivated other researchers to do the same experiment in order to overcome their work.

In this way, it may be of interest to specific countries and Courts to open their data and make available annotated datasets so that they can be used more easily by the scientific community, which will eventually result in a greater amount of work based on their data.

RQ3 - What are the main techniques used?

Table 14 presents the main techniques used and the IDs of the articles that use them, either for the development of the experiments or for the purposes of comparison with the experiments carried out. In a preliminary way, it is noticed the use of several techniques and resources for the development and comparison of experiments, including traditional machine learning techniques, deep learning, natural language processing, word embedding and segmentation, feature selection, explainable AI and ontology. Some techniques can be classified in more than one category, and in this analysis it was considered the category where it was applied for the development of the experiment. For example, FastText can be used both for text classification, in which it uses deep learning techniques, as a tool for word embedding, used in the middle of the pipeline of the proposed architectures. In this mapping, although it has been used in some studies to compare the results, FastText was used primarily in the development of the experiments as a word embedding tool. Same case as Autoencoder, which, although it is a

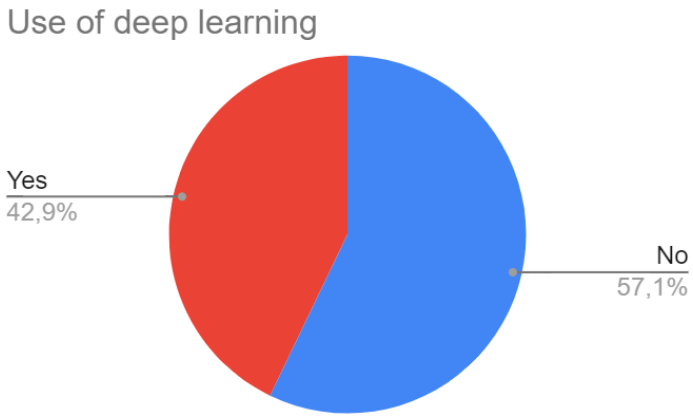
deep learning approach, was used in the experiments as a feature selection mechanism, and is therefore analyzed in this way in this mapping.

Table 14: Techniques used by the articles

Techniques	**IDs**	**Quantity**
Support Vector Machine (SVM)	[2], [3], [4], [8], [9], [12], [13], [14], [15], [16], [17], [18], [19], [21]	14
Random Forest (RF)	[2], [4], [10], [12], [13], [16], [19], [20]	8
Logistic Regression (LR)	[2], [5], [12], [13], [16], [19]	6
Naïve Bayes (NB)	[4], [7], [13], [16], [18], [19]	6
Term Frequency Inverse Document Frequency	(TF-IDF) [1], [7], [11], [19], [21]	5
FastText	[1], [3], [6], [17]	4
k-Nearest Neighbors (k-NN)	[2], [12], [15], [16] 4	
TextCNN[1], [3], [6], [17]	4	
Bagging Bootstrap aggregating)	[2], [12], [16]	3
Decision Tree (DT)	[4], [7], [11]	3
Convolutional Neural Networks (CNN)	[9], [18]	2
Hierarchical Attention Networks (HAN)	[3], [8]	2
Jieba	[1], [2]	2
Attention-based Neural Network proposed by [Hu et al. 2018]	[9]	1
Attention-based Neural Network proposed by [Luo et al. 2017]	[9]	1
Augmented Term Frequency (ATF)	[7]	1
Bi-LSTM	[18]	1
Boruta Feature Selection	[10]	1
Classification and Regression Trees (CART)	[16]	1
Conditional Random Field (CRF)	[18]	1
Deep Pyramid Convolutional Neural Networks (DPCNN)	[3]	1
Default Logic Framework	[20]	1
Dimension reduction	[4]	1
Discretization of numerical data	[3]	1

End-To-End Memory Network (ETE_MN) with WordNet and Bigram	[9]	1
Ensemble Classifier [Visentin et al. 2019]	[15]	1
Gradient Boosting Machine (GBM)	[15]	1
Grounded Theory Method	[20]	1
Latent Semantic Analysis (LSA)	[11]	1
Local Interpretable Model-agnostic Explanations (LIME)	[19]	1
Logarithmic Term Frequency (LTF)	[7]	1
Long Short-Term Memory (LSTM)	[9]	1
Manifold learning algorithm: Autoencoder	[12]	1
Manifold learning algorithm: Factor Analysis	[12]	1
Manifold learning algorithm: Gaussian Process Latent Variable Models (GPLVM)	[12]	1
Manifold learning algorithm: Isomap	[12]	1
Manifold learning algorithm: KernelPCA	[12]	1
Manifold learning algorithm: Landmark Isomap	[12]	1
Manifold learning algorithm: Locally-Linear Embedding (LLE)	[12]	1
Manifold learning algorithm: Multidimensional Scaling (MDS)	[12]	1
Manifold learning algorithm: Principal Component Analysis (PCA)	[12]	1
Manifold learning algorithm: Probabilistic PCA	[12]	1
Manifold learning algorithm: Sammon Mapping	[12]	1
Manifold learning algorithm: Stochastic Neighbor Embedding (SNE)	[12]	1
Manifold learning algorithm: SymSNE	[12]	1
Manifold learning algorithm: tSNE	[12]	1
Markov Logic Network (MLN)	[18]	1
Maxent (Maximum Entropy Classification)	[8]	1
Multi-Label k-Nearest Neighbor (ML-KNN)	[1]	1
Multi-label learning using local correlation (ML-LOC)	[1]	1
Multi-layer LSTM	[6]	1
Multilayer Perceptrons (MLP)	[19]	1
Mutual Information (MI)	[10]	1
Ontology	[5]	1
Pre-processing of text	[4]	1
Projective Adaptive Resonance Theory (PART)	[7]	1

Figure 38: Use of deep learning



Source: Elaborated by the author.

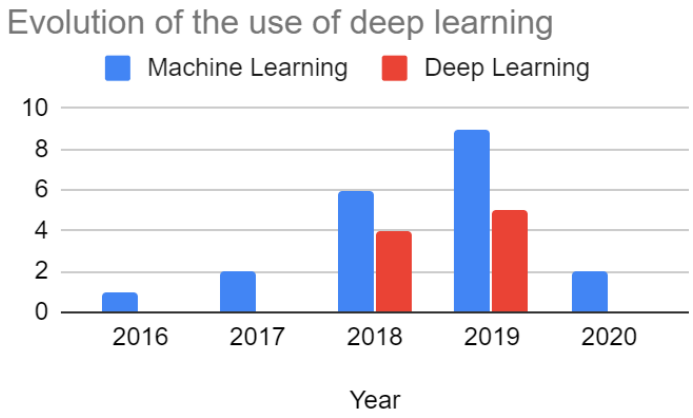
Random Forest Regressor	[20]	1
Recursive Feature Elimination (RFE)	[10]	1
Term Frequency (TF)	[7]	1
Text mining language TML	[18]	1
Text region embedding	[3]	1
TextRNN	[3]	1
Threshold filtering	[3]	1
Topic Model PTM	[17]	1
UIMA Ruta Rule-based Text Annotation	[19]	1
word2vec	[18]	1
XGBoost	[10]	1

Source: Elaborated by the author.

Although it is still less used than traditional machine learning, as shown in Figure 38. Use of deep learning, it was possible to confirm that the use of deep learning is increasing over the years, as shown in Figure 39. Evolution of the use of deep learning. Even with a limitation in the number of years evaluated, considering that the use and deep learning is more recent in this research field, it is possible to believe the numbers show, in fact, a trend of use of deep learning algorithms for the development of decision support applications in the context of law. The year 2020 cannot be considered in this evolution analysis, because the research took place in the month of May of the year and there is a normal delay due to the time for the publication of the articles, in addition to the fact that 2020 is an atypical year due to the pandemic of Covid-19.

Analyzing specifically the deep learning techniques used in the articles selected and presented in Table 15, it is noticed that there is a tendency to use Convolutional Neural Networks,

Figure 39: Evolution of the use of deep learning



Source: Elaborated by the author.

Table 15: Deep learning techniques

Deep learning techniques	IDs	Quantity
Convolutional Neural Networks	[1], [3], [6], [9], [17], [18]	6
Attention-Based Neural Networks	[3], [8], [9]	3
Long Short-Term Memory Networks	[6], [9], [18]	3
End-To-End Memory Networks	[9]	1
Multilayer Perceptrons	[19]	1
Projective Adaptive Resonance Theory	[7]	1
TextRNN	[3]	1

Source: Elaborated by the author.

Attention -Based Neural Networks and Long Short-Term Memory Networks. The results obtained by these deep learning techniques, together with those obtained by End-To-End Memory Networks and TextRNN are very promising, so it is possible to consider them as the state of the art in the development of deep learning architectures for the decision support in law.

RQ4 - What are the main techniques used?

Table 16 presents the platforms used for the experiments in the selected publications. This information was not available in some publications and more than one platform was used in others, so that the total amount does not equal the amount of articles analyzed. As expected, most of the works used Python as a development platform. In addition, it is possible to assume that a large part, perhaps all, of the works that did not mention the development platform, used Python and its libraries, since this programming language has become the standard for developing data science solutions and artificial intelligence.

Although Python is slower at run time and has some design restrictions compared to compiled languages like C or C ++, it is preferred by scientists and developers in the field of data analysis, numerical calculations and almost all technical domains, such as machine learning,

Table 16: Development platforms

Development platform	IDs	Quantity
Python	[1], [4], [5], [10], [11], [13], [14], [17], [19], [21]	10
C++	[7]	1
MATLAB	[2]	1
R	[15]	1
Scala	[8]	1
WEKA	[8]	1

Source: Elaborated by the author.

Table 17: Use of Law articles

Considers the Law	IDs	Quantity
Yes	[9], [10], [20]	3
No	[1], [2], [3], [4], [5], [6], [7], [8], [11], [12], [13], [14], [15], [16], [17], [18], [19], [21]	18

Source: Elaborated by the author.

AI, deep learning, etc. The following reasons for this preference can be highlighted (NAGPAL; GABRANI, 2019): less coding and high readability, portability and flexibility, platform independence, low and high level programming balance, continuity (ideal candidate for a first programming language), data structures and open source libraries available that greatly facilitate the development of deep learning solutions.

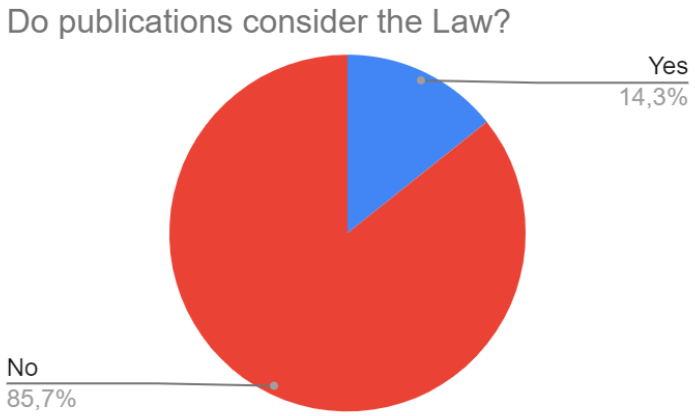
RQ5 - Are the articles of the Law considered for the prediction results?

One of the most specific points investigated in this systematic mapping, refers to the verification regarding the use of the Law articles to support the prediction of results. Table 17 shows the answer to this question for the selected articles. In this research question, a publication considers or does not consider the Law articles, so that Figure 40 visually illustrates the percentage of publications for each case.

This research question refers to the discussion between legal formalism and legal realism (LEITER, 2010; POSNER, 1986; SCORDATO, 2012). In legal formalism, it is believed that decision making by judges can be modeled deductively using formal rules or with more complex reasoning paradigms than deduction when formal rules are insufficient to guarantee a specific result. In legal realism, it is believed that judges make their decisions based primarily on the stimulus of the facts of the case, since the legal rules are rationally indeterminate on many occasions.

From the mapping carried out, this point can be an important gap in decision support systems, since only 14.3% of publications consider the Law to predict results. Regardless of the legal formalism or realism, the importance of the facts is related to the text of the Law, and changes in the text, for example, whether or not something is a crime or the time of penalty of

Figure 40: Percentage of publications that consider the Law



Source: Elaborated by the author.

a certain crime, should influence new decisions, even about similar past facts. In this respect, it makes sense to include the Law in Artificial Intelligence models, which, apparently, according to the analyzed studies, is being underused and perhaps it shouldn't be.

RQ6 - What are the metrics to perform the evaluation?

As shown in Table 18, the main metric used to evaluate the results of the experiments is accuracy, which answers the question "how often is the classifier correct?". In the sequence, the traditional multiclass metrics "F1" were also widely used: Precision (number of correct positive results divided by the total number of positive results returned by the classifier), Recall (number of correct positive results divided by the total number of samples that should have identified as positive - relevant samples) and F1-Score (harmonic mean between Precision and Recall, in order to bring a unique number to measure the general quality of the model). Some studies, in turn, used specific metrics such as Kappa, NDCG and K@rr to evaluate their results.

Thus, unless you want to measure something specific, the use of the metrics Accuracy, Precision, Recall and F1-Score are indicated, both for a more common understanding of the results obtained, as to allow for easier comparison of the work done with other works, enabling a better assessment of the model's quality.

RQ7 - What are the results obtained?

The summary of the results presented in the selected publications in this systematic mapping is presented in As shown in Table 19. Due to the differences between the studies, in general it is quite difficult and even unlikely to be able to make an adequate comparison between the results of the works, except in cases which use the same dataset. In addition, some works present the results in very detailed levels, while others present the results briefly.

Table 18: Metrics used to evaluate the results

Metrics	IDs	Quantity
Accuracy	[1], [2], [3], [4], [5], [6], [7], [8], [12], [13], [14], [15], [16], [17], [21]	15
F1-Score	[1], [3], [4], [8], [9], [16], [17], [18], [19], [21]	10
Precision	[1], [4], [8], [16], [17], [18], [19], [20], [21]	9
Recall	[1], [3], [4], [8], [16], [17], [18], [19], [21]	9
F1-Macro	[3]	1
F1-Micro	[3]	1
Frequency-weighted Mean F1	[8]	1
k@RR	[10]	1
Kappa	[7]	1
Mean Squared Error	[20]	1
Normalized Discounted Cumulative Gain (NDCG) (WANG et al., 2013)	[10]	1
ROC Curve	[11]	1

Source: Elaborated by the author.

Table 19: Summary of results obtained

ID	Publication	Results
1	(CHEN et al., 2019)	Penalty prediction model: Average accuracy of FastText 88.75% and TextCNN 88.76%. Prediction model of legal provisions based on TextCNN: F1-score 86.27%, better than the ML-KNN (ZHANG; ZHOU, 2007) 74.38% and ML-LOC (HUANG; ZHOU, 2012) 66.49% comparison methods. Charge prediction model based on FastText and TextCNN: For each model, Precision, Recall, F1 respectively. FastText 60.16%, 54.45%, 57.16%; TextCNN 64.65%, 56.56%, 60.33%; ML-KNN 51.34%, 47.88%, 49.55%.
2	(LIU; CHEN, 2017)	For Article 3, Bagging and SVM achieved an accuracy of more than 73%. For Article 6, SVM achieved an accuracy of more than 85%. For Article 8, SVM again achieved the best accuracy of more than 77%.

3	(ZHANG et al., 2019b)	The DPCNN model with model fusion and threshold filtering achieved the best results. The experiments on the CAIL2018-H dataset were significantly better than the others, demonstrating that the imbalance of the data set has an important effect on the proposed model. In addition, the proposed model is not well integrated into the manual decision process and does not have the ability to reason in legal judgment.
4	(LEI et al., 2017)	SVM achieved the best performance, with an average F1-score of 87%, followed in order of performance by RFC, DT and NB. On the other hand, by analyzing the performance of the classifiers individually within the categories, RF can have more stable results in small datasets.
5	(MOK; MOK, 2019)	87% correct predictions.
6	(ZHANG et al., 2019a)	Accuracy of fastText, TextCNN and Multi-layer LSTM: 88.67%, 88.75%, 89.14%.
7	(MAHFOUZ; KANDIL; DAVLYATOV, 2018)	The Naïve Bayes classifiers achieved the best result, with an accuracy of 88%.
8	(BRANTING et al., 2018)	The prediction results (Frequency-weighted mean F1) obtained were as follows: Motion-Rulings Maxent = 0.742 and SVM = 0.757; BVA SVM = 0.731 and HAN = 0.738; WIPO SVM = 0.950 with balanced data set and HAN = 0.944 with whole dataset (inclined).
9	(SHEN et al., 2018)	The proposed End-To-End Memory Network (ETE-MN) solution obtained in the F1-score experiments of 85.1%, surpassing SVM, CNN and LSTM models and the attention-based neural network models proposed by (LUO et al., 2017) and (HU et al., 2018), showing that neural networks using Position Embeddings (pF), Part-of-speech tag Embeddings (POSF), WordNet (WN) and bigram information as input can achieve better results than other methods.
10	(MARQUES et al., 2019)	The experiments proved that it is possible to automatically detect the most important legal articles for a motion using machine learning model interpretation techniques. The proposed method is also compatible with highly unbalanced datasets with a large number of features, as is the case with motions argued with few versus many legal articles. Legal corpus-trained word embeddings have also been shown to improve results.

11	(METSKER; TROFIMOV; GRECHISHCHEVA, 2020)	90% of the ROC curve.
12	(FANG; ZHAO, 2018)	Considering the number of experiments, the results obtained varied considerably from the point of view of the best and worst manifold learning algorithms and machine learning classifiers, being described for one of the 12 sets of experiments (4 extractions of n-gram features and 3 articles from ECtHR).
13	(VIRTUCIO et al., 2018)	The best results were obtained by the SVM, however, reaching an average of only 45% in the n-gram dataset, worse than 75% of accuracy achieved by (ALETRAS et al., 2016) and reaching a maximum of 49% in a subset of data. In the topic dataset, the best results were also achieved by the SVM, better than in the n-gram data set, reaching an average of 55% and a maximum accuracy of 66% in a subset of data, with this improvement was also observed by (ALETRAS et al., 2016). From this, another experiment was carried out only on the topic dataset, using the Random Forest classifier, reaching an average accuracy of 59% and a maximum of 68% in a subset of data, slightly better than that observed with the SVM.
14	(ALETRAS et al., 2016)	The result reached an average accuracy of 79%, that is, the models can reliably predict ECtHR decisions with high precision. In general, the best results were obtained with the subsections Circumstances and Facts, indicating that the facts of a case are the best sources for forecasting decisions.
15	(VISENTIN; NARDOTTO; O'SULLIVAN, 2019)	The results of the comparison with (ALETRAS et al., 2016) showed that the experiment carried out, with more statistical rigor, obtained better results than the original. In addition, the new ensemble classifier proposed by the authors had a promising result and surpassed the state of the art.

16	(SHAIKH; SAHU; ANAND, 2020)	CART achieved the best result in terms of accuracy, while Bagging and RF came in second. Bagging is the most accurate, with CART and RF in second. As for the identification of the majority of conviction cases, SVM came in first, followed by kNN, CART and NB, with equal recall rates for conviction cases. LR is consistent for all metrics. Regarding the identification of the majority of conviction cases and being precise at the same time, CART came in first, according to the F1 score metric. In short, CART had the best results in accuracy and F1 score. In addition, all the algorithms had good results in terms of accuracy and F1 score, obtaining results between 85% and 92% and 86% and 92%, respectively. In general, therefore, the results are promising, demonstrating the relevance of the factors considered and the success in forecasting results. However, there appears to be a weakness, that is, poor performance in predicting outcomes for cases with more than one defendant and where the outcome is different for different defendants.
17	(HE et al., 2018)	The results of the experiments showed that the proposed PTM model surpassed the other methods in all metrics.
18	(LI et al., 2019)	The results showed that the proposed model surpassed the SVM in all metrics by around 8%, demonstrating that the model can effectively predict decisions for divorce cases with different styles of expression and offers better performance than traditional methods, besides the predictive results of machine learning can be easily understood by the general public, according to the applied induction rules.
19	(WALTL et al., 2019)	The best F1-Score results obtained were 0.78 in experiment 1 (based on rules) and 0.83 in experiment 2 (based on machine learning). In general, the authors identified that the classifiers' decisions are highly related to the knowledge engineering approach, as similar tokens are highly relevant during the classification task.
20	(WESTERMANN et al., 2019)	The results of the experiments were similar to those of the baseline or slightly above and show the challenges of using computers to help carry out legal analyzes. One of the main reasons identified is that when extracting categorical information from the decision texts, a large amount of information is lost.

21	(MEDVEDEVA; VOLS; WIELING, 2019)	The result of experiment 1 showed an average accuracy of 75% in the prediction of the violation of 9 ECtHR articles. The result of experiment 2 showed, however, that the prediction of decisions for future cases based on past cases negatively affects performance (average precision range 58 to 68%). In turn, experiment 3 demonstrated that predicting results based only on the surnames of the judges who judge can achieve a relatively high ranking performance (average accuracy of 65%).

Source: Elaborated by the author.

The results achieved by some works were highlighted by their authors as good or very good. However, in some cases the authors themselves found that the results achieved were below expectations. Nevertheless, in general, the results show that the use of Natural Language Processing and Machine Learning allows its application for the construction of decision support systems in the context of Law.

Another point to be highlighted is the relationship of the dataset with the results obtained. If the dataset is very unbalanced or if you have a small amount of data, for example, it is much more difficult to achieve good results. Considering that, in the real world, the dataset is what it is and it is not possible to simply choose it, it is important that, for the development of good decision support systems, specific work is done on the data, improving it as far as possible to increase the possibility of achieving good results.

Discussion

From this systematic mapping, it is possible to conclude that, in recent years, there has been a growing scientific interest in AI based decision support systems to support judges of Law, which is in line with the evolution that is occurring in key support technologies for the development of these systems, such as the evolution of Natural Language Processing and Deep Learning.

Regarding the type of decision support, the vast majority of publications selected in this mapping reported experiments related to the prediction of the outcome of court cases, demonstrating a well-defined research focus. It is also possible to highlight the following types of decision support: penalty prediction for defendants in criminal cases, charge prediction and law article prediction, the latter two having a good potential for using in real applications, since they direct the judge for a specific crime or a specific article of the Law to which the case is related. However, in general, the articles report only experiments based on possible usage scenarios, with only one of the papers presenting a work that is part of a real development project,

although it is still an initial step of the project.

With this mapping it was also possible to identify that the work published in (ALETRAS et al., 2016) can be considered as a baseline and benchmarking for research in the area, mainly for experiments that use the dataset provided by the European Court of Human Rights (ECtHR), which was the main dataset used by the selected publications.

Another widely used dataset was CAIL2018 (Chinese AI and Law challenge dataset), which demonstrates that making prepared datasets publicly available can encourage experiments on that dataset, which can be a good strategy for regions or Courts that wish to advance the use of artificial intelligence in the context of law. It is clear that China is working hard in this direction, since a large number of authors are originally from that country and, naturally, the focus of their study tends to be their country of origin.

This systematic mapping showed that there is an increase in the use of deep learning techniques for the development of decision support systems in the context of AI in Law, highlighting Convolutional Neural Networks, Attention-Based Neural Networks, Long Short-Term Memory Networks, End-To-End Memory Networks and TextRNN. The motivation for such approaches can be identified partially in the aspects of the Neural nets architectures mentioned and in the support it can bring to the text processing task. Differently from other data, such as images, for example, in natural language textual documents the elements structure and the elements semantic relation is of paramount importance. Therefore, some of the recent architectures and mechanisms dedicated to model recurrence, relationship among elements and memory aspects can improve the results in this field.

A necessary direction for researchers in the field is to use contextual word embedding mechanisms in their architectures or pre-trained neural language models built from deep learning such as ELMo (PETERS et al., 2018), ULMFit (HOWARD; RUDER, 2018) and transformer based architectures (VASWANI et al., 2017) such as BERT (DEVLIN et al., 2019) and others. These recents advances have brought the NLP community close to having an "ImageNet for language" (RUDER, 2018). This analogy comes from the capability of models to learn higher-level nuances of language, similarly to how ImageNet has enabled training of CV models that learn general-purpose features of images.

A very important concern in the use of Artificial Intelligence in several domains, including the legal field, is related to ethics and the interpretability of the results produced by such systems. These aspects still appear as gaps in the studies identified in this systematic mapping. Recently, however, this concern has been addressed in Brazilian regulation, notably in Resolution No. 615/2025 of the National Council of Justice (CNJ, 2025), which establishes guidelines for the development, governance, auditability, monitoring, and responsible use of AI-based solutions in the Judiciary. The resolution sets forth foundational principles such as transparency, accountability, respect for fundamental rights, and the requirement of human oversight. It also defines that AI-generated outputs used in support of judicial decisions must preserve equality, avoid discriminatory or abusive bias, and promote plurality, contributing to fair judgments

and reducing marginalization and errors arising from prejudice. Additionally, it requires that AI systems provide, whenever technically feasible, sufficient explanations that enable human auditability and interpretation of the results.

One of the most specific issues investigated in this systematic mapping was the verification of the use of the Law in the construction of decision support systems for judges. Regardless of legal formalism or legal realism, the importance of the facts is related to the text of the Law, and changes in the text, apparently, should influence new decisions, even if on similar past facts. However, it was identified that almost all of the works did not use the text of the Law in the development of applications, so this point could be a gap in the decision support systems and an opportunity for the new works to be developed.

The main metrics used to evaluate the results of the experiments were the Accuracy and the traditional multiclass metrics "F1": Precision, Recall and F1-Score. Some studies, due to their peculiarities or objectives, used specific metrics such as Kappa, NDCG and K@rr to evaluate their results. In this way, unless you want to measure something specific, the use of the metrics Accuracy, Precision, Recall and F1-Score are indicated, both for a more common understanding of the results obtained, as well as to allow more easily a comparison of the current work with others works, allowing a better evaluation of the model quality.

The most used development platform was Python, which, currently, has become the standard platform for the development of data science and artificial intelligence solutions. Although it is slower at runtime and has some design restrictions compared to compiled languages like C or C++, its preference stems mainly from a lower need for coding, platform independence, and the availability of data structures and deep learning open source libraries that facilitate the development of such solutions.

The results obtained in the experiments carried out and presented in the publications selected in this systematic mapping study show that the use of Natural Language Processing and Machine Learning must be considered in the construction of decision support systems in the context of Law, since, for the most part, the results of the experiments were considered good or very good by their authors. The experiments that did not have such favorable results had their reasons analyzed by their authors, who mainly identified problems as a very unbalanced dataset or with a very small amount of data. Therefore, there is a direct relationship between the quality of the data and the results obtained. Thus, we can conclude that for the development of good decision support systems for judges, it is necessary to do specific work on data, improving them, as far as possible, to increase the possibility of achieving good results.

More specifically, this systematic mapping also identified that the use of semantics should be considered in the construction of models. The experiments demonstrated that the choice of features based on semantic information obtained by natural language processing of the facts text significantly affects the performance of the models. In addition, models built on high-level semantic features are more interpretable for humans, which is important for applicability in the real world. This aspect also demonstrates the importance of using new word embedding

approaches and language models that have greater capacity to capture the semantic similarity of texts.

3.1.5 Evaluation

Artificial Intelligence is being increasingly used in the various areas of today's society, including in the legal field. The application of AI in Law has existed for over 60 years, both with applications focusing on lawyers and the judiciary. With the increase in the number of lawsuits and the difficulty in expanding human resources in the same proportion, Artificial Intelligence is believed to be an important factor in automating tasks, attending to the growing number of lawsuits with the same resources.

Recent advances in Artificial Intelligence, mainly in Natural Language Processing and Machine Learning, have provided tools capable of analyzing legal texts in order to create predictive models for judicial results, considered as decision support systems. More recently, the development of prediction models based on deep learning and word embedding technologies has increased the possibilities of these models achieving good accuracy and being used in the real world.

In order to identify possible research gaps and opportunities in the area of Artificial Intelligence and Law, more specifically in decision support systems for judges, a literature review was carried out according to the methodology of Systematic Mapping Studies, also known as scope studies and designed to provide a broad overview of a research area.

The systematic mapping was performed on five scientific databases, considering only the research published since 2015. The results obtained in the mapping showed that this decision was adequate, since the number of publications in the area was not relevant in 2015, increasing its relevance every year. There was also a prevalence of research originating in China, focused on Chinese datasets. The studies used various techniques for the development and comparison of experiments, both for traditional machine learning and deep learning. Among the deep learning techniques, we can highlight Convolutional Neural Networks, Attention-Based Neural Networks, Long Short-Term Memory Networks, End-To-End Memory Networks and TextRNN.

The vast majority of works did not consider the Law for the creation of models, which can be a gap in current research and focus on new research. Most of the workers used similar evaluation metrics, the main ones being Accuracy, Precision, Recall and F1-Score. However, the works varied widely in their format, so it is not possible to make an adequate comparison between them.

The results of the experiments presented in the publications were promising and indicate that the use of Natural Language Processing and Machine Learning should be considered in the construction of decision support systems in the context of Law. Unsatisfactory results were related to problems such as a very unbalanced dataset or a very small amount of data, showing that there is a direct relationship between the quality of the data and the results obtained.

The experiments also demonstrated that the choice of features based on semantic information improves the performance of the models, in addition to being more interpretable for humans, which is important for applicability in the real world. In addition, in general, it was possible to notice that some studies deal with preliminary elements that make up the decision making, while others are more focused on decision making itself.

Considering recent advances in natural language processing based on the use of contextual word embedding and pre-trained language models, which provide greater semantic power, new work in this area certainly must consider their use to take advantage of the benefits of using the state-of-the-art in NLP tasks, which are intrinsically related to the construction of decision support systems in the context of Law.

An important concern in AI based decision support systems refers to ethics and the ability to interpret the results obtained and these aspects seem to be a gap in the works analyzed in this systematic mapping. Further work in this field should consider these requirements for the work to be effective and applicable in real systems.

3.2 Portuguese language resources related to AI applied to Law

This section presents a non-systematic review that was performed with the aim of complementing the work described in section 3.1, now focusing on the study of dedicated works in the Portuguese language applied to Law, in order to identify possible research gaps and opportunities in the area of Artificial Intelligence and Law. This survey on related works regarding the use of AI in the Judiciary in Brazil includes the production of datasets, language models, and other resources, presented in the following subsections.

3.2.1 brWaC corpus

Following the WaCky (Web-As-Corpus Kool Yinitiative) methodology (BARONI et al., 2009; BERNARDINI; BARONI; EVERT, 2006), (WAGNER FILHO et al., 2018) constructed the brWaC Corpus, a large web corpus for Brazilian Portuguese, aiming to achieve a size comparable to the state of the art in other languages, forming the basis and enabling further studies targeting Brazilian Portuguese. This resource is freely available for both querying, through a NoSketch Engine interface, and downloading in CoNLL and Moses formats.

The collection process resulted in a corpus with 2.68 billion tokens, a 72% increase in relation to the previous version, and 5.79 million types, distributed in 145 million sentences and 3.53 million documents. These counts strictly exclude any tokens with numbers or special characters (except hyphens). The obtained multi-domain corpus was annotated with tagging and parsing information. The incidence of non-unique long sentences, an indication of replicated content, which reaches 9% in other web corpora, was reduced to only 0.5%. Domain diversity was also maximized, with 120,000 different websites contributing content.

3.2.2 LeNER_BR dataset

LeNER-Br (ARAUJO et al., 2018) is the first Portuguese language dataset for named entity recognition applied to legal documents. It consists entirely of manually annotated legislation and legal cases texts and contains tags for persons, locations, time entities, organizations, legislation and legal cases.

To compose the dataset, 66 legal documents from several Brazilian Courts were collected. Courts of superior and state levels were considered, such as Federal Court of Justice, Superior Justice Tribunal, Minas Gerais Court of Justice and Brazilian Federal Court of Accounts. In addition, four legislation documents were collected, such as Lei Maria da Penha, giving a total of 70 documents. The dataset was manually annotated each one of the documents with the following tags: "ORGANIZACAO" for organizations, "PESSOA" for persons, "TEMPO" for time entities, "LOCAL" for locations, "LEGISLACAO" for laws and "JURISPRUDENCIA" for decisions regarding legal cases. The last two refer to entities that correspond to "Act of Law" and "Decision" classes from the Legal Knowledge Interchange Format ontology (HOEKSTRA et al., 2007) respectively.

To create the dataset, 50 documents were randomly sampled for the training set and 10 documents for each of the development and test sets. Named entities are assumed to be non-overlapping and not spanning more than one sentence. The total number of tokens in LeNER-Br is comparable to other named entity recognition corpora such as Paramopama Brazilian-Portuguese (MENDONçA JúNIOR et al., 2015) and CONLL-2003 English (TJONG KIM SANG; DE MEULDER, 2003) datasets, with 318,073, 310,000 and 301,418 tokens respectively.

A state-of-the-art machine learning model based on a LSTM-CRF architecture, was trained on this dataset and was able to achieve a good performance. According to the authors, there is room for improvement, which means that this dataset will be relevant to benchmark methods that are still to be proposed, Also according to the authors, future work would include the expansion of the dataset, adding legal documents from different courts and other kinds of legislation, e.g. Brazilian Constitution, State Constitutions, Civil and Criminal Codes, among others. In addition, the use of word embeddings trained on a large corpus of legislation and legal documents could potentially improve the performance of the model.

3.2.3 BERTimbau model

(SOUZA; NOGUEIRA; LOTUFO, 2020) trained the first pre-trained BERT (DEVLIN et al., 2019) model for Brazilian Portuguese, which they nickname BERTimbau. It achieves SOTA performances on three downstream NLP tasks: sentence textual similarity, recognizing textual entailment, and named entity recognition. The model was trained from scratch on the brWaC Corpus (WAGNER FILHO et al., 2018) for 1.000.000 steps, using a whole-word mask. The models improve the state-of-the-art in all three tasks, outperforming Multilingual BERT and

confirming the effectiveness of large pretrained language models for Portuguese.

The authors trained a BERT-CRF model, i.e., a BERT model with a token-level classifier on top followed by a linear-chain CRF to the NER task on the Portuguese language, combining the transfer capabilities of BERT with the structured predictions of CRF, exploring feature-based and fine-tuning training strategies for the BERT model. This was the first work to employ BERT models to the NER task in Portuguese.

3.2.4 PTT5 model

A pretrained language model based on T5 (RAFFEL et al., 2019) model architecture was trained by (CARMO et al., 2020) in Brazilian Portuguese on the brWaC Corpus (WAGNER FILHO et al., 2018), with the training starting from the corresponding original T5 checkpoints released. The author created the Portuguese vocabulary using a similar procedure than the original T5 paper, training a SentencePiece (KUDO, 2018) model on a corpus of 2 million sentences randomly chosen from the Portuguese Wikipedia. The models performance were evaluated against other Portuguese pretrained models and multilingual models on three different tasks, obtaining significantly better performance over the original T5 models.

The resulting models achieved better performance than the original T5 models on the Portuguese sentence entailment and NER tasks, and the Portuguese vocabulary proved to be better than using the original T5 vocabulary. For the two ASSIN 2 (REAL; FONSECA; GONçALO OLIVEIRA, 2020) tasks, semantic similarity and entailment prediction, pretraining all weights leads to better performance than pretraining the vocabulary embeddings only, but Portuguese T5 model obtained results a few points below to the BERTimbau Large.

3.2.5 Portuguese named entity recognition using BiLSTM-CRF

A deep learning architecture based on bidirectional Long Short-Term Memory with Conditional Random Fields (BiLSTM-CRF), which had shown state-of-the-art performance for English, Spanish, Dutch and German languages, was evaluated by (CASTRO; SILVA; SILVA SOARES, 2018) for named entity recognition task. The authors perform the tuning of hyperparameters for Portuguese corpora. The results achieve state-of-the-art performance for Portuguese language using the optimal values for them.

3.2.6 Assessing the Impact of contextual embeddings for Portuguese NER

In this work, (SANTOS et al., 2019) assess how different combinations of shallow word embeddings and contextual embeddings impact NER for the Portuguese Language. They show a comparative study of 16 different combinations of shallow and contextual embeddings and explore how textual diversity and the size of training corpora used in language models impact

the NER results. They also evaluate NER performance using the HAREM corpus (FREITAS et al., 2010). Their NER system based on a BiLSTM-CRF+FlairBBP neural network, using the Flair library (AKBIK; BLYTHE; VOLLGRAF, 2018), achieves the state-of-the-art in Portuguese NER evaluated by CoNLL-2002 Script, which is an entity-level evaluation metric, i.e., a predicted entity is considered correct only if its exactly span and type is correct.

3.2.7 Impact of intradomain finetuning in Portuguese NER

A study on the impact of intradomain finetuning of deep language Models for legal NER in Portuguese was carried out by (BONIFACIO et al., 2020). They investigated the impact of this step on named entity recognition (NER) for Portuguese legal documents, exploring different scenarios considering two deep language architectures: ELMo (PETERS et al., 2018) and BERT (DEVLIN et al., 2019), four unlabeled corpora and three legal NER tasks for the Portuguese language. Experimental findings show a significant improvement on performance due to language model finetuning on intradomain text. This work also evaluate the finetuned models on two general-domain NER tasks, in order to understand whether the aforementioned improvements were really due to domain similarity or simply due to more training data. The achieved results also indicate that finetuning on a legal domain corpus hurts performance on the general domain NER tasks. Additionally, the BERT mode finetuned on a legal corpus achieves the state-of-the-art performance on the LeNER-Br corpus (ARAUJO et al., 2018), a Portuguese language NER corpus for the legal domain, using the seqeval Python library, a well-tested implementation of CoNLL-2002 Script.

3.2.8 Deep learning for named entity recognition in legal domain

This work is a master thesis from (CASTRO, 2019), in which the Legal domain was studied, targeting specifically the Brazilian Labor Law. The author used a model based on deep neural networks, evaluating different forms of word representations. The evaluated models were applied to Portuguese language, for both Legal and general domains. To this end, language models based on the ELMo (PETERS et al., 2018) architecture were trained for both domains, as well as static word embeddings, specific for the Legal domain. It was verified the best type of pre-trained word embeddings for each domain, after performing a comparative study between the types of word embeddings applied to the NER task. For the training of the Legal domain NER models, ELMo and static word embeddings, two different corpora were produced and annotated, based on a collection of public documents from the Brazilian Labor Court. For the Portuguese general domain NER model, a new state-of-the-art result was achieved for the HAREM benchmark, with 83.22% F-Score for the selective scenario, and 78.04% for the total scenario. For the Brazilian Labor Law domain, a model with 93.81% F-Score was obtained.

3.2.9 Contextual representations and semi-supervised NER for Portuguese language

This paper describes the participation of (CASTRO, 2019) in IberLEF (Iberian Languages Evaluation Forum), Task 1: Named Entity Recognition (COLLOVINI et al., 2019).

For the Portuguese NER task, the authors experimented with different systems based on deep learning architectures, for both NER model and word representations. For the NER model, they used the BiLSTM-CRF architecture, which were a reference for sequential classification NLP tasks. For word representations they experimented with character level features from Convolutional Neural Networks, Wang2Vec (LING et al., 2015) pre-trained word embeddings, and the ELMo (PETERS et al., 2018) embeddings from a bidirectional language model. The authors evaluated different models with different types of word representations in 5 different corpora, and submitted a system based on two different ELMo, combined with character level features. The model was trained in a semi-supervised scenario, in order to account for the lack of certain types of categories in the used corpora.

The following public corpora were used for the model proposed in this work: WikiNER (NOTHMAN et al., 2013), LeNER-Br (ARAUJO et al., 2018), HAREM I (SANTOS; CARDOSO, 2006) and MiniHAREM golden collections, and Paramopama (MENDONçA JúNIOR et al., 2015). The authors also used a private legal corpus provided by the Datalawyer company, consisting of 76 annotated documents from the Brazilian Labor Court.

The main contribution is the use of ELMo embeddings for the Portuguese NER task, with the pre-trained ELMo model being publicly available at https://allennlp.org/elmo.

3.2.10 Impact of text specificity and size on word embeddings performance in Brazilian legal domain

The impact of text specificity and size on word embeddings performance in Brazilian Legal Domain is evaluated by (DAL PONT et al., 2020). The authors trained word embeddings models in the legal domain with different levels of specificity and size and evaluated their impact on text classification. They chose GloVe (PENNINGTON; SOCHER; MANNING, 2014) representation due to its good results in many NLP tasks, including text classification, and also for its training time which is significantly less than other techniques like Word2vec (MIKOLOV et al., 2013) and FastText (JOULIN et al., 2017).

To evaluate each of the trained embeddings, they used a set of judgments from the JEC located at the Federal University of Santa Catarina (JEC/UFSC), which is related only to failures on air transport services (Consumer Law), and Convolutional Neural Networks as a classification model.

The three corpora used in embeddings training is composed by 3.7 billion, 100 million and 1.19 billion words for general, air transport and global corpora, respectively. Others were created based on them according to the following smaller corpora sizes, considering the word

count: 1,000; 10,000; 50,000; 100,000; 200,000; 500,000; 1,000,000; 5,000,000; 10,000,000; 25,000,000; 100,000,000; 500,000,000; 750,000,000 and 1,000,000,000.

In the context of legal documents in Portuguese, the authors concluded that there is more assertiveness when the trained text resembles the text that we have to classify. This behavior does not occur with the size of the training corpus, because when you reach a certain amount of words, the results suggest stability. Despite the moderate gain in accuracy with the specific texts as test set (air transport curve), the authors considered this result relevant because it shows that the use of billions of tokens, as in previous works, does not bring great contributions. Therefore, the specificity of the text set impacts more positively on the classification task than the size of the text set.

3.2.11 Evaluation of Portuguese language models and word embeddings on semantic similarity tasks

In this paper, (RODRIGUES et al., 2020) proposed an evaluation of the performance of deep neural language models on the semantic similarity tasks provided by the ASSIN (FONSECA et al., 2016) dataset against classical word embeddings, both for Brazilian Portuguese and for European Portuguese. The experiments indicated that the ELMo (PETERS et al., 2018) language model was able to achieve better accuracy than any other pretrained model which has been made publicly available for the Portuguese language, and that performing vocabulary reduction on the dataset before training not only improved the standalone performance of ELMo, but also improved its performance while combined with classical word embeddings. They also demonstrated that FastText (JOULIN et al., 2017) skip-gram embeddings can have a significantly better performance on semantic similarity tasks than it was indicated by previous studies in this field.

3.2.12 Evaluation of Portuguese word embeddings on word analogies and natural language tasks

In this paper, (HARTMANN et al., 2017) evaluated different word embedding models trained on a large Portuguese corpus, including both Brazilian and European variants.

A large corpus from several sources were collected in order to obtain a multi-genre corpus, representative of the Portuguese language. They tokenized and normalized the corpus in order to reduce the vocabulary size, under the premise that vocabulary reduction provides more representative vectors. Word types with less than five occurrences were replaced by a special UNKNOWN symbol. Numerals were normalized to zeros; URL's were mapped to a token URL and emails were mapped to a token EMAIL. Then, they tokenized the text relying on whitespaces and punctuation signs, paying special attention to hyphenation. For instance, clitic pronouns like "machucou-se" were kept intact.

The authors trained 31 word embedding models using FastText (JOULIN et al., 2017), GloVe (PENNINGTON; SOCHER; MANNING, 2014), Wang2Vec (LING et al., 2015) and Word2vec (MIKOLOV et al., 2013). They evaluated them intrinsically on syntactic and semantic analogies and extrinsically on POS tagging and sentence semantic similarity tasks.

The obtained results were aligned with those from (FARUQUI et al., 2016), which suggest that word analogies are not appropriate for evaluating word embeddings; task-specific evaluations appear to be a better option. As future work, the authors intended to try different tokenization and normalization patterns, and also to lemmatize certain word categories like verbs, since this could significantly reduce vocabulary, allowing for more efficient processing. An evaluation with more NLP tasks would also be beneficial to the understanding of different model performances.

3.2.13 Named Entity Recognition with neural character embeddings

In this work (SANTOS; GUIMARãES, 2015) proposed a language-independent NER system that uses automatically learned features only. Their approach is based on the CharWNN deep neural network, which uses word-level and character-level representations (embeddings) to perform sequential classification. Most state-of-the-art named entity recognition (NER) systems until that time relied on handcrafted features and on the output of other NLP tasks such as part-of-speech (POS) tagging and text chunking.

The authors performed an extensive number of experiments using two annotated corpora in two different languages: HAREM I (SANTOS; CARDOSO, 2006) corpus, which contains texts in Portuguese; and the SPA CoNLL2002 (TJONG KIM SANG, 2002) corpus, which contains texts in Spanish. The experimental results gave evidence of the contribution of neural character embeddings for NER. Moreover, they demonstrated that the same neural network which has been successfully applied to POS tagging could also achieve SOTA results for language-independent NER, using the same hyperparameters, and without any handcrafted features.

3.3 General analysis of related work

This chapter described a systematic mapping study on AI based decision support systems for judges and a survey of related works regarding the use of AI in the Judiciary in Brazil, carried out in order to identify gaps and opportunities in the area of Artificial Intelligence and Law, in the context of Brazil and the Portuguese language.

The systematic mapping has shown that, although in recent years the use of natural language processing and deep learning in the development of AI-based decision support systems has increased, the use of attention-based neural networks is still timid. This can be explained due to the recent launch of this type of architecture and the period of systematic mapping studies. On the other hand, the systematic mapping identified that the use of semantics should be considered

in the construction of m odels, so that a necessary direction for researchers in the field is to use contextual word embedding mechanisms in their architectures, such as pre-trained neural language models and transformer-based architectures.

The survey of related works regarding the use of AI in the Judiciary in Brazil has shown that basic resources like datasets and general language models in Portuguese trained on some transformer-based architectures already exist. However, there is a need to increase the quantity and quality of Portuguese resources and there is still no methodology for the development of AI applications applied to Law following the recent natural language processing advances, based on the use of deep learning with transformer-based architectures, pre-trained language models and transfer learning.

4 PROPOSED METHODOLOGY

"Without a method, everything is vague and confused."
Descartes (inspired by Discourse on the Method, 1637)

This chapter presents the proposed methodology and its components for the development of Artificial Intelligence applications in the legal domain. Although the methodology is potentially applicable to other areas, it was designed with particular attention to the specificities of the legal field—such as the structure of datasets, the need for domain-specific model fine-tuning, and the complex normative context. Thus, the focus of this work lies in the development of AI systems tailored to legal applications.

Recent advancements in Natural Language Processing (NLP) have shifted the paradigm from rule-based and statistical methods to deep learning architectures, notably those based on attention mechanisms (OTTER; MEDINA; KALITA, 2021; GALASSI; LIPPI; TORRONI, 2021). The Transformer architecture (VASWANI et al., 2017), based entirely on attention, allows for substantial parallelization and has become the foundation for state-of-the-art results across various NLP tasks.

These advances are strongly connected to the emergence of large pre-trained language models (PLMs) (BENGIO et al., 2003), trained through unsupervised or self-supervised tasks to predict linguistic sequences. Transfer learning techniques (PAN; YANG, 2010) enable these models to be fine-tuned for specific tasks and domains, making them highly adaptable and effective in downstream legal applications.

The proposed methodology is significant because it systematizes the development process of AI applications in Law, incorporating state-of-the-art techniques while addressing the unique challenges of this field. Due to the novelty and rapid evolution of these technologies, available knowledge is often fragmented and mostly confined to academic or research environments. This creates a gap between cutting-edge research and its practical adoption in legal institutions such as courts or law firms. The methodology proposed herein bridges this gap, offering a validated development pipeline that has been tested in real-world scenarios.

Nonetheless, the field remains dynamic: continuous research and frequent breakthroughs may render specific techniques obsolete or introduce new best practices. Therefore, while the methodology provides a structured and robust foundation, it is intentionally designed to be revisited and updated as the field evolves. A clear example of such disruptive evolution occurred shortly after this research was qualified, with the public release of ChatGPT and other large-scale instruction-tuned language models (ZHAO et al., 2023), capable of performing a wide range of NLP tasks with conversational fluency and general reasoning abilities.

Beyond the methodology itself, this work contributes to the legal AI field by delivering reusable artifacts such as datasets and fine-tuned models, which support rapid prototyping and experimentation. These resources are detailed in Section 4.2.

The sections below provide an overview of the methodology, followed by a detailed description of its technical components, associated contributions, and a critical discussion of its practical implications.

4.1 Overview

This methodology guides the development of AI applications in Law by leveraging recent advancements in deep learning, particularly the Transformer architecture and transfer learning strategies. While the general principles are applicable to any domain, this work focuses on the specific requirements of the legal field and the Portuguese language.

Unlike traditional software development projects, where requirements and business logic are well defined and the success of the implementation is largely predictable, AI-based systems—especially those based on machine learning—entail inherent uncertainties. These include unpredictability regarding model performance, dependency on data quality and availability, and difficulties in evaluating generalization in real-world scenarios.

We identify two primary reasons for these challenges:

a) **AI applications are inherently data-dependent.** Traditional systems can be exhaustively tested against predefined rules. However, AI models generalize from examples and may encounter input distributions in production that were not observed during training. This raises the risk of incorrect or biased outputs, including hallucinations.

b) **AI projects resemble R&D initiatives.** The technologies involved are often novel to the development team and the organization, requiring experimentation with tools and approaches whose effectiveness cannot be guaranteed in advance.

To mitigate these challenges, the methodology introduces four strategic actions:

a) Close and sustained involvement of the business team to align expectations and domain requirements;

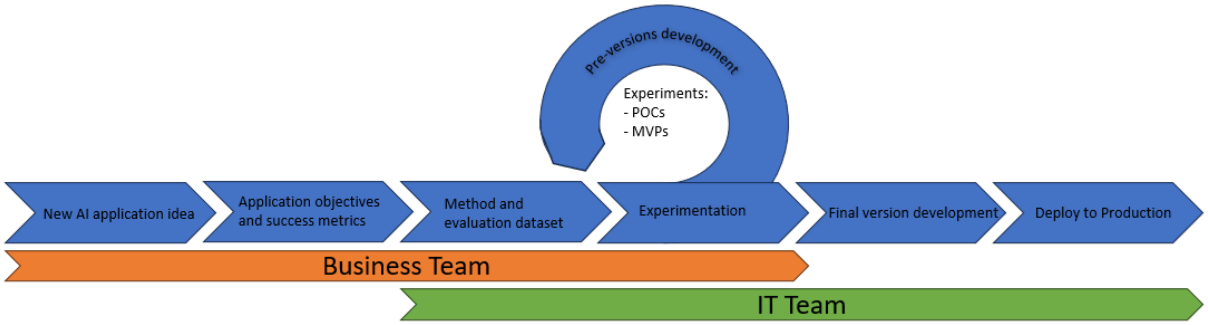
b) Construction of application-specific datasets that reflect real-world use cases;

c) Definition of rigorous experiment evaluation methods and metrics;

d) Iterative development of prototypes, Proof of Concept (POCs) and MVPs (Minimum Viable Products), to validate the feasibility and business value of the solution.

Based on this foundation and the practical experience accumulated in experiments conducted at the Court of Justice of Rio Grande do Sul (TJRS), the methodology places a strong emphasis on business-domain collaboration, not only for defining success criteria but also for co-developing datasets and participating in evaluation cycles. Figure 41 illustrates the proposed methodology's structure.

Figure 41: Overview of the methodology



Source: Elaborated by the author.

4.1.1 New AI application idea

The methodology flow starts with a new AI application idea to be developed. In this case, we are considering ideas that have already passed through the organization's internal flows for innovation and artificial intelligence demands and have already been approved by higher authorities such as committees or working groups, composed of Product Owners (POs) and stakeholders, in other words, they are ideas ready for development. This step is essentially the responsibility of the Business Team. It is not part of the scope of this work to define the approval flow of an AI application idea.

4.1.2 Application objectives and success metrics

Based on a duly approved AI application idea, our methodology foresees the definition of application objectives and success metrics. Therefore, at this stage, the Business Team needs to define the problem to be solved or the opportunity for improvement. A question typically asked at this stage is "What are the pains?".

In addition to clarity regarding the problem to be solved, this stage involves the definition, by the business area, of the objectives to be achieved and the project's success metrics.

Note that up until this stage, the IT Team has not yet participated in the project, except occasionally to clarify some technical point. It is important to emphasize that the methodology proposes the Business Team's commitment to the project in a way that is usually not what they expect. We have often seen projects where the area requesting the solution makes the "request", but, in fact, does not get involved with the project in a more intense way. This may also happens in traditional projects, but, as we have seen, it is a more impactful factor when we deal with AI projects.

4.1.3 Method and evaluation dataset

As we detail further in AI and Law application section, for an artificial intelligence application to be successful, it is not enough to simply choose a well-trained AI model, but it involves specific application developments, based on the problem to be solved and the business requirements. By a well-trained AI model we mean a model that performs well on benchmark datasets, such as MMLU (HENDRYCKS et al., 2021), HellaSwag (ZELLERS et al., 2019) and AI2 Reasoning Challenge (ARC) (CLARK et al., 2018), for example.

In general, what the end user of an AI application applied to Law wants is for some automation to occur in the judicial process or an automatic prediction, analysis, classification, information extraction applied to a legal proceeding, and AI is often the technological tool to support these tasks. Therefore, for the solution to be effective, it is necessary for the evaluation of the solution to be carried out with specific datasets built on the organization's data aligned with the application objectives, which we call application datasets. For example, to evaluate the quality of an application that classifies judicial processes into STJ or STF themes, it is necessary to evaluate the solution with a dataset that has a selection of documents and respective themes to be classified. The same applies to classifying a judicial process into judicial subjects or classes and so on.

Some differences between benchmarking and application datasets are:

a) Purpose of use

   i. Benchmarking dataset is used to evaluate and compare the performance of algorithms or models in specific tasks. It is designed to be a reference standard, allowing different models to be tested under similar conditions.

   ii. Application dataset is used to evaluate a model in a specific application context. It reflects the real data that the model will encounter in Production, being adapted to the needs and characteristics of the problem it wants to solve.

b) Generality versus specificity

   i. Benchmarking dataset generally composed of standardized data that is widely used in the research community, covering a variety of scenarios to ensure that the models are robust and generalizable.

   ii. Application dataset is specific to the domain or application in question, and may contain peculiarities and particular characteristics of the data that will be used in practice. This dataset tends to be more representative of the real environment in which the model will be deployed.

The application dataset that will be used to evaluate the project is developed jointly by the Business and IT teams. The Business Team participates in the selection and definition of the

data, and the IT Team develops scripts to generate the datasets, when necessary. From this stage onwards, therefore, the proposed methodology foresees the direct involvement of the IT Team in the project.

For each AI application to be developed, in addition to the dataset, a method for evaluating the experiments must be defined. The use of scientific methods is encouraged, but not mandatory, since they can generate additional effort for a work that probably would not be published and the time to make the application available is generally short. The most important is that the evaluation, in fact, is able to attest to the quality of the evaluated solution for its application in the real world.

Whenever possible, common automatic metrics such as accuracy and precision, or more specific ones such as ROUGE (LIN, 2004), BLEU (PAPINENI et al., 2002), etc. should be used. In certain situations, when automatic metrics are not sufficient to confirm the quality of the solution, strategies such as LLM-as-a-judge(ZHENG et al., 2023) or even human assessment can be used.

A critical issue addressed in this step is the handling of hallucinations: coherent but factually incorrect outputs generated by language models. Especially in legal applications, hallucinations may compromise the reliability of the system and even produce ethically problematic outcomes. For this reason, the methodology emphasizes transparency, validation through reliable datasets, and, when necessary, human-in-the-loop mechanisms.

In selecting the final model, cost-benefit analysis is also essential. For instance, although GPT-4o may outperform a smaller model in accuracy, token costs or processing latency may justify the use of GPT-4o-mini if it achieves satisfactory results.

4.1.4 Experimentation

As it is generally not possible, unlike the development of traditional applications, to guarantee the technical capability of a solution, we propose an experimentation stage, through the prototyping of rapid solutions, also called POCs (Proof of Concepts) or MVPs (Minimum Viable Products), to evaluate the solution before its final development.

At this stage, the IT team is directly involved to develop pre-versions of the application as POCs and MVPs to assess the quality of the solution and its adherence to business requirements. The Business Team participates as an evaluator of the pre-versions developed.

The experiments developed in this stage do not necessarily need to be functional applications, but could be just reports or spreadsheets generated by some procedure or script, for example. In cases where it is necessary to develop a simple and quick prototype application, we recommend the use of market tools such as Streamlit[1] and Gradio[2], among others. We also

[1] https://streamlit.io/
[2] https://www.gradio.app/

suggest the use of Jupyter notebooks[3] or Google Colaboratory[4] as support tools in this stage.

Each experiment must be evaluated with the method and evaluation dataset defined in the previous stage to verify whether it meets requirements expected by the business area. If it does not, the IT Team will develop a new pre-version, using other technological strategies with the objective of achieving the defined success metrics. In this stage, the Business Team participates as an evaluator of the experiments and is responsible for deciding whether the pre-version is approved and should proceed to the development of the final version of the solution and its deployment in Production. Thus, at the end of the experimentation stage, we have an approved technical solution ready to be updated to a real Production application.

4.1.5 Final version development

At this stage, the IT Team develops a real application based on the technical solution used in the approved pre-version. In general, it involves the inclusion of technical requirements such as security, performance, scalability, logging, among others. The proposed methodology does not foresee the involvement of the Business Team from this step, but may occasionally occur to clarify business doubts, such as how the application will be included as a functionality in the legal proceedings system.

The final version must be aligned with Resolution No. 615/2025 of the National Council of Justice (CNJ, 2025), which establishes comprehensive guidelines for the development, governance, auditability, monitoring, and responsible use of artificial intelligence systems in the Brazilian Judiciary. The resolution emphasizes ethical principles, transparency, human oversight, and data governance as foundational elements in the design and operation of AI solutions. In this context, AI and Law applications should ensure the persistence of all inferences made by AI models, including inputs, outputs, and relevant metadata, to support future analyses, monitoring, and audits. Additionally, we recommend logging processing times, tokens consumed, and associated computational costs to enable a complete and transparent cost-benefit analysis of the solution over time.

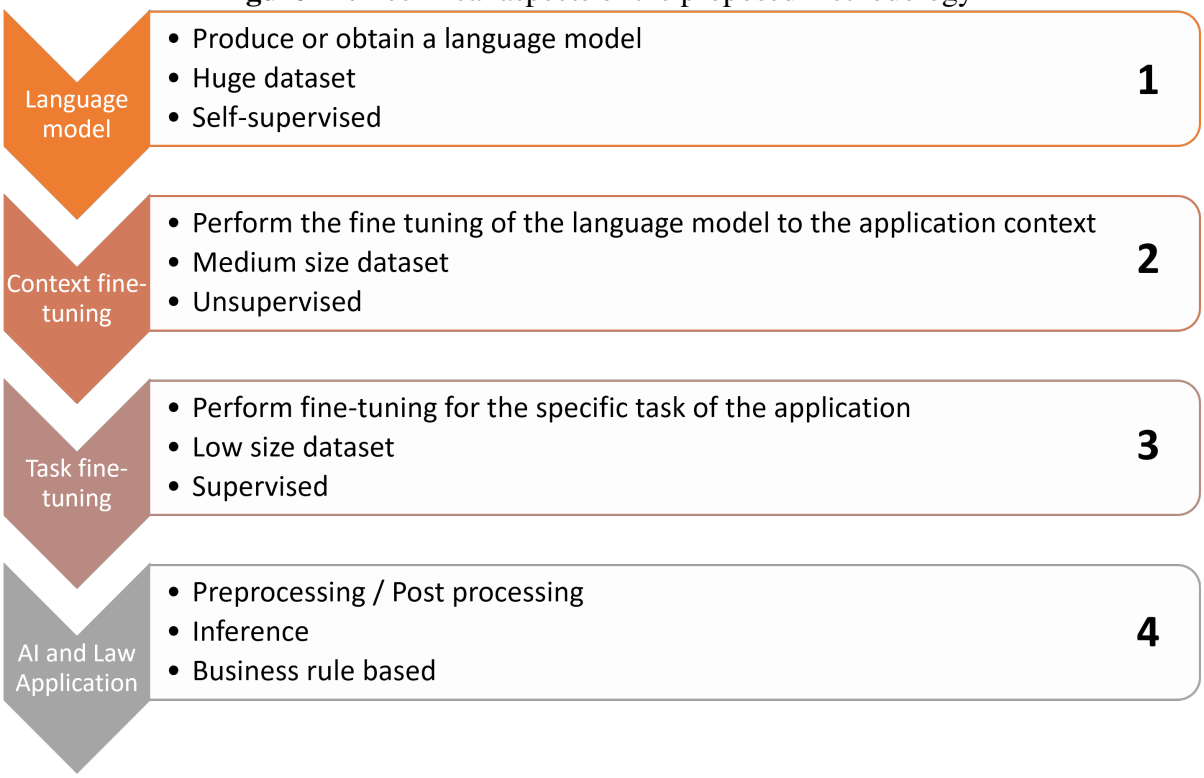
When using internally trained models, institutions must maintain complete version control and documentation of training datasets, preserving reproducibility and supporting audits in line with ethical AI principles.

4.1.6 Deploy to production

The final stage is the deployment of the solution to the production environment. This is a technical responsibility of the IT Team and involves ensuring that the system is integrated, stable, and accessible to users. Post-deployment monitoring is essential to identify model drift,

[3] https://jupyter.org/
[4] https://colab.research.google.com/

Figure 42: Technical aspects of the proposed methodology



Source: Elaborated by the author.

operational issues, or new risks such as hallucinations under unexpected inputs.

4.2 AI and Law application development

In this section, we describe the technical aspects of the proposed methodology, which must be considered by the IT Team to develop the experiments and the final version of the AI and Law application. Figure 42 illustrates the technical aspects of the methodology, divided into four layers. Each layer is explained in the following subsections.

4.2.1 Language model

The first layer involves producing or obtaining a pre-trained language model (PLM) able to process texts in Portuguese, also known as foundation models(BOMMASANI et al., 2022; BAI et al., 2023). This step is entirely dependent on the language, since PLMs are pre-trained in language specific texts, thus the availability and quality of these models is language dependent. Some languages like English and German have much more and bigger language models already available. At this moment, in Portuguese there are models in BERT (DEVLIN et al., 2019), GPT-2 (RADFORD et al., 2019) and T5 (RAFFEL et al., 2019) transformer-based architectures. More recently, with the emergence of Large Language Models (LLMs), several models are being trained with texts from different languages, called multilingual models (QIN et al., 2024),

which also can be used in this layer of the proposed methodology. Examples of multilingual LLMs are mT5(XUE et al., 2021), Mixtral(JIANG et al., 2024), LLama(TOUVRON et al., 2023), GPT-4(OPENAI et al., 2024), among very others.

The production of a language model is not something trivial and involves training models on large amounts of raw text in a self-supervised way, demanding large computational resources and substantial energy consumption, typically on equipment with Graphics Processing Unit (GPU) or Tensor Processing Unit (TPU). As a result, these models are costly to train and develop, both financially and environmentally.

The general strategy to achieve better performance is by increasing the model's sizes as well as the amount of data they are pretrained on. This could be a reason for training new language models. Another reason is due to the emergence of new architectures that still do not have pre-trained models in Portuguese.

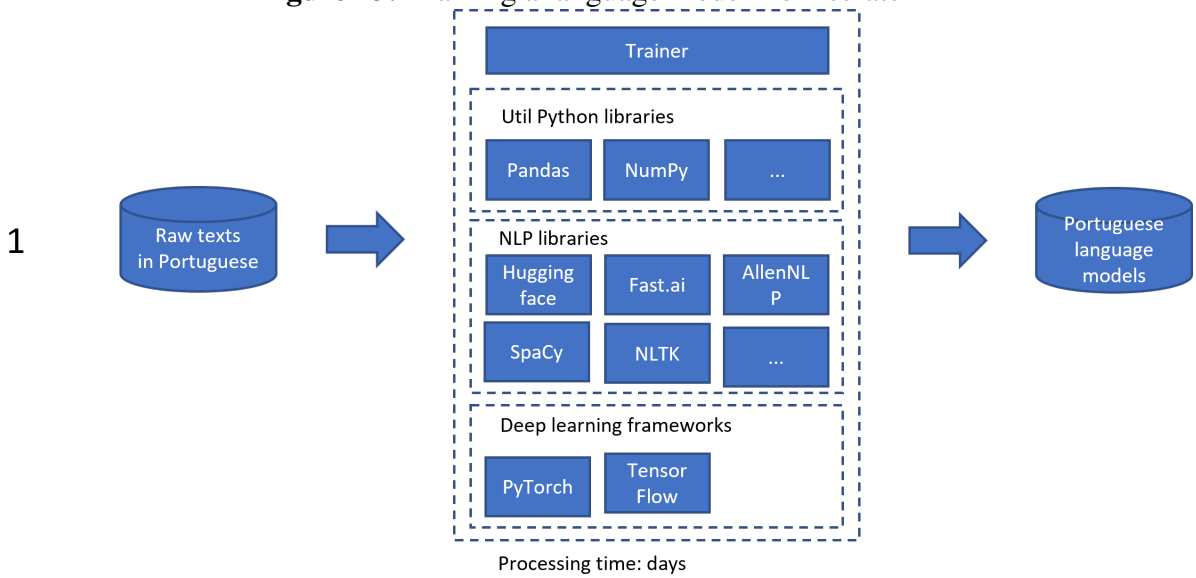
Thus¸ in general, it is not necessary to produce new foundation language models in Portuguese. It is only mandatory if we want to use language models in architectures that do not yet have models available in Portuguese. As mentioned before, the following are examples of foundation models in Portuguese: BERT, in base and large versions, T5, in small, base and large versions, and GPT-2, in the small version. One example of a LLM foundation model in Portuguese is Sabiá(PIRES et al., 2023). Therefore, if some application needs to use architectures such as GPT-3 (BROWN et al., 2020) or Megatron (SHOEYBI et al., 2020), it will be necessary to generate the language model in the respective architecture.

Another reason to train a language model from scratch would be if we wanted to use a larger amount of data in its training. The BERT models currently available in Portuguese were trained using the Brazilian Portuguese brWaC Corpus (WAGNER FILHO et al., 2018), which was recently made available with a resulting corpus including 145 million sentences and 2,7 billion tokens, based on 38 million Uniform Resource Locators (URLs) of .br top-level domain pages. BERTimbau (SOUZA; NOGUEIRA; LOTUFO, 2020) is the first pre-trained BERT model for Brazilian Portuguese. It achieves SOTA performances on three downstream NLP tasks: Named Entity Recognition, Sentence Textual Similarity, and Recognizing Textual Entailment.

Figure 43 illustrates the components needed to train a language model from scratch. In this case, the input is raw texts in Portuguese and the output is a Portuguese language model. The processing is performed by a trainer component, usually developed using a Python technology stack, including from the basis to the top: a deep learning framework, usually PyTorch or TensorFlow; NLP libraries as HuggingFace, Fast.ai, AllenNLP, SpaCy, etc., and others util Python libraries like Pandas, NumPy, etc. The figure is not comprehensive and does not include all the software components needed and technical details for training a language model, like the tokenizer, which is a very important component to do this job, for example.

Training a model from scratch could directly use texts from a specific context, such as legal, but this is not very common. In general, training a language model in a specific context is done through context fine-tuning, detailed in the section 4.2.2.

Figure 43: Training a language model from scratch



Source: Elaborated by the author.

4.2.2 Context fine-tuning

The second layer involves performing the fine-tuning of the pre-trained language model in the application context. The objective is to improve a general Portuguese language model with the nuances of the application context where it will be used. For example, Law field applications should be context fine-tuning with texts from court proceedings. More specifically, if the application where the final model will be used is going to process lawyer texts, the language model should be fine-tuned with petition texts. On the other hand, if the application will process texts written by judges, the language model should be fine-tuned with texts from orders and sentences. Or both, if the application will process texts from lawyers and judges.

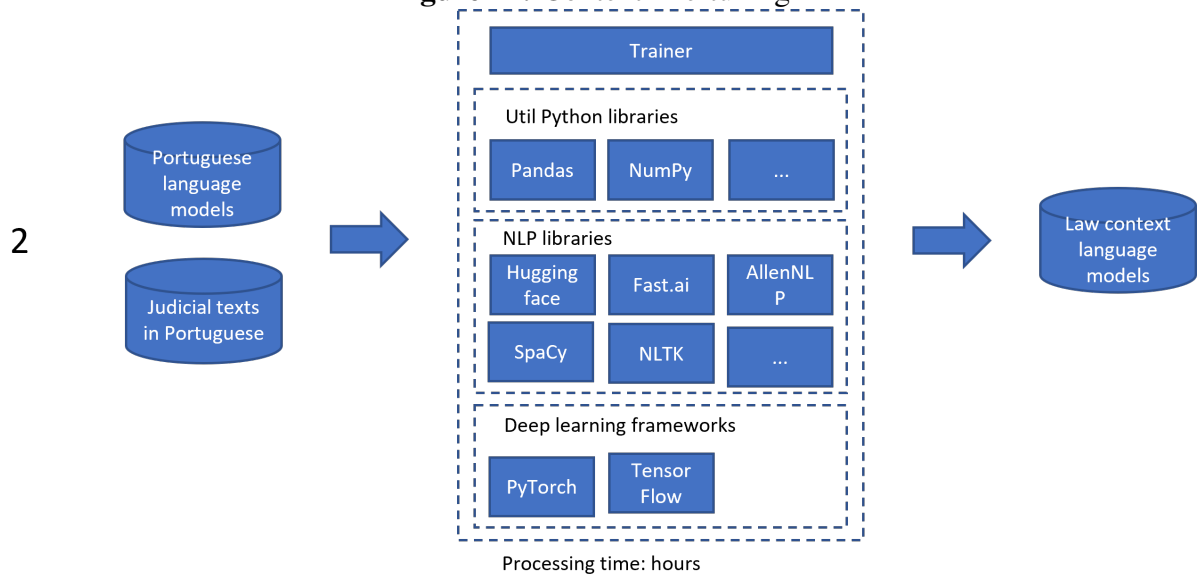
In this layer, the language model is also trained in an unsupervised way, typically with medium size amounts of raw text. Training does not require the same processing power as the previous layer, but still requires substantial effort, making use of machines with GPU or TPU.

This is an optional step, but it can improve the results of the NLP downstream tasks. (BONIFACIO et al., 2020) fine-tuned the BERT Mutilingual and ELMo brWaC language models on the Acordaos TCU corpus[5]. BertBR (CIURLINO, 2021) also performed additional BERT training for the legal language, by fine-tuning BERTimbau (SOUZA; NOGUEIRA; LOTUFO, 2020) on the Iudicium Textum Dataset (WILLIAN SOUSA; FABRO, 2019).

This step is not Law domain specific. It could be used in every NLP application context. BioBERTpt (Portuguese Clinical and Biomedical BERT) (SCHNEIDER et al., 2020), transferred learned information encoded in a multilingual-BERT model to a corpora of clinical narratives and biomedical-scientific papers in Brazilian Portuguese (OLIVEIRA et al., 2022).

[5]https://www.kaggle.com/ferraz/acordaos-tcu.

Figure 44: Context fine-tuning



Source: Elaborated by the author.

The same way that language models trained from scratch in a given language such as BERTimbau can be made available and reused easily in applications, which is one of the great benefits of language models, fine-tuned models trained with context-specific data can also be made available and reused. For instance, Luciano/gpt2-small-portuguese-finetuned-tcu-acordaos[6] and Luciano/gpt2-small-portuguese-finetuned-petições[7] are GPT2 Portuguese models fine-tuned respectively on Acordaos TCU corpus and on a set of petitions from lawyers, and Luciano/bert-base-portuguese-cased-finetuned-tcu-acordaos[8] and Luciano/bert-base-portuguese-cased-finetuned-peticoes[9] are BERT Portuguese models fine-tuned respectively on the same data as above.

Figure 44 illustrates the second layer of the proposed methodology, with the components needed to fine-tune a language model to a domain context. In this case, the input is raw judicial texts in Portuguese along with the Portuguese language model from the first step. The output is a Portuguese language model in the judicial context. The processing is performed by a trainer component the same or very similar to the first layer.

4.2.3 Task fine-tuning

The third layer of the proposed architecture is the fine-tuning of the pre-trained language model in a specific NLP downstream task, according to the application needs. Examples of NLP tasks are Sequence Classification, Extractive Question Answering, Text Generation, Named

[6]https://huggingface.co/Luciano/gpt2-small-portuguese-finetuned-tcu-acordaos.
[7]https://huggingface.co/Luciano/gpt2-small-portuguese-finetuned-peticoes.
[8]https://huggingface.co/Luciano/bert-base-portuguese-cased-finetuned-tcu-acordaos.
[9]https://huggingface.co/Luciano/bert-base-portuguese-cased-finetuned-peticoes.

Entity Recognition, Text Summarization, Machine Translation, etc. The input of this layer is the pre-trained model outputted in one of the previous layers, i.e., the input could be the Law context-specific LM from the layer two or the general Portuguese or multilingual LM from the layer one.

In this layer, we train a model with supervised data and usually with low size dataset. We consider this as a mandatory step only for non large language models, like BERT or T5, due to the fact that Law field is very specific, and the applications are very specialized and must have high accuracy to be effective in the real world. This step is not mandatory for LLMs, given the improved capabilities of these models, since they can be effective in a wide variety of tasks using zero-shot learning (PUSHP; SRIVASTAVA, 2017). However, in this case of not performing task fine-tuning for LLMs, the next step will require intensive and accurate prompt engineering work to "teach" the model about the task to be performed. Strategies like Retrieval-Augmented Generation (RAG) (GAO et al., 2024) and few-shot learning (WANG et al., 2020) could be used to compensate for not training at the task level, that is mandatory for smaller language models.

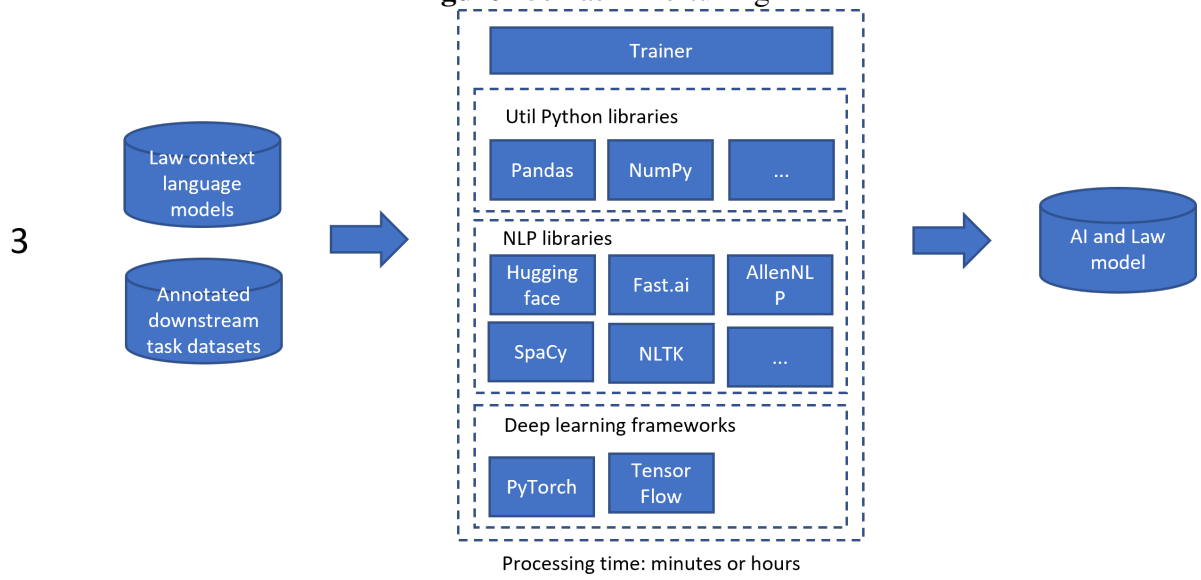
LeNER_Br (ARAUJO et al., 2018) is an example of an annotated dataset to be used in this step for NER task. Another example would be an annotated dataset for the machine translation task, with examples of the translation of judicial language sentences to simple language sentences, to be used for the development of applications to support the use of a language more accessible to the common citizen, which is being strongly encouraged by the CNJ (CNJ, 2015).

The same way that models trained in the previous steps, the model trained in this step can be shared and reused by applications that have the same goals and context of use. For instance, the models Luciano/bertimbau-large-lener_br[10] and Luciano/bertimbau-base-lener_br[11] could be used by courts or law firms that have the need of doing named entity recognition in judicial texts in Portuguese.

The third layer of the methodology is illustrated in Figure 45, that shows the components needed to fine-tune a language model to a downstream task, like sentiment analysis, named entity recognition or summarization. This time, the input is not raw texts, but an annotated dataset according to the NLP downstream task to be trained. The other input is the Portuguese language model from the first step or the Portuguese judicial context language model from the second step, if it exists. The output we call as AI and Law model, which means a judicial NLP downstream task model in Portuguese. The processing is performed by the same trainer component of the second step.

[10]https://huggingface.co/Luciano/bertimbau-large-lener_br.
[11]https://huggingface.co/Luciano/bertimbau-base-lener_br.

Figure 45: Task fine-tuning



Source: Elaborated by the author.

4.2.4 AI and Law application

The final layer is the use of the pre-trained model from the previous layer in an application, which means the use of an AI model for inference. For example, in a NER task, this is the process of using the fine-tuned model from the third layer to make a prediction and use this prediction in conjunction with additional processing in the application context to add business value.

Although training is computationally very expensive and is best accelerated with GPUs, inference is less computationally intense, but typically still benefits from a GPU due to the massive amount of data GPUs can process concurrently. To enable the inference strictly on CPU, works on optimizing inference performance of transformer-based model on Central Processing Unit (CPU) are being done (DICE; KOGAN, 2021; HSU et al., 2020). In training, a great quantity of inputs examples, often in large batches, are used to train a deep neural network. In inference, the trained network is used to discover information within new inputs that are fed through the network in smaller batches. The training is the phase in which the computer essentially tries to learn to do a job from data examples, while the inference is the phase of doing the job learned from training.

Inference is a bit different for LLMs, since these large models are accepting much larger inputs, called context window, than smaller models accept. Currently, some LLMs are accepting context length as large as 128k or more tokens[12][13], allowing entire books to be submitted in a single inference call. Some models are even allowing infinite context windows (MUNKHDALAI; FARUQUI; GOPAL, 2024).

[12]https://platform.openai.com/docs/models
[13]https://artificialanalysis.ai/leaderboards/models

Figure 46: Flow of the final step of the proposed methodology

Source: Elaborated by the author.

This stage must be aligned with ethical and legal requirements, particularly Resolution No. 615/2025 of the CNJ (CNJ, 2025), which mandates transparency, explainability, and traceability in AI systems used by the judiciary. Therefore, all inferences must be logged, including inputs, outputs, timestamps, and model versions, enabling later auditing and ensuring accountability.

Furthermore, developers must implement mechanisms to detect and mitigate hallucinations, especially when summarizing or extracting sensitive legal information. Where automated safeguards are insufficient, the methodology recommends incorporating human oversight or using alternative strategies such as LLM-as-a-judge (ZHENG et al., 2023).

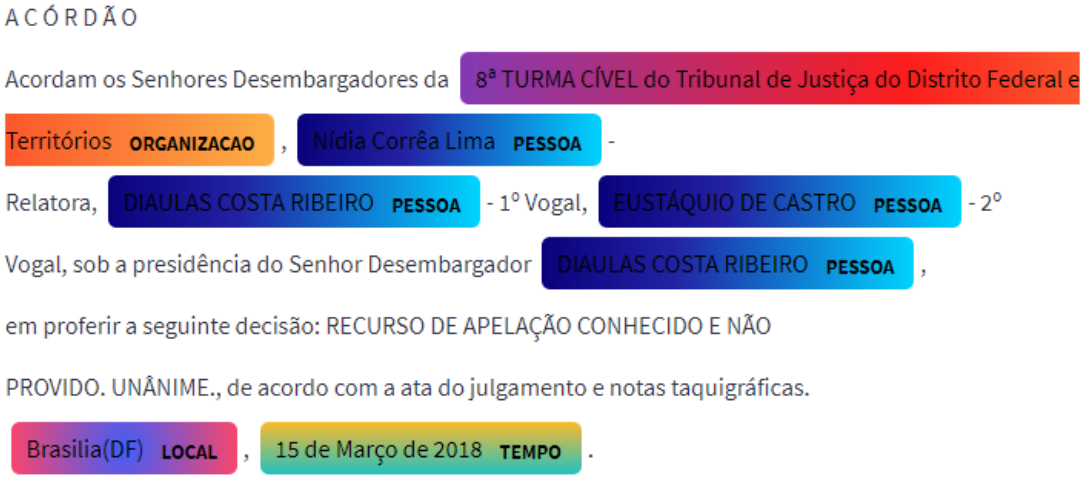
This final layer of the methodology is business rule based, which means that we must understand the user goals and the applied business context to be capable to deliver the right solution. This goes beyond the simple inference from an AI model. The importance of this phase includes minimizing the scope of the model by removing any parts of the text not necessary to make the prediction. Figure 46 presents the flow of the final step of the proposed methodology.

Although the language models in transformers architecture like BERT decrease the importance or even eliminates the NLP preprocessing (ÖZçIFT et al., 2021), the preprocessing is still needed in business context, for instance, removing headers and footers or splitting document text into sentences. This is even less important for large language models.

This step includes preprocessing the documents and post processing the AI result obtained by model inference. Typical NLP preprocessing could involve the following items:

- Remove Hypertext Markup Language (HTML) tags;
- Remove extra whitespaces;
- Convert accented characters to American Standard Code for Information Interchange (ASCII) characters;
- Expand contractions;
- Remove special characters;
- Lowercase all texts;
- Convert number words to numeric form;
- Remove numbers;

Figure 47: NER response with postprocessing



Source: Elaborated by the author.

- Remove stopwords;
- Lemmatization.

Nevertheless, not all these preprocessing items have to be done for all applications. Each case must be analyzed and only items appropriate to the business context should actually be applied. For example, lowercase judicial texts probably would not be a wanted preprocessing item to be done before the inference.

Another important preprocessing issue refers to limit of the model input length, so that the model cannot process long pieces of text. BERT and others transformer models consume at most 512 tokens, truncating anything beyond this length. Considering that BERT uses a subword tokenizer, the limit of words to be accepted as input would be much smaller than this. The emergence of LLMs, as mentioned above, has changed this limitation considerably.

On the other hand, the postprocessing is important because not always the AI response is ready to be consumed by software users or even the postprocessing could improve the results applied in business scenario. For example, Figure 47 and Figure 48 shows, respectively, a NER model response with and without postprocessing.

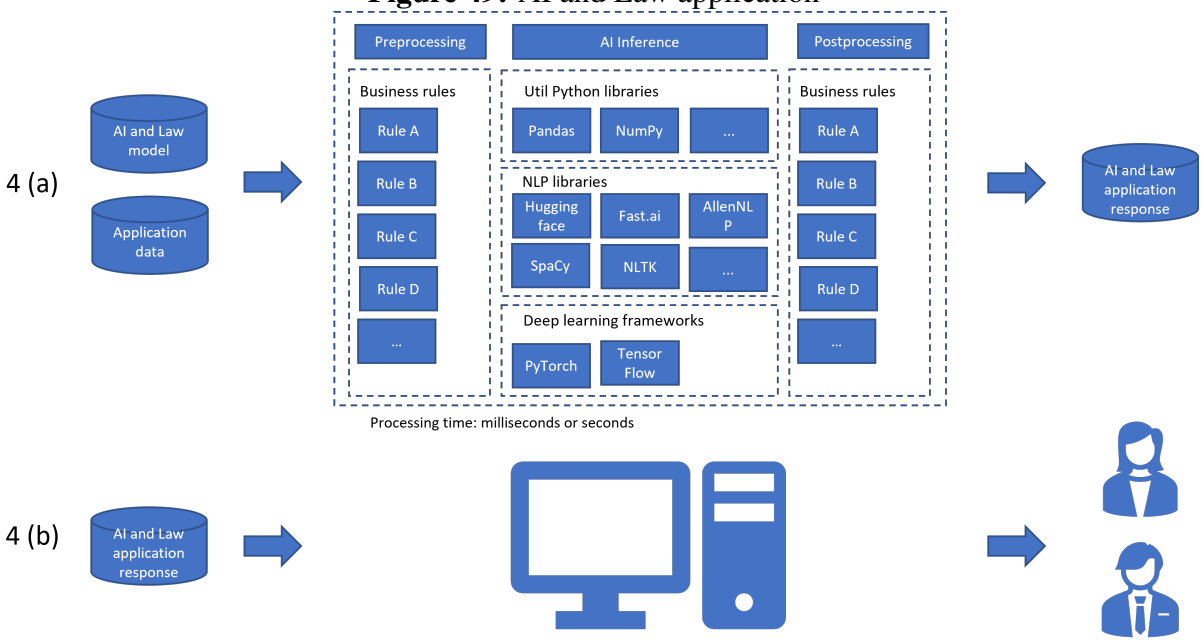
The fourth and last layer of the methodology is illustrated in Figure 49. The part (a) of the figure illustrates the components needed to perform the AI inference. In this case, the input are the AI and Law model from the third step, and the application data, i.e., texts from an application being used by the users. The output we call as AI and Law response, which means the AI result to be consumed by the users. The processing in this step contains three elements: preprocessing, AI inference and postprocessing. The preprocessing and postprocessing contains the business rules needed to process the input texts and the AI inference response, respectively. The AI inference component could be very similar to the trainer component from the previous steps. It could integrate the application architecture or could be built separated as an API and be called by the application.

Figure 48: NER response without postprocessing

A C Ó R D [UNK] O Acordam os Senhores Desembargadores da [UNK] TURMA CÍVEL do Tribunal de Justiça do Distrito Federal e Territórios ORGANIZACAO , N PESSOA ídia Corrêa Lima PESSOA - Relatora, D PESSOA IA PESSOA UL PESSOA AS COSTA RIBEIRO PESSOA - [UNK] Vogal, E PESSOA US PESSOA T PESSOA Á PESSOA Q PESSOA U PESSOA IO DE CASTRO PESSOA - [UNK] Vogal, sob a presidência do Senhor Desembargador D PESSOA IA PESSOA UL PESSOA AS COSTA RIBEIRO PESSOA , em proferir a seguinte decisão : RECURSO DE APELAÇÃO CONHECIDO E NÃO PROVIDO. UNÂNIME., de acordo com a ata do julgamento e notas taquigráficas. Brasil LOCAL ia (DF) LOCAL , 15 de Março de 2018 TEMPO .

Source: Elaborated by the author.

Figure 49: AI and Law application



Source: Elaborated by the author.

The part (b) illustrated in Figure 49 shows the human consumption from the AI, generally occurring through the use of web or mobile applications. This is not always the case, because artificial intelligence is often used for machine processing and automation, with no directly human consumption.

4.3 Example set of components

Besides the steps of the proposed methodology for building AI and Law applications, this work also presents a view of an example set of components for the methodology, illustrated in Figure 50. These components are divided in the following layers: data, model, context adaptation, task, application, and extensions. The figure also presents the methodology layers, numbered as 1 to 4, for reference and to facilitate the understanding. New components will continue to be developed and made available in future, with the aim of providing an increasingly complete methodology and a range of components suitable for the development of AI and Law applications.

We call the basis of the stack as data layer, which contains the Portuguese corpora that is the base to build the language model in Portuguese. Theses corpora are basically raw data in Portuguese, usually scrapped from one or more sources on the Internet. The more notorious is Brazilian Portuguese brWaC Corpus (WAGNER FILHO et al., 2018), with a resulting corpus including 145 million sentences and 2,7 billion tokens, based on 38 million URLs of .br top-level domain pages. This is the corpus used for training BERTimbau (SOUZA; NOGUEIRA; LOTUFO, 2020) and PTT5 (CARMO et al., 2020) models, which are, respectively, BERT (DEVLIN et al., 2019) and T5 (RAFFEL et al., 2019) models trained in Portuguese language.

Another important resource in this layer is OSCAR (Open Super-large Crawled Aggregated) corpus, which is a huge multilingual corpus obtained by language classification and filtering of the Common Crawl corpus using the Ungoliant architecture (SUáREZ; SAGOT; ROMARY, 2019). The Portuguese data size is 124 Gigabyte (GB), reducing to 64 GB after deduplication the files with a simple non-collision resistant hashing algorithm. The Portuguese corpus contains a total of 20,641,903,898 words and 10,751,156,918 after the deduplication process. This layer is the input of the first step of the proposed methodology.

The model layer contains the pre-trained language models in Portuguese built on a transformer-based architecture. In this layer, we have BERTimbau, PTT5 and GPT-2-small (GUILLOU, 2020) models. More pre-trained models on other architectures or trained on different corpus are needed. For instance, there are no PLM in Portuguese trained on the OSCAR corpus. This layer is the output of the first step of the proposed methodology and part of the input of the second step.

The next layer refers to judicial resources to be used to context adaptation in a further fine-tuning process. This means to give the language model more knowledge in judicial texts, since judicial text are not usual in the common sources on the Internet like Wikipedia or Common

Figure 50: Stack of the methodology components

Source: Elaborated by the author.

Crawl. This layer contains data and models, which means that we can both use directly a judicial context adapted language model in Portuguese for fine-tuning in a downstream task (next layer), what is preferred, as well as a judicial corpus in Portuguese to train our own language models adapted to the judicial context, and then use these new models for fine-tuning in a downstream task. Are examples of judicial data the Acordaos TCU corpus[14], the Iudicium Textum Dataset (WILLIAN SOUSA; FABRO, 2019) and a private lawyer's petition dataset used in some experiments. Are examples of judicial context models the four models resulted by training in the experiments described in sections 5.2 and 5.3. The data in this layer along with the model from the previous layer are the input of the second step of the proposed methodology. On the other hand, the model in this layer is the output of the second step and part of the input of the third step.

In the task layer there are the NLP downstream task resources, i.e., models ready to use in an NLP downstream task, as named entity recognition or summarization, for example, and annotated datasets ready to use for finetuning in a specific task. LeNER_Br (ARAUJO et al., 2018) is an example of annotated dataset to be used for the NER task. Ideally, whenever possible, one should use models directly from this layer for a greater reuse of models, avoiding the need to train new models for every application. The data in this layer and the model from the previous layer are the input of the third step of the proposed methodology. On the other hand, the model in this layer is the output of the third step and part of the input of the fourth step.

We also performed a series of LLM fine-tuning experiments including Parameter-Efficient Fine-Tuning (PEFT) (XU et al., 2023) methods like LoRA (HU et al., 2021), QLoRA (DETTMERS et al., 2023), $(IA)^3$[15] P-tuning[16], Prompt-tuning[17] and Prefix-tuning[18], described in more detail in section 5.5. All trained models are publicly available on Hugging Face[19]. Some experiments were context fine-tuning and others were task fine-tuning, that is, they are components of the Context adaptation and Task layers.

The application layer contains the business rules to be applied in preprocessing for the input texts and in the postprocessing for the response of the model inference. This is very context and application specific, but there may be some common patterns to be used in general. This is important for the effectiveness of the application in the real world, on which we intend to measure with a novel metric to be proposed. The application data and the downstream task model from the previous layer are the input of the final step of the proposed methodology, along with the business rules. The output is the AI and Law application response.

We are also proposing three extensions in the form of optional components, i.e., not mandatory, but recommended for AI and Law applications: explainable AI, judicial ontologies and a novel AI and Law Application metric. In the future, more extensions may be aggregated. An

[14]https://www.kaggle.com/ferraz/acordaos-tcu.
[15](LIU et al., 2022)
[16](LIU et al., 2023b)
[17](LESTER; AL-RFOU; CONSTANT, 2021)
[18](LI; LIANG, 2021)
[19]https://huggingface.co/Luciano

explainable AI (ADADI; BERRADA, 2018) component is being proposed as it is a very necessary aspect on AI applied to Law applications (GUNNING et al., 2019). We also propose the use of ontologies (NOY; MCGUINNESS, 2001) to include in the language models the formal explicit description of concepts in the Law domain, and a novel evaluation metric in the application level, with focus on the real task to be executed by the users. The explainable AI and the ontologies extensions will be explored in future, after this doctorate work.

For AI and Law Application metric extension, as we described in section 4.1, we propose, for each application to be developed, the need of definition of application objectives and success metrics and an application dataset to evaluate pre-versions of the application so that it is possible to choose the appropriate AI technical solution to achieve the business objectives.

4.4 Methodological contributions

The proposed methodology contributes to the development of effective, trustworthy, and high-quality AI applications in the legal domain by structuring the development process around recent advances in Natural Language Processing (NLP) and aligning technical practices with the specific demands of legal tasks. Its contributions can be summarized as follows:

a) A structured and general methodology for developing AI and Law applications, grounded in modern NLP paradigms such as transformer-based architectures, pre-trained language models, transfer learning, and domain adaptation;

b) A modular development flow that explicitly incorporates legal-specific elements and distinguishes four methodological steps (from model preparation to application deployment), enabling reproducibility, scalability, and clearer task delegation;

c) A development framework that integrates both technical and domain-specific knowledge by emphasizing the active participation of legal domain experts (Business Team) alongside the IT Team, especially in the definition of objectives, datasets, and evaluation criteria;

d) The formalization of the need to define application-level datasets and success metrics from the outset, aligning AI model outputs with real-world legal goals and ensuring measurable impact;

e) The inclusion of pre-deployment stages such as Proofs of Concept (POCs) and Minimum Viable Products (MVPs), supporting iterative validation of the technical solution in authentic legal scenarios before full-scale deployment;

f) A set of reusable and domain-adapted components (e.g., datasets, models, evaluation strategies) organized into methodological layers, providing guidance and resources for concrete implementation;

g) The recommendation of explainable AI (XAI) techniques as an integral part of the methodology, especially relevant for ensuring transparency and auditability in legal systems;

h) A novel proposal for application-level evaluation metrics, which extend beyond traditional AI evaluation (e.g., accuracy, F1-score) by including pre and post-processing and task-specific business logic;

i) An integrative approach between NLP and linguistics through the proposed use of legal ontologies, supporting semantic consistency and enhancing domain adaptation of language models.

These contributions are interrelated and mutually reinforcing. For example, the explicit modeling of development steps (item b) and the emphasis on cross-disciplinary collaboration (item c) are essential to ensuring that the methodology is not only technically sound but also legally relevant. Similarly, the use of reusable components (item f) and the adoption of evaluation practices that reflect actual legal needs (items d, e, and h) contribute directly to improving the practical effectiveness and reliability of AI applications in Law.

The inclusion of explainable AI mechanisms and attention to hallucination risks are essential ethical elements of the methodology, ensuring that AI outputs in legal contexts are interpretable, auditable, and aligned with judicial standards of responsibility and transparency. These features strengthen the reliability of AI applications in high-stakes legal environments and reflect the current regulatory and societal demands for trustworthy AI.

4.5 Discussion

This chapter detailed the proposed methodology for building AI applied to Law applications. The proposal is based on the recent advances on NLP, which consider the use of transformer-based architecture, pre-trained language models and transfer learning as the common approach to developing SOTA applications. The methodology also includes the use of LLMs, which emerged during the development of this work.

The methodology contains aspects to develop NLP applications in general, and specific components and look at the judicial field to focus on AI applied to Law applications. The main contributions are explained in the previous section and contemplates a methodology and the benefits arising from its use, like easier development and more expected quality of applications, components ready to use as models and datasets, improved reliability arising from the use of XAI and foresees future innovations like the integration of ontologies and transformers and a novel metric to measure the accuracy in the application level.

The proposed methodology also details the development flow of AI applications applied to Law, highlighting the need for the Business Team to participate during the development process. This aspect is necessary for the final result of the application to reach success metrics. This commitment of the Business Team with the development of the application is especially important

because artificial intelligence applications are not known in advance the best technical solution for them, nor whether it will meet the accuracy requirements expected by the business area. Therefore, the participation of the Business Team during the process is mandatory, according to our proposed methodology.

The center of the development flow proposed by the methodology includes the joint participation of the Business Team and the IT Team in the "Method and evaluation dataset" and "Experimentation" stages. This shows that the core of the proposed methodology flow includes building application level datasets and defining evaluation metrics to support the iterative development of POCs and MVPs and their respective technical solution evaluation.

The proposed methodology is aligned with Resolution No. 615/2025 of the National Council of Justice (CNJ, 2025), which establishes comprehensive guidelines for the development, governance, auditability, monitoring, and responsible use of Artificial Intelligence in the Judiciary. This resolution replaces and updates the former Resolution No. 332/2020 (CNJ, 2020), expanding its scope to include recent advances such as Large Language Models (LLMs) and other generative AI systems. With respect to explainability, the resolution requires that AI systems provide, whenever technically feasible, clear and sufficient explanations to ensure human auditability and interpretation of outputs. Additionally, it sets explicit rules for the use of generative AI within the Judiciary, including restrictions on the use of external or privately contracted models by judges, civil servants, or third parties, unless explicitly authorized by the CNJ's Artificial Intelligence Committee. These provisions reinforce the need for transparency, accountability, and ethical oversight in the adoption of AI technologies within judicial processes.

By integrating ethical concerns and concrete technical safeguards into the development process, from corpus selection to inference, we aim to ensure that AI and Law applications are not only accurate but also trustworthy and aligned with the fundamental values of the legal system.

5 EXPERIMENTS

"The true method of knowledge is experiment."
William Blake

The proposed methodology has already been validated through a series of experiments, which together demonstrate the feasibility and effectiveness of the approach in real-world legal AI applications. These experiments cover all the main layers and steps described in the methodology, from data acquisition and model training to contextual adaptation, downstream task finetuning, and application-level integration , thus confirming the applicability and coherence of the entire methodological framework.

This chapter presents the experiments conducted to date, including a peer-reviewed study accepted at the International Conference on Computational Processing of Portuguese (PROPOR 2022), described in Section 5.1. It is important to note that the manuscript submitted to PROPOR 2022 was written in an earlier stage of this research. Therefore, some portions of that text may reflect the technological context and terminology available at that time, which might now appear outdated due to the rapid evolution in the field of natural language processing, particularly with the advent of more advanced large language models and fine-tuning techniques. However, the core contributions of that work remain valid and relevant within the scope of the proposed methodology.

The following sections detail additional experiments that, although not yet published, were conducted with methodological rigor and substantially reinforce the validation of the proposed approach.

Some components of the architecture, particularly those related to the optional extensions layer, remain as future work, either due to the need for more complex integrations (such as judicial ontologies) or because they require broader institutional evaluation (as in the case of application-level metrics across different courts or legal offices). Nonetheless, the core of the methodology has been fully exercised and validated through these experiments.

5.1 Fostering Judiciary applications with new fine-tuned models for Legal Named Entity Recognition in Portuguese

Artificial Intelligence applied to Law is getting attention in the community, both from the Judiciary and the lawyers, due to possible gains in procedural celerity and the automation of repetitive tasks, among other use cases. Many of its benefits can be derived from Natural Language Processing since legal proceedings are textual document-based. Named Entity Recognition is the NLP task of identifying and classifying named entities in unstructured text to help achieve these goals. In recent years, transformers emerged as the new state-of-the-art architec-

ture for some tasks in NLP systems. NLP resources in Portuguese, such as models and datasets, are needed to allow Portuguese speaking countries to benefit from this new NLP level.

This experiment describes the first fine-tuned BERT models trained exclusively on Brazilian Portuguese for Legal NER. The models achieved new state-of-the-art on LeNER-Br dataset, a Portuguese NER corpus for the legal domain. We also built a prototype application for Judiciary users to evaluate how the model performed with authentic law documents. The results showed that the models were able to extract information with good quality. Both the models and the prototype application are publicly available for the community.

A general view of the motivation is described below.

Named Entity Recognition (NER) is the problem of locating and categorizing important nouns and proper nouns in a text (MOHIT, 2014), being one of the leading Natural Language Processing (NLP) tasks. NER has great importance in real applications in the Law field due to the large amount of unstructured data usually written by lawyers in petitions or by judges in orders, decisions, and sentences (DOZIER et al., 2010). Advances in NLP based on traditional word embedding such as Word2Vec (MIKOLOV et al., 2013) and GLoVE (PENNINGTON; SOCHER; MANNING, 2014) and more recently contextualized word embeddings based on pre-trained neural language models built from deep learning such as ELMo (PETERS et al., 2018), ULMFit (HOWARD; RUDER, 2018) and transformer-based architectures (VASWANI et al., 2017; WOLF et al., 2020) such as BERT (DEVLIN et al., 2019) quickly leverage the State-Of-The-Art (SOTA) in NLP tasks (SANTOS; CARDOSO, 2006).

Although far away from the amount available in English and other languages, resources in Portuguese are rapidly becoming available, supporting the NLP research in countries like Brazil and Portugal. The Brazilian Portuguese brWaC Corpus (WAGNER FILHO et al., 2018) was recently made available with a resulting corpus including 145 million sentences and 2,7 billion tokens, based on 38 million URLs of .br top-level domain pages. First HAREM (SANTOS; CARDOSO, 2006) and Second HAREM (FREITAS et al., 2010) were respectively the first and second evaluation contest for named entity recognizers for Portuguese. Both corpora are manually labeled with ten named entities. Paramopama (MENDONçA JúNIOR et al., 2015) is a tagged Brazilian Portuguese corpus for named entity recognition, extending the PtBR version of WikiNER corpus (NOTHMAN et al., 2013), revising incorrect assigned tags to improve corpus quality and extend the corpus size. WikiNER is a multilingual silver-standard corpus for named entity recognition, automatically built by exploiting the text and structure of Wikipedia. Another silver-standard corpus is SESAME (Silver-Standard Named Entity Recognition dataset) (MENEZES; SAVARESE; MILIDIú, 2019), a massive dataset built on the Portuguese Wikipedia and DBpedia. LeNER-Br (ARAUJO et al., 2018) is the first Portuguese language dataset for named entity recognition applied to legal documents, and consists entirely of manually annotated legislation and legal cases texts and contains tags for persons, locations, time entities, organizations, legislation, and legal cases. Table 20 presents a comparison between NER datasets available in Portuguese.

Table 20: Comparison between available NER datasets for Portuguese

Corpus	Sentences	Tokens	Year
First HAREM	5.000	80.000	2006
Second HAREM	3.500	100.000	2008
WikiNER	125.821	2.830.000	2013
Paramopama	12.500	310.000	2015
LeNER-Br	10.392	318.073	2018
SESAME	3.650.909	87.769.158	2019

Source: Elaborated by the author.

NER is a token classification task, which usually has good results in BERT based models. According to the best of our knowledge, there was no Legal Named Entity Recognition in Portuguese using a BERT model pre-trained exclusively on Portuguese texts when we started this work. Considering our research in AI applied to Law and the resources currently available, we decided to train BERT models in Brazilian Portuguese for Legal NER and compare them with previous benchmarking to evaluate improvements in the use of BERT models in the development of applications for the Judiciary and for the legal domain in general.

In this experiment, we fine-tuned a BERT model pre-trained on Brazilian Portuguese for the NER task in the legal domain, achieving new state-of-the-art performance on the LeNER-Br corpus. We also built a prototype application so that Judiciary users could try out the model with real documents. We discuss the results and suggest future work to support the development of real and effective applications for the Judiciary.

5.1.1 Materials and methods

This section presents details on the construction of the models and on the prototype application. We emphasize the hyperparameters used and the tokenization process of the fine-tuning, as well the functionalities of the application.

5.1.1.1 Fine-tuned Legal NER

We trained two models by fine-tuning both the bert-base-portuguese-cased (BERTimbau Base[1]) and the bert-large-portuguese-cased (BERTimbau Large[2]) pre-trained models (SOUZA; NOGUEIRA; LOTUFO, 2020). We trained it on the LeNER-Br[3] dataset (ARAUJO et al., 2018), generating two legal fields domain-specific BERT models for the NER task. The BERT Base model has 12 layers, 768 hidden dimensions, 12 attention heads, and 110M parameters. The BERT Large has 24 layers, 1024 hidden dimensions, 16 attention heads, and 330M parameters. For both, the maximum sentence length is 512 tokens, and the vocabulary size is 29794

[1] https://huggingface.co/neuralmind/bert-base-portuguese-cased.
[2] https://huggingface.co/neuralmind/bert-large-portuguese-cased.
[3] https://huggingface.co/datasets/lener_br.

Figure 51: Example of the subword tokenization

```
['-', 'Tratando-se', 'de', 'ação', 'indenizatória', 'ajuizada', 'por',
'pessoa', 'incapaz', ',', 'é', 'obrigatória', 'a', 'intervenção', 'do',
'Ministério', 'Público', 'na', 'condição', 'de', 'fiscal', 'da', 'or-
dem', 'jurídica', '.']
['[CLS]', '-', 'Trata', '##ndo', '-', 'se', 'de', 'ação', 'inden',
'##iza', '##tória', 'aju', '##izada', 'por', 'pessoa', 'incapaz', ',',
'é', 'obrigatória', 'a', 'intervenção', 'do', 'Ministério', 'Público',
'na', 'condição', 'de', 'fiscal', 'da', 'ordem', 'jurídica', '.',
'[SEP]']
(In the case of an indemnity action filed by an incapable person, the
intervention of the Public Prosecutor's Office as supervisor of the
legal system is mandatory.)
```

Source: Elaborated by the author.

words.

In general, the main technical aspects of the models are the same as the BERTimbau (SOUZA; NOGUEIRA; LOTUFO, 2020), considering that the former is a fine-tuning of the latter. The following hyperparameters were used to build the models: learning_rate = 2e-05; weight_decay = 0.01; lr_scheduler_type = linear; optimizer = Adam with betas = (0.9,0.999) and epsilon = 1e-08; gradient_accumulation_steps = 1. The only difference between the fine-tuned base and large versions is the batch size (train_batch_size and eval_batch_size), respectively 8 and 4.

We used a single Tesla P100-PCIE 16GB GPU. The GPU limited the batch size of large version in 4. The base version accepted a batch size of 16, but better results were obtained with 8. To build the models, we used the Transformers library (WOLF et al., 2020). This is a powerful tool for developing transformer-based models and supports a great variety of architectures (almost 70) like BERT, DistilBERT, RoBERTa, BART, ELECTRA, GPT-2, T5, etc.

The tokenizer is an important component of the solution. We used a fast version of the BERT tokenizer with a subword tokenization algorithm. The fast version allows a significant speed-up in particular when doing batched tokenization. Subword tokenization relies on the principle that frequently used words should not be split into smaller subwords, but rare words should be decomposed into meaningful subwords. It allows the model to have a reasonable vocabulary size while being able to learn meaningful context-independent representations. It also enables the model to process words it has never seen before by decomposing them into known subwords. In our case, as the dataset is already splitted into words, the tokenizer splitted each of those words again. We had also used the tokenizer to add the special tokens [CLS] and [SEP] to identify the start and the separation of the sentences. Due to added special tokens and possible splits of words in multiple tokens, the tokenizer return is longer than the dataset sentence, so we had to do some postprocessing to align the labels after the tokenization.

Figure 51 shows a sentence in the dataset and the result of its tokenization. As we can see, some words considered rare by the tokenizer, such as “indemnity” (“indenizatória”, in Portuguese), were divided into subwords.

The fine-tuning performed 10 epochs over the training data. At the end of the training, the best model considering the F1-score metric was loaded to evaluate the results in the test set. The trained models bertimbau-base-lener_br[4] and bertimbau-large-lener_br[5] were made freely available for the community on HuggingFace’s model hub.

5.1.1.2 Prototype application

A Proof of Concept (POC) application was developed and made available[6], allowing an evaluation of users from the Judiciary considering a real context of use. After using the POC application, users from the Court of Justice of the State of Rio Grande do Sul answered a questionnaire, evaluating the application about the perceived usefulness.

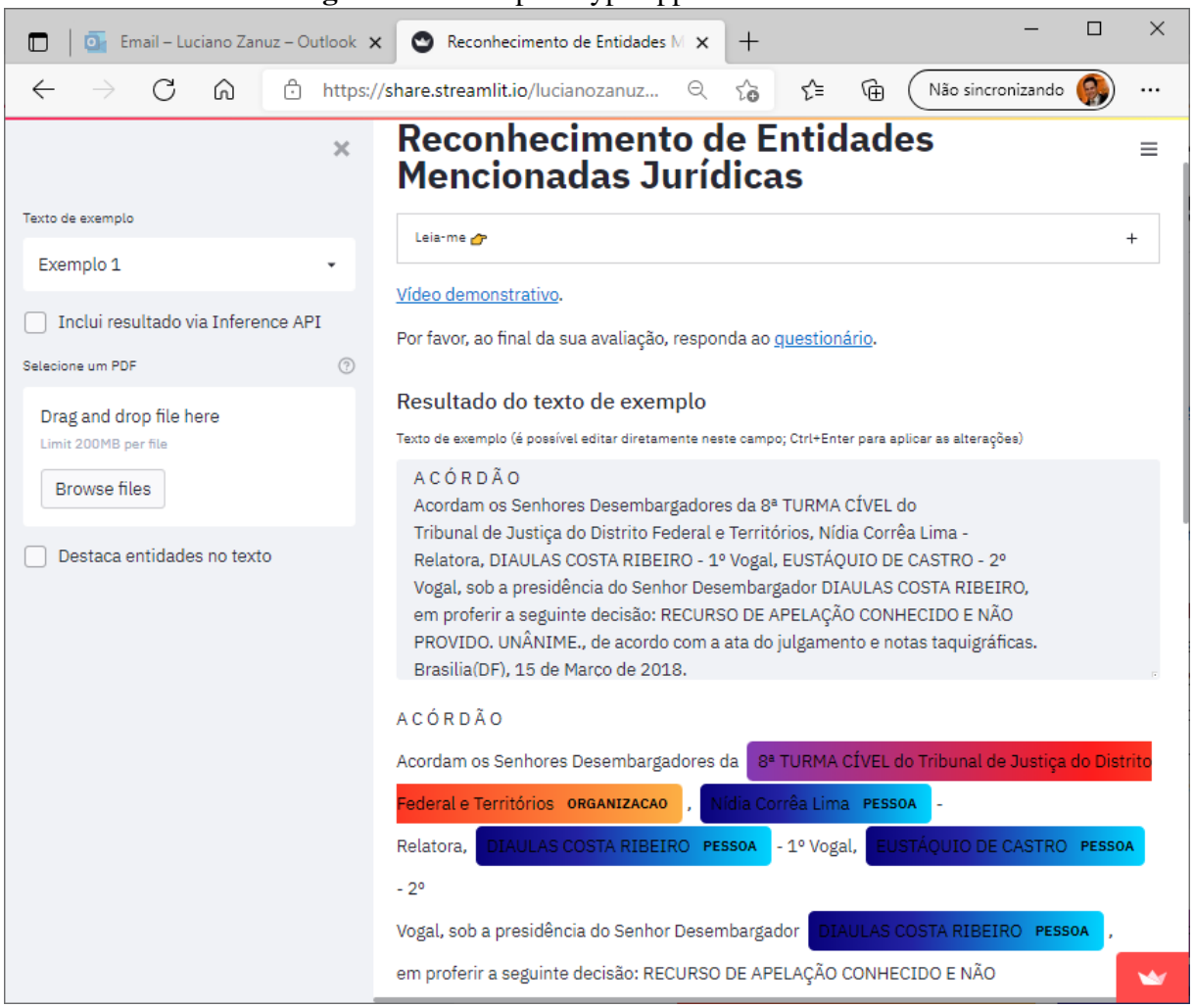
A NER system would be useful in several use cases in the Judiciary. For instance, it could be applied to all petitions received from lawyers to identify entities of interest, which could be used to facilitate human analysis or as input to automations. It could also be used to process text from hearings transcripts and identify people, places and other entities mentioned, which could be presented in a graphical user interface. Among other useful use cases.

The application enables users to test the extraction of entities from text and directly from PDF files. Figure 52 illustrates the main app screen. The app has two legal texts as examples presented in a text area, where users can make changes or type in a new text. The POC presents the possibility for users to upload their documents in PDF. The entities extraction is presented in visual markups on the text and in a table format, and it is possible to download the extractions in CSV format.

To allow users to work with real documents that are usually larger than the model’s limit of 512 tokens, the strategy adopted was to split the document into sentences. For this, we used the Spacy library along with the pt_core_news_sm Portuguese model. The aggregation strategy used to fuse tokens based on the model prediction into group entities follows the default schema ((A, B-TAG), (B, I-TAG), (C, I-TAG), (D, B-TAG2) (E, B-TAG2) will end up being [“word”: ABC, “entity”: “TAG”, “word”: “D”, “entity”: “TAG2”, “word”: “E”, “entity”: “TAG2”]). The scores are averaged first across tokens, and then the maximum label is applied, so words don’t end up with different tags. This is necessary since subword tokenization was used.

[4]https://huggingface.co/Luciano/bertimbau-base-lener_br.
[5]https://huggingface.co/Luciano/bertimbau-large-lener_br.
[6]https://share.streamlit.io/lucianozanuz/streamlit-lener_br/main.

Figure 52: Main prototype application interface



Source: Elaborated by the author.

Table 21: Comparison between our models and previous benchmarking

Entity Type	LSTM+ CRF	BERT- Multi	BERT- Acordaos	ELMo- Acordaos	BERTimbau Base	BERTimbau Large
Person	80.83±1.83	91.38±0.87	93.66±0.21	98.28±0.41	96.91±0.27	97.38±0.44
Time	90.12±2.28	94.43±0.39	92.39±1.82	93.77±1.05	96.01±0.10	96.04±0.58
Location	64.81±3.05	68.67±1.89	65.78±2.26	75.21±1.13	75.39±4.71	75.67±3.18
Organization	84.37±0.62	84.76±0.57	86.43±0.49	86.54±0.74	86.21±0.66	86.66±1.17
Law	93.01±0.61	93.40±1.23	95.48±0.47	88.08±1.18	94.36±1.10	95.90±0.83
Legal Case	81.22±1.65	84.04±0.63	83.26±0.73	80.81±1.14	87.32±0.99	87.76±0.87
Overall	**85.42±0.61**	**88.81±0.29**	**89.39±0.08**	**88.41±0.41**	**90.62±0.20**	**91.14±0.39**

Source: Elaborated by the author.

5.1.2 Results and discussion

This section presents the results of the trained models and the evaluation of the prototype application.

5.1.2.1 Fine-tuned Legal NER

The results reported in this section are F1-score averages over five runs and the respective standard deviations obtained on the test set of the dataset. The metrics were computed using the seqeval library[7]. Seqeval is an entity-level evaluation metric, which means that a predicted entity is only considered correct if it matches the exactly chunk and tag. This is different from LeNER-Br (ARAUJO et al., 2018) and BertBR (CIURLINO, 2021) papers, where the metrics are based on the token-level evaluation.

Table 21 presents the comparison between the results obtained by our models and previous benchmarking (BONIFACIO et al., 2020) on the LeNER-Br dataset. LSTM+CRF is the original model of LeNER-Br paper (ARAUJO et al., 2018), BERT-Multi and BERT-Acordaos are based on pre-trained BERT Multilingual, which is trained on Wikipedia, and the ELMO model is trained on the brWaC Corpus (WAGNER FILHO et al., 2018). Both BERT-Acordaos and ELMo-Acordaos models have an additional step of fine-tuning on the intradomain Acordaos TCU corpus. Our models BERTimbau-Base and BERTimbau-Large are based on pre-trained BERTimbau (SOUZA; NOGUEIRA; LOTUFO, 2020), which is trained on the brWaC Corpus, with no additional intradomain fine-tuning.

As shown in the table, our models defined a new state-of-the-art for Portuguese Legal NER, achieving an overall F1-score of 91.14 on BERTimbau-Large and 90.62 on BERTimbau-Base. Therefore, it increases the previous benchmarking of BERT base version in 1.81 points, or 1.23 points if compared to the intradomain fine-tuned BERT-Acordaos. BERTimbau-Base obtained better results in all entities comparing to BERT-Multi, and only worst in Organization and Law

[7]https://github.com/chakki-works/seqeval.

Table 22: Basic statistics for the unlabeled Portuguese corpora

Corpus	Documents	Sentences	Tokens
brWaC	3,530k	145,000k	2,680mi
Wikipedia	934k	13,900k	1,400mi
Acordaos-TCU	298k	9,000k	912mi

Source: Elaborated by the author.

if compared to BERT-Acordaos. Our BERTimbau-Large outperformed the BERT models in all entities, being only smaller than ELMo-Acordaos in entity Person.

The results show the importance of unlabeled data on the model quality. As illustrated in Table 22, brWac is considerably larger than Wikipedia and Acordaos-TCU. Our models, which are based exclusively on brWac corpus, obtained better results even not being fine-tuned in the legal intradomain Acordaos-TCU corpus. On the other hand, agreeing with (BONIFACIO et al., 2020), we believe that we could improve the results by adding a previous step of fine-tuning a general language model with an intradomain context-specific corpus, as it occurred with them.

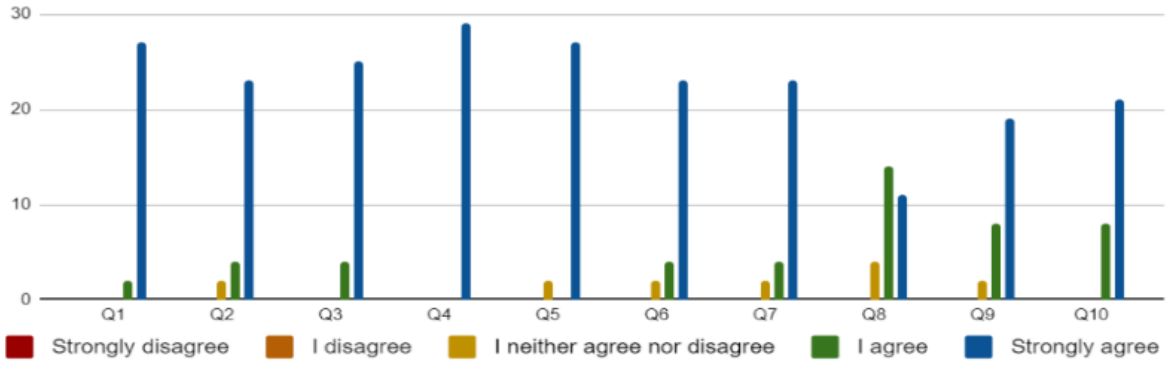
5.1.2.2 Prototype application

The application has been made available for about ninety users from the Brazilian Judiciary, including judges and office servers. The users evaluated the application and answered a questionnaire based on the Technology Acceptance Model (TAM) (MARANGUNIć; GRANIć, 2015) by Fred Davis, which is a dominant model in investigating factors affecting users' acceptance of the technology. Although TAM assesses perceived ease of use and usefulness, in this case we were interested only in the latter, regarding the goals of the POC.

The questionnaire had ten statements with a positive bias, in which users answered their level of agreement. The answers were standardized on a five-point Likert scale ("Strongly agree", "I agree", "I neither agree nor disagree", "I disagree" and "Strongly disagree"). The statements were: Q1. The use of AI applied to Law is useful for the Judiciary; Q2. AI can help automate and speed up legal proceedings; Q3. AI can be an important decision support tool for magistrates and civil servants; Q4. AI can be an important tool for extracting information from legal texts; Q5. NER in legal texts is useful for legal activity; Q6. The entities recognized by the trained model are relevant in the legal context; Q7. It would be useful to train models capable of extracting other entities, including those based on specific contexts, such as health; Q8. The resulting quality (accuracy) of the extractions was acceptable; Q9. It would be useful to assess whether models trained in specific contexts are more accurate in extracting entities from those contexts (for instance, if models trained with petition texts are more accurate in extracting entities from petition texts); Q10. It would be useful to automatically apply NER to the textual documents of legal proceedings, such as petitions from lawyers and transcripts of hearings, for extraction and structured storage of entities.

As shown in Figure 53, the results were very encouraging. Twenty-eight users answered

Figure 53: Summary of questionnaire responses



Source: Elaborated by the author.

the questionnaire with no negative response (Strongly disagree or disagree). In the first four questions, we took the opportunity to verify the user's opinion about AI and Law in general. The results indicate that this group of people from Judiciary believe in AI as a useful and important tool for decision support and information extraction to increase automation and celerity of legal proceedings. The last six questions are specific about NER and the prototype application. The results show that Judiciary users perceived the usefulness of the NER application in the real-world.

The less positive rating was on Q8, about the accuracy of the extractions (Strongly agree: 10; I agree: 14; I neither agree nor disagree: 4), in which we received some interesting feedback from the users: "Nothing escapes!"; "The identification of Law and Location presented some confusion. The identification of people seemed correct in most cases."; "It was quite acceptable, only getting a little confused in Legal Case that sometimes separates the case numbers into several parts."; "Overall it works fine. But I could see that there is room for improvement. Long dates it identifies well but lacks in other date formats. I also noticed that it can't capture complete addresses as a single token.". User feedback is in line with our tests. Although the model obtained very good results in the dataset, the real-world documents are more diverse. Some results clearly reflect a probable underfitting caused by few samples used in training.

The models used in this experiment obtained new state-of-the-art on LeNER-Br dataset, achieving an overall F1-score of 91.14 in the large version and 90.62 in the base version, considering the seqeval entity-based evaluation metric. We also built a prototype application for Judiciary users to evaluate how the model performed with real documents. User evaluation results showed that the models were able to extract information with good quality. However, there could be gains with the training of models on a new corpus containing a greater amount and variety of legal texts, more adherent to the context where they would be applied, for instance, using text documents by lawyers. Both the models and the application were made publicly available to the community.

5.2 Automated Judgment Report Generation in Brazilian Legal Proceedings Using Generative AI

The use of Artificial Intelligence (AI) in the judiciary is transforming legal workflows by automating tasks such as document analysis and text generation. This study investigates the application of Generative AI to support the drafting of judgment reports. We developed a system that integrates a prompt-design interface and a Judgment Report Generator capable of producing draft reports from case records. To evaluate its performance, we compiled a validation dataset of 105 human-authored reports and compared them to AI-generated outputs using traditional NLP metrics, LLM-as-a-Judge evaluation, and human assessments by legal experts. While these evaluation methods have been used separately in prior studies, their combined application provides a more comprehensive assessment. Results show that, with carefully designed prompts and clearly defined tasks, Large Language Models can effectively support legal drafting, improving efficiency while maintaining legal standards. The experiments also demonstrated that smaller models can outperform larger ones in specific, well-structured scenarios.

5.2.1 Introduction

Since the formal introduction of "Artificial Intelligence" (AI) at the Dartmouth Conference in 1956, AI has profoundly transformed numerous sectors of human society. In the legal domain, AI applications date back over six decades, starting with Lucien Mehl's 1958 article Automation in the Legal World. By 1963, researchers like Lawlor were already exploring the potential of computers to analyze and predict judicial decisions (LAWLOR, 1963). The AI and Law research community gained further momentum with the launch of the International Conference on Artificial Intelligence and Law (ICAIL) in 1987 (BENCH-CAPON et al., 2012). While early efforts in AI and Law focused on rule-based systems and knowledge representation, the field has increasingly embraced machine learning-driven approaches since the 2000s.

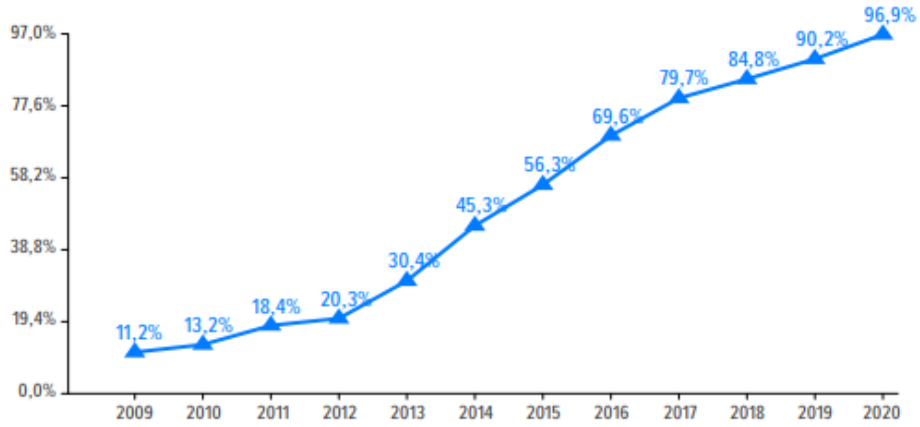
A critical enabler of these advances is Natural Language Processing (NLP), a research field dedicated to enabling computers to understand and manipulate natural language text or speech for practical applications. NLP powers tasks such as semantic similarity analysis, information extraction, text summarization, automatic question answering, and document classification, capabilities that are particularly relevant to the legal field. Over time, NLP has evolved from statistical and probabilistic approaches to neural network-based models (OTTER; MEDINA; KALITA, 2021), culminating in the development of Pre-trained Language Models (PLMs) powered by transformer architectures (VASWANI et al., 2017), built over attention layers (BAHDANAU; CHO; BENGIO, 2016; GALASSI; LIPPI; TORRONI, 2021). Examples such as GPT (RADFORD et al., 2018), BERT (DEVLIN et al., 2019), and T5 (RAFFEL et al., 2019) have set new benchmarks for NLP tasks.

More recently, Large Language Models (LLMs), PLMs with billions of parameters, have

revolutionized NLP with their ability to perform complex generative tasks such as text generation and summarization (ZHAO et al., 2023). These models are transforming how AI systems interact with text, offering new opportunities for automation and scalability, especially in fields handling large volumes of unstructured textual data, such as Law. The term Generative AI stems from the architectural design of these LLMs, which are predominantly decoder-only models derived from the Transformer architecture (VASWANI et al., 2017). This design choice makes them particularly well-suited for generative tasks, as the decoder's structure excels in producing coherent and contextually relevant text outputs.

In Brazil, the legal system is witnessing a rapid shift towards digitization, with electronic lawsuits becoming the norm (CNJ, 2021), as illustrated by Figure 54. In 2023 alone, 35.1 million new cases were filed electronically, adding to over 253.3 million cases digitized in the past 15 years (CNJ, 2024a). This unprecedented scale demands innovative solutions to process, classify, and summarize legal documents efficiently, and NLP-driven automation represents a compelling opportunity. One particularly labor-intensive task is drafting the judgment report, a section of judicial decisions summarizing the case's key facts in chronological order. According to the Code of Civil Procedure (*CPC - Código do Processo Civil*, in Portuguese) (BRASIL, 2015), this report must include details such as party names, action type, initial requests, counterclaims, and significant procedural events. As of September 30, 2024, the Court of Justice of Rio Grande do Sul alone faced a backlog of 176,921 pending judgments (CNJ, 2024b), underscoring the need for automation.

Figure 54: Evolution in the percentage of electronic lawsuits in Brazil



Source: (CNJ, 2021)

This work presents a practical solution for the automated generation of judgment reports using Generative AI, implemented as a fully integrated component within an electronic lawsuit platform. While similar technologies have been explored in adjacent domains, we are not aware of prior studies specifically focused on the generation of judicial judgment reports in the context of the Brazilian legal system. Our methodology leverages state-of-the-art AI techniques and is designed for real-world application, enabling users to generate reports directly from court

documents through a structured and guided interface.

The system evaluation was conducted using 105 real-world legal proceedings and employed a multi-dimensional approach, combining traditional NLP metrics, the LLM-as-a-Judge validation technique, and human expert assessments. In addition, a prompt engineering playground was developed and embedded in the system, allowing users to prototype and test different prompt strategies with real data. This interactive component supports continued refinement and experimentation, encouraging adoption and adaptation in future legal AI projects.

The primary contribution of this work lies in the integration of technical and practical elements into a deployable legal solution. We propose a structured methodology that combines domain-specific prompt engineering with the capabilities of large language models to extract key legal elements, such as parties, claims, defenses, and procedural events, and guide report generation. This enables the production of judgment reports that are coherent, accurate, and legally aligned.

Furthermore, we present a flexible evaluation framework that incorporates traditional and semantic NLP metrics, model-based assessment via LLMs, and expert human review. This layered evaluation strategy contributes to understanding how different methods align in assessing legal text generation quality. By addressing the specific challenges of the Brazilian legal system, this work also contributes to the relatively underdeveloped ecosystem of AI resources for legal applications in Brazil, paving the way for broader adoption of AI in legal contexts.

The remainder of this paper is organized as follows. Section 5.2.2 reviews related studies. Section 5.2.3 outlines the proposed methodology. Section 5.2.5 details the experiments and discusses the results. Finally, section 5.2.7 highlights our conclusions and suggests future research directions.

5.2.2 Related Work

This section reviews related works on the application of artificial intelligence to the summarization and generation of legal texts, with a special focus on large language models (LLMs) and the Brazilian legal context. The goal is to situate the present study within the broader landscape of AI for legal decision support while clarifying how the task of generating judgment reports relates to and differs from existing work on legal summarization.

The concept of a judgment report (*relatório de sentença*, in Portuguese) in Brazilian law is legally defined in Article 489 of the Brazilian Code of Civil Procedure (*CPC - Código do Processo Civil*, in Portuguese). It refers to a formal section of a judicial decision that must contain, at a minimum, the names of the parties, the identification of the case, a summary of the claims and defenses, and a description of the main procedural events. This distinguishes the task from general legal summarization, which often aims to compress or abstract the main idea of a legal document (e.g., case law, contracts, statutes) for a broader audience or downstream processing.

In contrast, the generation of judgment reports is not only content-specific, but also structure-constrained and legally regulated. It requires selecting and organizing information in accordance with formal rules and judicial writing practices. Unlike general summarization or brief generation (e.g. headnotes or verdict summaries), judgment reports serve as a required part of the judicial ruling and must follow an official format, which has implications for both their content and legal function.

To the best of our knowledge, there are no prior works specifically targeting the automation of legally compliant judgment reports in Portuguese, nor systems that integrate such functionality into real-world electronic judicial platforms. This work differs from earlier summarization efforts by addressing a highly structured, domain-specific, and legally constrained form of text generation. Furthermore, our methodology combines prompt engineering, functional system development, and a multi-layered evaluation strategy (including human experts and LLM-as-a-Judge), which goes beyond purely experimental or benchmark-driven studies.

Papers included in this section were selected using a non-systematic, snowballing approach (WOHLIN, 2014), prioritizing recent works that incorporate LLMs into legal summarization or generation pipelines. The comparison of objectives, techniques, evaluation metrics, and results helps contextualize the scope and focus of the present study within ongoing research in this area.

5.2.2.1 Legal text summarization

(FEIJO; MOREIRA, 2019) investigate the suitability of extractive and abstractive approaches in summarizing legal rulings. The authors experimented with ten extractive and four abstractive models in a real dataset containing 10K rulings from the Brazilian Supreme Court (VARGAS FEIJó; MOREIRA, 2018). These models were compared with an extractive baseline based on heuristics to select the most relevant parts of the text. The results show that abstractive approaches significantly outperform extractive methods regarding ROUGE scores.

Summarizing legal documents has a wide range of applications for NLP, especially processing information from a large body of unstructured text to its efficient retrieval appropriate to a search query. While extractive summarization extracts essential sentences or phrases from the source text to obtain the summary, considering the words generated in the summary will be a subset of the original text, abstractive summarization always tries to obtain a brief, readable summary. It introduces novel words to obtain the summary (SHEIK; NIRMALA, 2021). The study conducted in (FEIJO; MOREIRA, 2019) concludes that abstractive summarization significantly outperforms extractive methods in terms of ROUGE score, and, although abstractive summarization is more complicated as it demands a large amount of resources and complex algorithms to fuse the sentence and create it in a compressed form, with the advancement of the use of deep learning techniques supported by LLM, this complexity has been overcome.

(LINS et al., 2024) proposes CLSJUR.BR, a contrastive learning model for automatic and abstractive summarization of legal documents in Portuguese. Inspired by SimCLS (LIU; LIU,

2021), the CLSJUR.BR architecture is divided into three stages: pre-processing, candidate generation, evaluation of summaries, and election of the final summary. CLSJUR.BR was trained and evaluated based on the Ruling.BR dataset (VARGAS FEIJó; MOREIRA, 2018), which is a corpus in Portuguese composed of 10,623 judicial sentences from the Brazilian Federal Supreme Court, dated between 2012 and 2018. The Ruling.BR's judicial sentences are structured into "Summary", "Report", "Vote" and "Judgment", in which the topic "Summary" summarizes and discloses the content of judicial decisions, being used as the reference summary in the evaluation of the models. The evaluation metrics are based on ROUGE scores (LIN, 2004). The results showed that, within the scope of legal summarization for the Brazilian Legal System, the model's characteristic of generating several candidate summaries for each document, through the sampling generation strategy made it possible to obtain better summaries than just generating a single summary.

The authors in (SHUKLA et al., 2022) analyze various extractive and abstractive summarization methods applied to legal case documents. The performance of models such as BART, Legal-Pegasus, and Longformer are compared for abstractive methods. The authors utilize three datasets, two from the Indian Supreme Court and one from the UK Supreme Court, each consisting of legal case judgments with summaries. Evaluation is performed using ROUGE-1, ROUGE-2, and ROUGE-L F-scores, along with BERTScore, which uses a pre-trained BERT model to evaluate semantic similarity. Two types of evaluations are performed: document-wide, comparing entire summaries to reference summaries, and segment-wise, focusing on specific segments such as Facts or Background in legal judgments. Additionally, expert evaluation is conducted by law professionals on the highest-performing summaries. The authors demonstrate that Legal-Pegasus produces the best summaries among the pre-trained models, mainly due to its pre-training on legal documents, followed by BART-based methods. The expert evaluation highlighted that the abstractive summaries could have been more organized and contained complete sentences compared to the extractive models. While the experts saw potential in these summaries, they emphasized the need for improvement.

While early LLMs have not demonstrated impressive capabilities in abstractive summary generation, the latest advancements in the field considerably improved the performance in this task, as suggested by the recent work of (DEROY; GHOSH; GHOSH, 2023, 2024). The authors applied domain-specific abstractive summarization models and the latest general-domain LLMs to the same Indian and UK Supreme Court datasets and to the dataset of government legal reports from the United States (US). We find Text Davinci-003, Turbo-GPT-3.5, GPT-4 Turbo, and Llama2-70b between the LLMs explored in this work.

The metrics used to measure the performance of the summarization models in matching gold-standard summaries are ROUGE, METEOR, and BERTScore. Furthermore, the authors also evaluate the consistency of the generated summaries based on the original document. The selected consistency metrics assess how well the summary aligns with the original document and produce scores in the range [0, 1], with higher scores indicating greater consistency.

1. **SummaC** (LABAN et al., 2022): This metric uses Natural Language Inference (NLI) to compare each sentence in the summary with the original document. By determining how likely a sentence in the summary logically follows from the document's content, SummaC scores the summary's overall consistency. Lower NLI scores for a sentence suggest a mismatch or hallucination of information.

2. **NumPrec** (DEROY; GHOSH; GHOSH, 2023): This metric evaluates the accuracy of numbers in the summary, such as dates and monetary values, by comparing the presence of numbers in both the summary and the original document. The metric returns the fraction of numbers that appear consistently in both.

3. **NEPrec** (DEROY; GHOSH; GHOSH, 2023): Named Entities Precision (NEPrec) measures how well named entities, like person or organization names, are preserved in the summary. It returns the fraction of named entities present in the original document that are also found in the summary, relying on the Spacy toolkit for entity detection.

The findings highlight that abstractive models and large language models (LLMs) like Text-Davinci-003, Turbo-GPT-3.5, Llama-70b, and GPT-4 generally outperform extractive models in quantitative metrics, even without domain-specific training. However, generative models often exhibit hallucinations and inconsistencies in their summaries. Fine-tuning these models with domain-specific data can help reduce these issues and improve summary quality, as observed in datasets like UK-Abs and IN-Abs. Additionally, well-crafted prompts can minimize hallucinations, though this may reduce overall summary quality. The authors explored a semantic similarity-based approach and showed promise in reducing hallucinations, but complex errors, such as confusion between names or numbers, remain challenging to prevent fully. In the legal domain, the extreme length of case judgments presents a unique challenge, where chunking the text improves information quality but may lead to redundancy and lack of coherence.

The authors claim that LLMs pre-trained abstractive summarization models still need to be prepared to be deployed in complex fields such as law. They point to a lack of techniques to detect complex errors in abstractive summaries. Currently, a human-in-the-loop approach is better suited for manually identifying errors in the generated summaries.

5.2.2.2 Legal text generation

(LIN; CHENG, 2024) presents a process for fine-tuning an LLM, enabling users to perform fine-tuning on a large pre-trained language model locally and apply it to legal document drafting. The proposed workflow begins with collecting legal documents and a simple preprocessing procedure to organize the data into an unlabeled dataset. Subsequently, the dataset is fed into a large pre-trained language model on a local machine for fine-tuning. This process aims to achieve a language model tailored explicitly for generating legal document drafts, thereby enhancing the applicability of the large pre-trained language model in the Chinese legal

domain. The basic model of the experiment in this paper is the pre-trained large-scale language model BLOOM (WORKSHOP et al., 2023) using the Casual Language Model, an open-source multilingual LLM, including traditional Chinese text. The authors' objective was to try to let BLOOM generate a draft of criminal facts about offenses of fraudulence, but the result was quite unsatisfactory. The authors found that as long as a sufficient amount of legal professional texts are collected to fine-tune an LLM, the performance of the text generated by the model can be further improved. These results show that large language models have extensive application prospects in generating legal document drafts.

(SAKIYAMA; NOGUEIRA; ROMERO, 2023) address the problem of automating the writing of keyphrases, a sequence of key terms present in documents used in courts throughout Brazil. The authors proposed using a text-to-text framework based on generative Transformers for this task. They evaluated several generative Transformer models (such as PTT5, mT5, OPT, and BLOOM) and compared their performance. The best-performing model, PTT5 (CARMO et al., 2020), was adopted as the text generator since it achieved a BLEU score of 37.54% in the test set, outperforming other evaluated methods by up to 24.6%. The authors also assessed the influence of keyphrases and the quality of the generated ones, performing several experiments based on a real case of legal information retrieval. By using traditional information retrieval methods, such as TF-IDF and BM25, in combination with the original, generated keyphrases, or both, they observed gains statistically significant (p-value < 0.05) in all experiments.

Although LLMs have demonstrated impressive capabilities, these models still encounter limitations on extraction tasks. Retrieval-Augmented Generation (RAG) (GAO et al., 2024) is a novel approach that combines classic retrieval techniques and LLMs to address some of these limitations. (AQUINO et al., 2024) proposed a workflow that leverages LLMs within Retrieval Augmented Generation (RAG) pipelines to extract structured information from Brazilian legal documents related to fraud in public procurement processes. The authors evaluated the 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. The results also show that each extracted variable potentially holds an optimal combination of parameters, highlighting the context-dependency of each extraction.

5.2.2.3 Evaluation

The NLP task of summarizing legal texts has been a focus of research worldwide for some time, addressing the need to reduce the volume of text that humans must review, given the large number of texts involved in legal proceedings. In contrast, the relatively unexplored area of legal text generation is likely to see significant growth with the recent advent of large language models. Finally, to the best of our knowledge, there are no studies in the literature on the automated generation of judgment reports using artificial intelligence.

5.2.3 Materials and Methods

This section outlines the methodology used to develop the Judgment Report Generator, a real project developed for the Court of Justice of Rio Grande do Sul in southern Brazil. We begin by introducing the foundational components, LLMs and prompt engineering, with the objective of clarifying the underlying technology that powers the report generator. Next, we describe the LLM Chat Playground and the Judgment Report Generator features, aiming to provide a comprehensive understanding of the tools and functionalities available to users. Finally, we discuss the evaluation process, detailing our approaches for assessing and validating the quality of the generated results.

5.2.3.1 Large Language Models and prompt engineering

Typically, Large Language Models (LLMs) refer to Transformer language models that contain hundreds of billions (or more) of parameters[8], which are trained on massive text data (SHANAHAN, 2024), such as GPT-4 (OPENAI et al., 2024), PaLM (CHOWDHERY et al., 2024), Galactica (TAYLOR et al., 2022), and LLaMA (TOUVRON et al., 2023). LLMs exhibit strong capacities to understand natural language and solve complex tasks via text generation (ZHAO et al., 2023).

Recently, the research on LLMs has been primarily advanced by both academia and industry, and a remarkable progress is the launch of ChatGPT[9] (a powerful AI chatbot developed based on LLMs), which has attracted widespread attention from society. The technical evolution of LLMs has been significantly impacting the entire AI community, which would revolutionize how we develop and use AI algorithms (ZHAO et al., 2023).

A human interacts with LLMs through prompts, which are natural language instructions sent to the LLM, from which the LLM sends its response. The LLM can respond more or less satisfactorily depending on the prompt that interacted with it. The technique of optimizing the language in a prompt to interact with an LLM to elicit the best possible performance is called prompt engineering. Therefore, following the release of LLMs, prompt engineering emerges as a new field of interest in Artificial Intelligence research. We highlight some well-known prompt engineering techniques that have proven their effectiveness.

Since the emergence of LLMs, prompt engineering has become a very active and promising research area due to its importance in achieving effective results with these large language models. One of the most well-known prompt engineering strategies is Chain-of-Thought (CoT) (WEI et al., 2022) prompting, which enables LLMs to tackle complex arithmetic, commonsense, and symbolic reasoning tasks by providing a simple series of intermediate reasoning steps. This strategy is also known as Manual-CoT.

[8] In existing literature, there is no formal consensus on the minimum parameter scale for LLMs since the model capacity is also related to data size and total compute. (ZHAO et al., 2023)

[9] https://openai.com/index/chatgpt/

Zero-shot-CoT (KOJIMA et al., 2023) is a zero-shot template-based prompting for a chain of thought reasoning. It differs from the original chain of thought prompting (WEI et al., 2022) as it does not require step-by-step few-shot examples, and it differs from most of the prior template prompting (LIU et al., 2023a) as it is inherently task-agnostic. It elicits multi-hop reasoning across various tasks with a single template. The core idea of the method is simple: just adding "Let us think step by step" to extract step-by-step reasoning.

In practice, Manual-CoT (WEI et al., 2022) has obtained stronger performance than Zero-Shot-CoT (KOJIMA et al., 2023). However, this superior performance hinges on the hand-drafting of effective demonstrations. Specifically, hand-drafting involves nontrivial efforts in designing questions and reasoning chains for demonstrations (ZHANG et al., 2022). To eliminate such manual designs, (ZHANG et al., 2022) proposes an Auto-CoT paradigm to construct demonstrations with questions and reasoning chains automatically. Specifically, Auto-CoT leverages LLMs with the “Let us think step by step” prompt to generate reasoning chains for demonstrations one by one, i.e., *let us think not just step by step, but also one by one*.

Another automatic prompt engineering approach was proposed by (ZHOU et al., 2023). Inspired by classical program synthesis and the human approach to prompt engineering, the authors propose Automatic Prompt engineering (APE) for automatic instruction generation and selection. The method treats the instruction as the program, optimized by searching over a pool of instruction candidates proposed by an LLM in order to maximize a chosen score function.

5.2.3.2 LLM Chat Playground

To support experimentation and improve prompt engineering, we developed the LLM Chat Playground, a feature integrated into the electronic judicial process system.

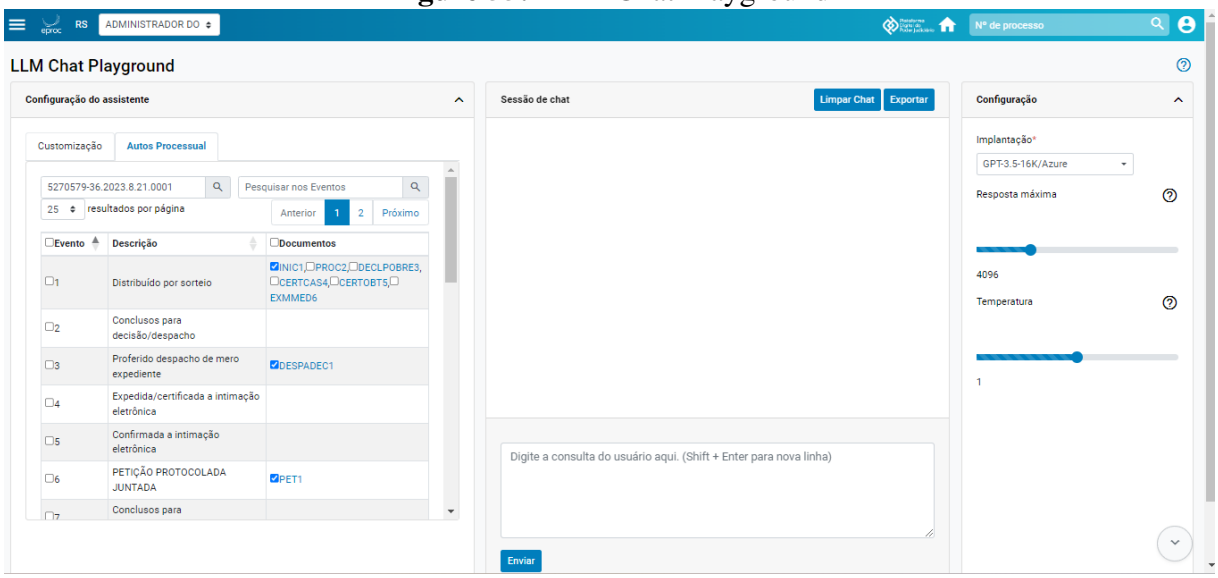
As shown in Figure 55, the tool allows users to search for specific court cases and select documents for testing prompts. On the right panel, users can configure the LLM (currently GPT-4o/Azure and GPT-4o-Mini/Azure), as well as key parameters such as max tokens and temperature. On the left panel, they can customize system messages and few-shot examples. In the central area, users can write and send prompts, view responses, and export results. This interactive setup enables legal professionals to refine prompt strategies by testing variations across real case documents.

This experimentation led to the identification of an initial prompt version, referred to as "version 1" (see A.1), which is now used in the Judgment Report Generator.

5.2.3.3 Judgment Report Generator

The Judgment Report Generator is accessible for court cases in the Concluded for judgment status. Once activated, the interface presents relevant procedural documents, preselecting those with predefined prompts, while allowing manual adjustments.

Figure 55: LLM Chat Playground



Source: Elaborated by the author

As shown in Figure 56[10], users can review and select documents (e.g., initial petition, judicial orders, contestations). Each document is processed by the LLM using a prompt tailored to its type; if no specific prompt is configured, a default summarization prompt is applied.

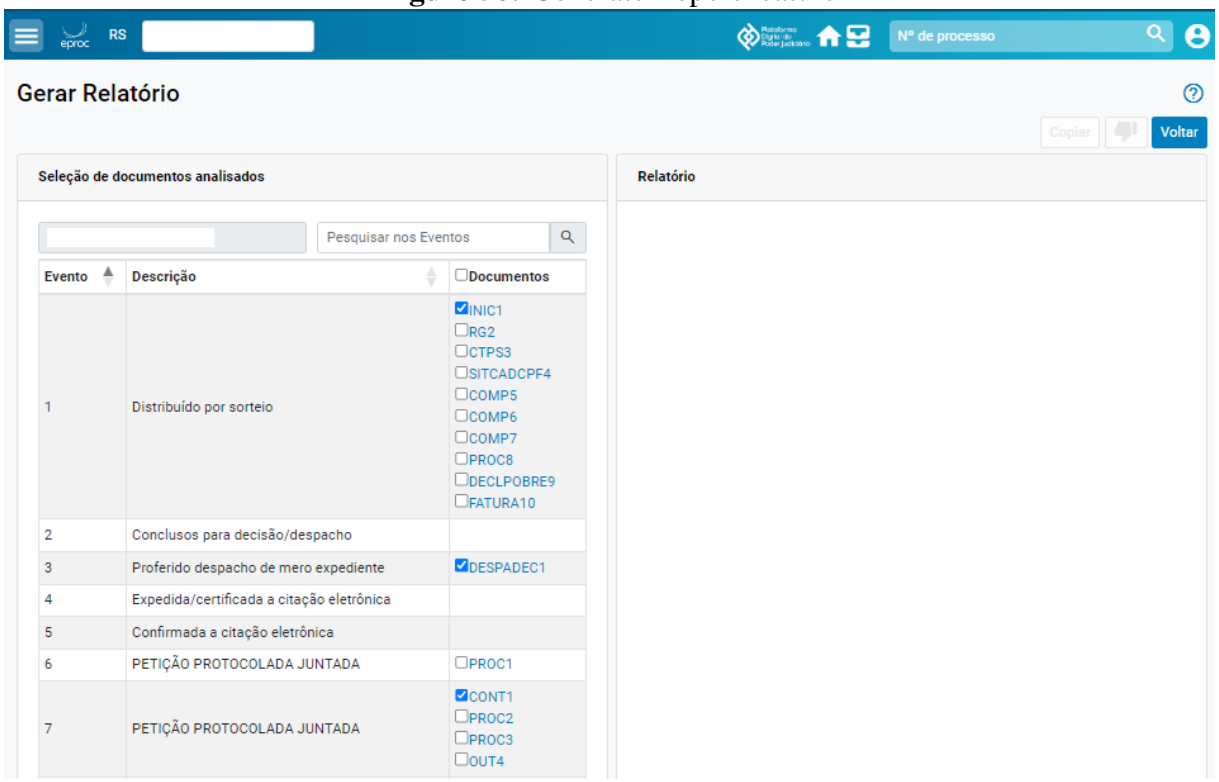
Listing 5.1: Judgment Report Generation Workflow

```
Algorithm: JudgmentReportGenerator
Input: CaseID (court case in "Concluded for judgment" status)
Output: Draft judgment report (text)

Begin

  Load all procedural documents associated with CaseID

  For each document D in CaseID:
    If D has a predefined prompt:
      Select the specific prompt template for D's type
    Else:
      Apply default summarization prompt
    Send D and selected prompt to LLM
    Receive summary SD for D
    Store SD in list Summaries

  Concatenate all summaries in Summaries -> SummaryBlock
```

[10] All the screenshots are available in Github: https://github.com/lucianozanuz/Manuscript-Number-JJIMEI-D-25-00093—Supplementary-repository

```
    Select final synthesis prompt (for complete report generation)

    Send SummaryBlock and final synthesis prompt to LLM

    Receive complete judgment report draft -> DraftReport

    Return DraftReport

End
```

Figure 56: Generate Report feature



Source: Elaborated by the author

In the second step, the system concatenates all intermediate summaries and submits them to the LLM along with a final synthesis prompt to generate the complete draft of the judgment report. The pseudocode describing the logic of the judgment report generator is presented in Listing 5.1, while all prompt templates used throughout the process are listed in Appendix A.1.

5.2.4 Evaluation Methodology

To assess the quality of AI-generated judgment reports, we adopted a three-level evaluation strategy: (i) automated NLP metrics, (ii) LLM-as-a-Judge, and (iii) human assessment.

Each method compares the report generated by the system (target text) with the original report authored by a judge (reference text).

The evaluation was designed to:

- Measure the quality of AI-generated reports;
- Identify the most effective combination of model and prompt;
- Select optimal prompts for LLM-as-a-Judge evaluation;
- Analyze correlations between automated metrics and human judgment;
- Determine which automated metrics best align with human assessments for future use.

5.2.4.1 Automated NLP metrics

Automated metrics offer rapid evaluation and are grouped into two categories: string-based and embedding-based. The former (e.g., ROUGE (LIN, 2004), BLEU (PAPINENI et al., 2002), METEOR (BANERJEE; LAVIE, 2005)) compare textual overlap, while the latter (e.g., BERTScore (ZHANG et al., 2020), BLEURT (SELLAM; DAS; PARIKH, 2020), and vector distance metrics) assess semantic similarity using embeddings from pre-trained models.

These metrics were selected due to their widespread use in summarization and translation tasks and their relevance for comparing texts that aim to convey the same legal content.

5.2.4.2 LLM-as-a-Judge

While traditional metrics can quantify surface similarity, they struggle to capture deeper semantic and legal adequacy. Thus, we incorporated the LLM-as-a-Judge paradigm (LI et al., 2024; ZHENG et al., 2023; OPENAI, 2024), where an LLM evaluates the output of another LLM based on a structured prompt and predefined criteria. This method, shown to align closely with human judgment (HUANG et al., 2024), is scalable, fast, and interpretable.

Our evaluation prompts were based on the Multidimensional Quality Metrics (MQM) framework, allowing the model to score dimensions such as accuracy, legal verity, and fluency. Issue severity was translated into penalties: 5 (major), 3 (minor), and 0 (none), as detailed in Appendix A.2.

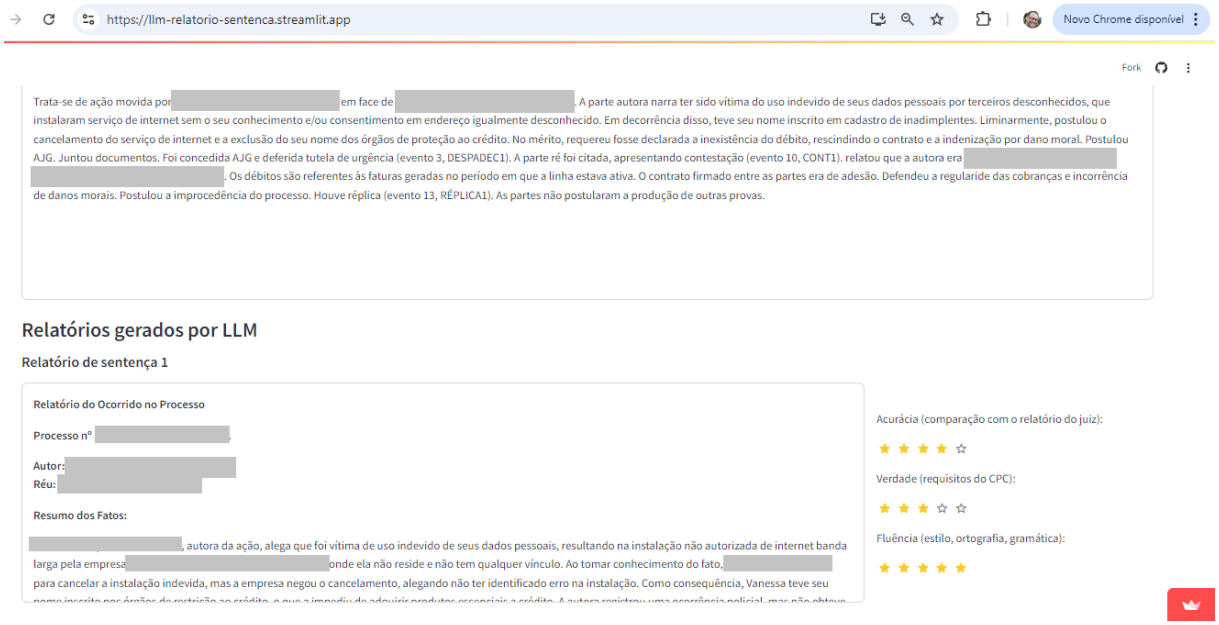
5.2.4.3 Human evaluation

In addition to NLP metrics and LLM-based evaluations, we conducted a human evaluation involving three experienced judges and two judges' assistants from the Court of Justice of Rio Grande do Sul. These professionals regularly draft legal decisions. Prior to the evaluation, they aligned on how to apply MQM metrics, what aspects to consider, and how to rate identified

issues. Each instance included a human-written reference report and the AI-generated report for the same legal case.

To facilitate the process, we developed a web application[11] where evaluators upload an Excel file containing the reports to be assessed, as shown in Figure 57.

Figure 57: Web application for human assessment



Source: Elaborated by the author

Each of the five legal experts assessed 25 reports: 20 unique and 5 common across all evaluators. Thus, all 105 reports in the dataset received human evaluation, and five were reviewed by all five experts for inter-annotator comparison. The average evaluation time per expert was approximately 120 minutes.

5.2.4.4 MQM evaluation framework

Both LLM-as-a-Judge and human evaluation applied the Multidimensional Quality Metrics (MQM) framework (BURCHARDT, 2013; MARIANA, 2014; CORTES et al., 2024), known for its fidelity to human judgment and ability to evaluate distinct quality aspects. MQM typically categorizes issues into five branches, with the first three forming the MQM Core (BURCHARDT, 2013):

- **Accuracy:** Fidelity to the content of the reference.
- **Verity:** Adherence to real-world or legal requirements.
- **Fluency:** Language quality, independent of translation status.

[11] `https://llm-relatorio-sentenca.streamlit.app/`

- **Design:** Formatting and layout issues.
- **Internationalization:** Localizability and region-specific conventions.

We customized MQM to focus on Accuracy, Verity, and Fluency, using the following evaluation criteria:

- **Accuracy / Terminology:** Use of required legal terms.
- **Accuracy / Omission:** Missing mandatory content.
- **Accuracy / Addition:** Unwarranted content additions.
- **Verity / Legal Requirements:** Compliance with the Civil Procedure Code (BRASIL, 2015).
- **Fluency / Style:** Formality and clarity in line with plain language principles (INOVAçãO., 2021).
- **Fluency / Spelling, Grammar, Locale:** Linguistic correctness and adherence to Brazilian standards.

To simplify LLM-as-a-Judge evaluation, we mapped issue severity to penalty scores: 5 (major), 3 (minor), and 0 (none). The full prompt is detailed in Appendix A.2.

Penalty structure for LLM-as-a-Judge:

- **Accuracy:**
  - 5: Serious mismatch with the reference (e.g., different court case).
  - 3: Omission or inclusion of inappropriate legal terms.
  - 0: Faithful representation of the same case and proper terminology.
- **Verity:**
  - 5: Missing essential legal elements (e.g., party names, action type).
  - 3: Incomplete reporting of relevant case details (e.g., hearings, expert reports).
  - 0: Full compliance with legal reporting requirements.
- **Fluency:**
  - 5: Major issues, such as structural interruptions.
  - 3: Minor issues (e.g., grammar, spelling, date formats).
  - 0: No issues; clear, correct, and properly formatted text.

For human evaluation, we adapted MQM to a 5-star rating system, commonly used on the Web, for the three criteria: Accuracy, Verity, and Fluency (respectively, *Acurácia*, *Verdade* and *Fluência*, in Portuguese). Ratings were converted to MQM penalties using: $MQMscore = 5 - \text{stars}$. Thus, a 5-star rating corresponds to zero penalty. Although this conversion is a simplified and non-standard interpretation of the MQM framework, it was chosen to make the evaluation process more accessible and time-efficient for the participating legal professionals. The original MQM requires a detailed annotation of individual error types, which would have demanded extensive time and specialized training from evaluators. Given that our reviewers are highly qualified but not trained in MQM and often face time constraints, the 5-star approach provided a more practical alternative. This adaptation preserved the multidimensional structure of MQM and enabled consistent comparative assessments across generated outputs.

We define the overall MQMScore metric as follows:

$$AP = \frac{1}{n}\sum_{i=1}^{n} Accuracy \quad VP = \frac{1}{n}\sum_{i=1}^{n} Verity \quad FP = \frac{1}{n}\sum_{i=1}^{n} Fluency$$

$$MQMScore = 100 - AP - VP - FP$$

Where:

- $n = 105$ for LLM-as-a-Judge; $n = 125$ for human evaluation;
- AP: Average Accuracy penalty;
- VP: Average Verity penalty;
- FP: Average Fluency penalty;
- $MQMScore$: Final quality score (higher is better).

5.2.5 Experiment Results and Discussion

In the following topics, we demonstrate and discuss the results obtained from evaluating the Judgment Report Generator application. We conducted three levels of evaluations to validate this work: first using traditional NLP metrics, second using LLM-as-a-Judge, and, finally, third through human validation by judges and their judicial assistants.

5.2.5.1 Experiment description

To build the dataset used for the evaluation, we first randomly selected 105 already sentenced legal proceedings, the sentences of which were written between July 2023 and July 2024. From each case, we extracted the judgment report written by a judge.

Next, we developed a script that goes through the selected legal proceedings and generates the judgment report automatically for each one, the same way that a judge can do by using the Judgment Report Generator, as described in the previous section. The script only selects documents with configured prompts.

Prompt version 1 was defined through iterative testing conducted by key business users using the LLM Chat Playground. During these sessions, users evaluated the model's performance in generating judgment reports and identified an initial set of prompts suitable for implementation.

Prompt version 2 was developed as a refinement of the initial approach. It incorporates explicit legal requirements drawn from the Brazilian Code of Civil Procedure, which states that the judgment report must contain the names of the parties, the identification of the case, a summary of the claim and the defense, and a record of the main procedural events. Furthermore, this version includes a model example of a judgment report to guide the structure and content of the generated outputs. This version was designed to increase the consistency of the output and the adherence to judicial standards. The goal of prompt version 2 is to ensure greater legal compliance, clarity, and uniformity in the reports produced by the system.

Table 23: Models and prompt versions

Model ID	Model name	Prompt version
gpt4o1	GPT-4o	v1
gpt4om1	GPT-4o-Mini	v1
llama1	Meta-Llama-3.1-8B-Instruct 4bit	v1
gpt4o2	GPT-4o	v2
gpt4om2	GPT-4o-Mini	v2
llama2	Meta-Llama-3.1-8B-Instruct	v2

Source: Elaborated by the author.

We evaluated two prompt versions across three LLMs: GPT-4o/Azure, GPT-4o-Mini/Azure, LLama 3.1 8b Instruct 4bit/Huggingface, totaling six configurations (Table 23). All prompt templates are provided in Appendix A.1.

The final dataset contains 105 judgment reports written by judges and, for each of them, six automated judgment reports generated by the Judgment Report Generator, totaling 630 rows. Table 24 illustrates the dataset structure.

It is worth noting that the automatically generated judgment reports only consider documents with configured prompts; that is, they do not use documents that the judge may have considered relevant at the time of manual writing of the judgment report, which can lead to differences between the judgment report written by the judge and the ones generated by AI. This constraint arises from the methodology adopted for dataset construction, where only documents with pre-configured prompts could be systematically processed without requiring legal experts to manually identify and classify relevant inputs for each case. While this choice introduces a potential selection bias, particularly the underrepresentation of edge cases or more complex

Table 24: Dataset structure

Row	Human Judgment Report	LLM Judgment Report
1	Human report for case #1	gpt4o1 report for case #1
2	Human report for case #1	gpt4om1 report for case #1
3	Human report for case #1	llama1 report for case #1
4	Human report for case #1	gpt4o2 report for case #1
5	Human report for case #1	gpt4om2 report for case #1
6	Human report for case #1	llama2 report for case #1
...	...	...
630	Human report for case #105	llama2 report for case #105

Source: Elaborated by the author.

litigation scenarios, we believe the impact on the overall evaluation is limited, as such cases represent a minority in the broader judicial workflow. Furthermore, in real-world usage, judgment report generation tools are also typically applied to documents with predefined prompt configurations, aligning the experimental setup with common practical deployment.

5.2.5.2 Automated evaluation with NLP metrics

We applied 18 automated metrics to compare each AI-generated report with the human-written reference. These include string-based metrics (ROUGE, BLEU, METEOR), embedding-based metrics (BERTScore, BLEURT), and cosine/Euclidean distances using different embedding models (Table 25).

Each of the 105 cases was scored across all six model-prompt combinations, totaling 11,340 metrics $(18 \times 6 \times 105)$. All raw results are available on GitHub[12].

Table 26 summarizes how often each configuration produced the best result for each NLP metric. GPT-4o-Mini with prompt v2 outperformed other configurations in 8 of 10 metrics.

Table 27 reports the average results across all cases for traditional NLP metrics only, deliberately excluding embedding distance measures, which are analyzed separately. GPT-4o-Mini with prompt v2 outperformed other configurations in the majority of these metrics.

We also evaluated semantic similarity using embedding distance metrics, comparing AI-generated reports to the judge-written references. Lower scores indicate greater similarity. We employed OpenAI's text-embedding-ada-002 and six additional models from Huggingface and Sentence Transformers (see Table 25).

Table 28 summarizes the best-performing models per metric. GPT-4o-Mini with prompt v2 outperformed all other configurations in six out of eigth metrics, while GPT-4o with prompt v2 led in two.

Table 29 presents the mean scores per configuration. Again, GPT-4o-Mini with prompt v2

[12]https://github.com/lucianozanuz/Manuscript-Number-JJIMEI-D-25-00093—Supplementary-repository/tree/main/results.

Table 25: Automated metrics used

Alias	Metric	Model/Description
rouge1	ROUGE-1	Unigram (1-gram) based scoring
rouge2	ROUGE-2	Bigram (2-gram) based scoring
rougeL	ROUGE-L	Longest common subsequence based scoring
rougeLs	ROUGE-Lsum	Longest common subsequence based scoring splits text using "\n"
bleu	BLEU	N/A
meteor	METEOR	N/A
bertP	BERTScore Precision	N/A
bertR	BERTScore Recall	N/A
bertF	BERTScore F1	N/A
bleurt	BLEURT	bleurt_checkpoint = BLEURT-20
ada1	Cosine distance	openai/azure/text-embedding-ada-002
ada2	Euclidean distance	openai/azure/text-embedding-ada-002
hf1	Cosine distance	sentence-transformers/all-mpnet-base-v2
hf2	Cosine distance	sentence-transformers/paraphrase-multilingual-mpnet-base-v2
hf3	Cosine distance	sentence-transformers/paraphrase-multilingual-MiniLM-L12-v2
hf4	Cosine distance	intfloat/multilingual-e5-base
hf5	Cosine distance	intfloat/multilingual-e5-large
hf6	Cosine distance	stjiris/bert-large-portuguese-cased-legal-tsdae-gpl-nli-sts-MetaKD-v1

Source: Elaborated by the author.

Table 26: Best NLP metrics for the complete dataset

Metric	gpt4o1	gpt4om1	llama1	gpt4o2	gpt4om2	llama2
rouge1	0	1	34	27	**43**	0
rouge2	0	1	8	46	**48**	2
rougeL	0	1	9	35	**57**	3
rougeLs	1	13	15	25	**49**	2
bleu	0	0	6	44	**55**	0
meteor	0	2	9	**47**	44	3
bertP	0	0	13	36	**56**	0
bertR	0	1	4	37	**62**	1
bertF	0	0	4	35	**65**	1
bleurt	20	13	4	**41**	23	4

Source: Elaborated by the author.

Table 27: Mean NLP metrics for the complete dataset

Metric	gpt4o1	gpt4om1	llama1	gpt4o2	gpt4om2	llama2
rouge1	0.2762	0.3788	0.3907	0.3876	**0.4269**	0.3194
rouge2	0.1333	0.1584	0.1489	**0.2116**	0.2088	0.1225
rougeL	0.1657	0.2161	0.2174	0.2575	**0.2713**	0.1836
rougeLs	0.2214	0.2767	0.2560	0.2796	**0.3011**	0.2200
bleu	0.0383	0.0495	0.0592	0.1061	**0.1062**	0.0439
meteor	0.0301	0.0411	0.0486	**0.0849**	0.0746	0.0356
bertP	0.6720	0.6837	0.7054	0.7263	**0.7316**	0.6794
bertR	0.6971	0.7042	0.7012	0.7363	**0.7428**	0.7012
bertF	0.6842	0.6937	0.7032	0.7311	**0.7370**	0.6900
bleurt	-0.1344	-0.1462	-0.1811	**-0.1167**	-0.1373	-0.1726

Source: Elaborated by the author.

Table 28: Best embedding distance metrics

Metric	gpt4o1	gpt4om1	llama1	gpt4o2	gpt4om2	llama2
ada1	5	2	1	**61**	36	0
ada2	5	2	1	**61**	36	0
hf1	8	12	14	28	**31**	12
hf2	0	2	8	42	**53**	0
hf3	2	0	18	36	**49**	0
hf4	1	3	7	38	**53**	3
hf5	5	4	4	27	**53**	12
hf6	3	8	13	31	**47**	3

Source: Elaborated by the author.

achieved the highest similarity in all Huggingface-based metrics, confirming that prompt v2 yields more semantically aligned outputs than prompt v1.

Table 29: Mean embedding distance metrics

Metric	gpt4o1	gpt4om1	llama1	gpt4o2	gpt4om2	llama2
ada1	0.0520	0.0521	0.0584	**0.0413**	0.0432	0.0724
ada2	0.3214	0.3217	0.3396	**0.2854**	0.2926	0.3776
hf1	0.1918	0.1748	0.1954	0.1651	**0.1541**	0.2182
hf2	0.2694	0.2382	0.2281	0.1514	**0.1433**	0.3715
hf3	0.3717	0.3378	0.2857	0.2083	**0.1959**	0.4885
hf4	0.0744	0.0704	0.0719	0.0619	**0.0594**	0.0789
hf5	0.0734	0.0718	0.0734	0.0666	**0.0624**	0.0764
hf6	0.2156	0.1804	0.1942	0.1692	**0.1543**	0.2196

Source: Elaborated by the author.

The results of the automated metrics, including all NLP and embeddings distance metrics, comparing judgment reports written by the judge and the one generated by the Judgment Report Generator are:

- Prompt v2 outperformed v1 in 100% of the metrics;
- GPT-4o-Mini with prompt v2 ranked first in 14 of 18 (77.78%) metrics;
- GPT-4o with prompt v2 ranked first in 4 of 18 (22.22%) metrics.

5.2.5.3 LLM-as-a-Judge evaluation

We also evaluated the quality of AI-generated judgment reports using the LLM-as-a-Judge approach (LI et al., 2024; ZHENG et al., 2023; OPENAI, 2024), with GPT-4o serving as the evaluator and a prompt detailed in Appendix A.2. This evaluation followed the MQM framework (BURCHARDT, 2013), as outlined in Section 5.2.4, focusing on three core dimensions: Accuracy, Verity, and Fluency. The corresponding scores are presented in Table 30.

Table 30: LLM-as-a-Judge MQM results

Metric	gpt4o1	gpt4om1	llama1	gpt4o2	gpt4om2	llama2
Accuracy	3.0857	3.1333	3.4190	**3.0571**	**3.0571**	3.4286
Verity	3.0571	3.1333	3.4381	**2.8571**	2.9714	3.4667
Fluency	0.5143	0.3429	2.1143	0.2857	**0.1143**	2.3619
MQMScore	93.3429	93.3905	91.0286	93.8000	**93.8571**	90.7429

Source: Elaborated by the author.

GPT-4o-Mini with prompt v2 received the highest overall score, with the best fluency and joint-best accuracy. Again, prompt v2 consistently outperformed prompt v1.

5.2.5.4 Human evaluation

Five legal professionals from the Court of Justice of Rio Grande do Sul evaluated 25 reports each using the same MQM dimensions. Reports from Meta-Llama models were excluded due to low prior scores. Table 31 shows the average results.

Table 31: MQM human evaluation results

Metric	gpt4o1	gpt4om1	gpt4o2	gpt4om2
Accuracy	1.7520	1.8320	1.1600	**1.1120**
Verity	1.2720	1.3920	1.1200	**1.1120**
Fluency	2.6960	2.3680	1.4640	**1.4080**
MQMScore	94.2800	94.4080	96.2560	**96.3680**

Source: Elaborated by the author.

As with previous evaluations, GPT-4o-Mini with prompt v2 ranked highest across all dimensions.

A subset of five reports was reviewed by all experts to test agreement. In this small sample, GPT-4o with prompt v2 narrowly outperformed GPT-4o-Mini, likely due to sample size effects (Table 32).

Table 32: MQM human evaluation with shared subset

Metric	gpt4o1	gpt4om1	gpt4o2	gpt4om2
Accuracy	1.6400	1.4000	**0.6400**	1.2000
Verity	1.3200	**0.8400**	0.9600	1.4000
Fluency	2.8800	2.3600	**0.9200**	1.4000
MQMScore	94.1600	95.4000	**97.4800**	96.0000

Source: Elaborated by the author.

To better understand the subjectivity in human assessment, we conducted a brief debriefing with the five legal experts. The interviews revealed differing views on what defines a high-quality judgment report: some prioritized detail and comprehensiveness, while others valued conciseness and clarity.

Despite this, all five experts independently ranked GPT-4o-Mini with prompt version 2 as the best-performing configuration, reinforcing the consistency and reliability of the results.

Interestingly, GPT-4o-Mini outperformed the larger GPT-4o model in this task. We attribute this to the well-defined nature of the task and the use of carefully engineered prompts, which may reduce performance gaps between models. In such specialized scenarios, smaller models can be not only competitive but also preferable when efficiency is a concern.

To minimize bias, evaluators were blinded to the source of each report. A subset of five reports was assessed by all experts to verify consistency. Slight variations in rankings were

observed, likely due to the small sample size and subjective interpretation. These findings highlight the importance of larger evaluation sets to mitigate individual variability.

5.2.5.5 Correlation between evaluation methods

This study applied four distinct evaluation approaches to assess the quality of judgment reports: traditional string-based NLP metrics, machine learning-based metrics, LLM-as-a-Judge, and human expert evaluation. Although these methods differ fundamentally in how they assess text quality, ranging from surface-level token overlap to deep semantic reasoning and expert judgment, they all converged on the same result: the GPT-4o-Mini model with prompt version 2 consistently produced the best outputs.

There are several possible explanations for this convergence. First, prompt version 2 provided a well-structured and legally grounded template, which likely led to more coherent and complete outputs across models. Because this version explicitly included legal requirements and a concrete example, even smaller models could generate more accurate and legally adequate reports, which would be reflected in all types of evaluation.

Second, while string-based metrics (e.g., ROUGE, BLEU) focus on n-gram overlap, and machine learning-based metrics (e.g., BERTScore, BLEURT) assess semantic similarity, both are positively influenced by fluent and well-organized text. Prompt version 2 likely guided models toward producing outputs that were simultaneously lexically and semantically aligned with reference texts, leading to better scores in both metric groups.

Third, LLM-as-a-Judge evaluations reward clarity, factual consistency, and domain appropriateness. Since version 2 prompts included domain-specific structure and legal framing, this likely enhanced the quality of the generated text in ways that align with what LLMs are trained to value. Similarly, human experts—judges and legal assistants—recognized the added structure and completeness provided by prompt version 2, independently scoring those outputs higher despite individual preferences.

In summary, the convergence observed across all evaluation methods suggests that prompt design played a central role in shaping the quality of the model outputs. A well-defined prompt with domain knowledge can act as a strong equalizer, helping even smaller models generate outputs that are perceived as superior across diverse evaluation frameworks.

5.2.5.6 Discussion

This study employed a multi-faceted evaluation strategy to assess the performance of the proposed Judgment Report Generator, highlighting its innovative approach and differentiation. The evaluation incorporated a combination of automated NLP metrics, the LLM-as-a-Judge technique, and human assessments by five legal experts. This robust methodology enabled the comparison of AI-generated judgment reports with those authored by judges in real legal

proceedings, ensuring real-world relevance and practical applicability.

5.2.5.6.1 Differentiated evaluation approach

A key innovation in this work lies in the comprehensive evaluation framework. Traditional string-based metrics like ROUGE, BLEU, and METEOR were augmented by machine learning-based metrics such as BERTscore, which assess semantic similarity rather than just literal overlap. The incorporation of the novel evaluation technique LLM-as-a-Judge, based on the MQM framework, provided a deeper qualitative and contextual assessment of the generated reports. This method allowed for nuanced feedback, including error categorization and explanation, which are particularly relevant in the legal domain where precision and adherence to formal requirements are critical.

5.2.5.6.2 Text processing for automation

The Judgment Report Generator was designed with automation in mind, leveraging advanced text processing methodologies. By integrating specific prompts tailored to extract and summarize key elements from legal documents, such as party names, action types, claims, and procedural events, the system ensures compliance with the Code of Civil Procedure (*CPC - Código do Processo Civil*, in Portuguese). This structured approach minimizes the need for manual intervention, streamlining the process and enabling seamless integration into electronic lawsuit platforms.

To assess the scalability of the proposed solution, we measured its computational performance and estimated costs during the automated generation of 105 judgment reports using the GPT-4o-Mini model with prompt version 2. The total processing time was approximately 9,211 seconds, with a mean time of 87.7 seconds per report (minimum: 22.9 seconds; maximum: 183.9 seconds), suggesting the system is capable of producing reports in near real time.

In terms of cost, the generation consumed a total of 1,857,520 tokens (1,604,546 prompt tokens and 252,974 completion tokens), corresponding to an API usage cost of approximately US$0.39 for the entire batch. In contrast, manual generation of these reports by judicial analysts can take an estimated 30 minutes per report, resulting in an estimated labor cost of US$4.77 per report, based on official salary data from the judiciary of the State of Rio Grande do Sul[13]. These figures indicate substantial efficiency and cost advantages, supporting the claim of scalability and demonstrating the feasibility of deploying this system at scale within judicial workflows.

[13]Law No. 15.737/2021 of the State of Rio Grande do Sul.

5.2.5.6.3 Findings and implications

The results showed that prompt version 2, which incorporates explicit legal requirements and examples of real judgment reports, significantly outperformed version 1 across all evaluation techniques. This improvement underscores the importance of precise and detailed prompt engineering in maximizing the performance of LLMs. Furthermore, contrary to expectations based on public benchmarks, the GPT-4o-Mini model surpassed GPT-4o in this specific legal application. This finding highlights the necessity of validating AI solutions with application-specific datasets rather than relying solely on generic benchmarks.

5.2.5.6.4 Practical recommendations

Given the high cost and complexity of human evaluations and LLM-as-a-Judge assessments, traditional NLP metrics remain a viable and cost-effective option for initial evaluations. Semantic metrics, such as BERTscore and embeddings-based distances, offer a balanced trade-off between complexity and insight. For broader analyses, a combination of metrics is recommended to ensure a comprehensive understanding of results. When deeper qualitative insights or explainability are required, LLM-as-a-Judge provides a valuable tool for detailed evaluation.

5.2.5.6.5 Future opportunities

The current production system, based on prompt version 1, shows potential for improvement with the adoption of version 2. Additionally, there is significant room for further refinement by testing more sophisticated prompts and exploring alternative LLMs tailored to legal applications. Future work could focus on iterative experimentation with prompt designs and the integration of more advanced techniques for legal text summarization, ensuring the generator remains at the forefront of AI-driven legal automation.

5.2.5.6.6 Conclusion

In summary, this work demonstrates the effectiveness of combining diverse evaluation techniques with tailored text-processing strategies to create an automated solution ready for real-world deployment. By addressing the specific needs of the legal domain and employing a differentiated evaluation methodology, this study sets a new standard for applying AI to legal text generation, paving the way for broader adoption and continued innovation.

5.2.6 Hallucination Analysis and Legal Risk Considerations

Although hallucinations, i.e., the generation of factually incorrect or misleading content, were not explicitly measured in our experimental evaluation, they remain a relevant concern in real-world deployments of generative AI for legal drafting. In the context of judgment report generation, such errors could compromise the factual accuracy or legal coherence of the document if not properly reviewed.

However, according to CNJ (Brazilian National Council of Justice) Resolution No. 615/2025 (CNJ, 2025), the use of generative AI to produce supporting texts for judicial acts is considered low risk, as long as the final document is reviewed, edited and validated by a judge. The proposed system follows these guidelines, since all results are designed for human oversight and remain fully editable, with clear expectations of judicial oversight.

Therefore, while the legal risks associated with hallucinations are real, their practical impact is mitigated through structured prompt design, data anonymization, and compliance with the CNJ's transparency and responsibility requirements. Continued human oversight ensures that AI serves only as an assistive tool, maintaining legal integrity and accountability in judicial workflows.

5.2.7 Conclusions

Artificial intelligence in Law is gaining traction with recent advances in natural language processing, particularly with the emergence of Large Language Models (LLMs) such as ChatGPT. These new capabilities open opportunities for AI-supported use cases in the judiciary. In Brazil, the significant increase in the percentage of electronic lawsuits (Figure 54) (CNJ, 2021), considering 35.1 million new cases were filed with the courts electronically only in 2023 (CNJ, 2024a), consists in an opportunity to use Generative AI to process and generate texts from legal proceedings.

In this study, we present a practical application of Generative AI for the automated generation of judgment reports, integrated into the lawsuit system of the Court of Justice of Rio Grande do Sul. The solution includes a Judgment Report Generator and a prompt prototyping playground. While related work explores LLMs in legal settings, this study addresses the specific challenge of generating judgment reports in Portuguese, aligned with Brazilian legal standards.

A key aspect of this work is its evaluation strategy, which applies multiple layers of analysis. We constructed a private validation dataset comprising 105 judgment reports authored by judges. The evaluation combined traditional NLP metrics, semantic similarity metrics, and two assessments based on the MQM (Multidimensional Quality Metrics) framework: the recently proposed LLM-as-a-Judge technique and detailed human evaluations conducted by judges and their assistants. This triangulated evaluation approach allowed for a more comprehensive un-

derstanding of model performance.

The system's prompt design was central to achieving coherent and compliant outputs. Prompts were structured to extract key legal information, such as parties, claims, and procedural elements, while guiding the generation process according to legal norms, particularly the Brazilian Code of Civil Procedure (*CPC - Código do Processo Civil*). The comparative analysis between prompt versions showed that incorporating legal framing and example-based guidance led to improved quality in the generated reports.

Interestingly, the GPT-4o-Mini model outperformed the larger GPT-4o model in this task. This result suggests that, for domain-specific and well-scoped tasks, smaller models can be competitive, particularly when using well-engineered prompts, offering a more efficient alternative to resource-intensive LLMs.

The main contributions of this study include:

- Domain-specific AI tools. The development of tailored LLM prompts for generating judgment reports in Portuguese, compliant with Brazilian legal standards, alongside a set of prompts for the LLM-as-a-Judge evaluation framework.

- Innovative evaluation methods. A demonstration that traditional NLP metrics, when combined with semantic methods like BERTscore and advanced techniques like LLM-as-a-Judge, can effectively correlate with human evaluations.

- Efficient AI applications. Evidence that smaller, resource-efficient models like GPT-4o-Mini can deliver superior results in legal applications, challenging the paradigm of "bigger is better."

Future work will explore advanced prompting strategies (SAHOO et al., 2024; VATSAL; DUBEY, 2024) and Retrieval-Augmented Generation (RAG) approaches (GAO et al., 2024) to further improve output quality. Extending the system to additional legal tasks and jurisdictions may support broader adoption of AI technologies in the legal domain.

By integrating current AI techniques into a functional system with real-world legal data, this study contributes to ongoing efforts to enhance judicial processes through automation. The value of this work lies not in isolated novelty, but in its practical implementation, structured evaluation approach, and contribution to the underexplored area of Portuguese-language legal AI applications.

While this study is grounded in the Brazilian legal system, including specific laws, procedural rules, and language, the underlying methodology and system architecture may serve as a reference for similar initiatives in other jurisdictions. We acknowledge, however, that legal traditions and procedural frameworks differ across countries. As such, transferring this approach to other legal systems would require careful adaptation to local laws, judicial structures, and linguistic nuances.

5.3 Text generation task fine-tuned in judicial context

In this experiment, we fine-tuned in the judicial context a Portuguese language model pre-trained on a text generation task, based on a GPT-2 architecture. This experiment is positioned in the third layer of the proposed architecture, called as context adaptation layer. This is different from the experiment presented in the previous section, which is positioned in the fourth layer of the proposed architecture.

The pre-trained language model used in this experiment was GPorTuguese-2 (GUILLOU, 2020), a GPT-2 small language model for Portuguese (gpt2-small-portuguese)[14]. GPorTuguese-2 is a state-of-the-art language model for Portuguese based on the GPT-2 small model. This model was not trained from scratch, instead it was trained on Portuguese Wikipedia using transfer learning and fine-tuning techniques in just over a day, on one GPU NVIDIA V100 32GB and with a little more than 1GB of training data based on English pre-trained GPT-2 small[15], whose objective was being a proof-of-concept that it is possible to get a state-of-the-art language model in any language with low resources.

In this experiment, we used two judicial text corpora: Acordaos TCU corpus[16], which comprises judgments from the Brazilian Federal Court of Accounts during the period from 1992 to 2019, and a private lawyer's petition corpus from a Brazilian State Court with more than 18 thousand initial petitions. The two corpora have essentially different kind of texts. The Acordaos TCU corpus has texts from judges, with the main goal of describing a judgment. The initial petitions corpus has texts from lawyers, with the fundamental goal of describing a request for the judicial court. This difference is important to prove that the language model could capture the nuances of different types of contexts.

The experiment was conducted with only a few part of each corpus, due to low resources and time available. This was just an initial experiment to see if even with this little corpus the language model would be affected. In the future, appropriate and more scientifically rigorous experiments will be conducted.

[14] https://huggingface.co/pierreguillou/gpt2-small-portuguese.
[15] https://huggingface.co/gpt2.
[16] https://www.kaggle.com/ferraz/acordaos-tcu.

Table 33: Comparison of responses of text generation experiment from the Huggingface hosted inference API

Input text	**gpt2-small-portuguese**	**gpt2-small-portuguese-finetuned-tcu-acordaos**	**gpt2-small-portuguese-finetuned-peticoes**
Maria das Dores, nascida em 4 de agosto de 1961,	Maria das Dores, nascida em 4 de agosto de 1961, é uma psicóloga, advogada e política chilena. Foi ministra de Direitos Humanos e Legislação na Universidad Católica. É casada com o ex-deputado federal Eduardo Augusto de la Moravia.	Maria das dores, nascida em 4 de agosto de 1961, em virtude de prejuízos causados pela "Sra. Maria Auxiliadora de Melo e outros às fls. 19/18" (do extinto Ministério da Ação Social), de responsabilidade do Sr.	Maria das Dores, nascida em 4 de agosto de 1961, vem correndo nos dois lados do Rio Paraná, hábeis e sem condições em suas terras, sempre procurando, a todos os bens. É difícil, portanto, o sustento da autora; o que
Quem era José Carlos Ribeiro? José Carlos Ribeiro era um	Quem era José Carlos Ribeiro? José Carlos Ribeiro era um advogado brilhante e administrador, o maior inimigo do estado monárquico português e de que tinha-se empenhado em impedir a sucessão desse. No ano de 1834, Joaquim José Ribeiro passou a integrar	Quem era José Carlos Ribeiro? José Carlos Ribeiro era um sócio-proprietário da Companhia Agropecuária do Rio Claro, do qual era representante. Considerando que, no processo devidamente organizado, apurou-se contra o responsável débito na quantia de Cz$	Quem era José Carlos Ribeiro? José Carlos Ribeiro era um dos maiores empresários do Estado do Rio Grande do Sul, nascido em 18 de março de 1916. Nascido em Caxias do Sul, foi criado em Caxias do Sul e falecido em 1998, no Hospital Santa
O condomínio Jardim Planalto, na cidade de Porto Alegre,	O condomínio Jardim Planalto, na cidade de Porto Alegre, é um conjunto arquitetônico composto de diversos edifícios e apartamentos populares, localizados em áreas de serviços públicos, como saúde, educação ambiental, educação comunitária e comércio. O local teve a sua história bastante	O condomínio Jardim Planalto, na cidade de Porto Alegre, foi adquirido e, em seu início, mantido pela responsável, no valor de Cr$ 1.720.000.000,00 (um milhão, quinhentos e sessenta e dois mil, um	O condomínio Jardim Planalto, na cidade de Porto Alegre, CEP 83.0610-000, situa-se numa antiga chácara, de aproximadamente 700 metros quadrados, onde foram encontrados documentos, fotos de empregados e outras fotos do casal. Em razão disso

O shopping center	O shopping center foi fundada por iniciativa do ex-diretor da Companhia City Brasil, Luiz Paulo Nunes, com a finalidade centralizadora de servir aos habitantes da cidade do Rio de Janeiro. A partir de 2015, a shopping centers já conta com	O shopping center no Parque da Independência, ao lado do terminal rodoviário no município, mediante a cobrança judicial de saldo devedor, no valor de R$ 5.410.000,00 (cinco milhões de cruzeiros), contra a prefeitura, mediante, no	O shopping center, mais especificamente a Avenida Pasteur - Tatuapé Sul + Caxias do Sul (antiga Av. Porto Alegre - CEP 90.010-00) - que possui uma área construída em área de 30 hectares, além da
O advogado	O advogado e professor da Universidade Federal de Santa Catarina, na década de 1990, foi um dos alunos do curso de pós-graduações no departamento de direito penal da UFSC, durante sua carreira profissional. Em 1991 foi a primeira pessoa a	O advogado e ex-Prefeito do Município de Várzea Grande - BA, Dr. Hélio de Assis Souza, instaurada em razão de sua omissão no dever de prestar contas da prestação de contas dos recursos transferidos da extinta Superintendência Federal de Atendimento aos Clin	O advogado de modo a evitar a impossibilidade do requerente de cumprir a obrigação de prestar mais atenção, em nome de seu enteado advogado do Direito. Assim, a tutela judiciária de urgência não é o único recurso que é invocado nos autos. Assim

Source: Elaborated by the author.

The result of the experiment generated two new models: gpt2-small-portuguese-finetuned-tcu-acordaos[17] and gpt2-small-portuguese-finetuned-peticoes[18]. Table 33 shows five different input texts in Portuguese and respective responses from the Huggingface hosted inference Application Programming Interface (API) for both the new models and the gpt2-small-portuguese model. The responses shows that the fine-tuning in fact adapted the context of each model to the new corpus, even though the fine-tuning was done with little data.

The gpt2-small-portuguese responses generated generic texts, with no specific context to be noticed. The gpt2-small-portuguese-finetuned-tcu-acordaos responses visibly contained references to financial related texts, what make sense considering its origin from the Court of Accounts, also including mentions of older Brazilian currency before Real, according to texts from judgments since 1992, so before the Real currency be instituted in Brazil. The same way, the gpt2-small-portuguese-finetuned-peticoes responses, which was fine-tuned with texts from lawyers from southern Brazil, contained common terms in lawyers' vocabulary and references to cities and locales from southern Brazil. Thereby, this preliminary analysis confirms that the

[17] https://huggingface.co/Luciano/gpt2-small-portuguese-finetuned-tcu-acordaos.
[18] https://huggingface.co/Luciano/gpt2-small-portuguese-finetuned-peticoes.

context adaptation of a language model works, even with fine-tuning with little data.

5.4 Fill-mask task fine-tuned in judicial context

Similar to the previous experiment, we also fine-tuned in the judicial context a pre-trained Portuguese language model based on a BERT architecture. This experiment is also positioned in the context adaptation layer, with the difference being the objective of the NLP task, this time with a fill-mask objective.

The pre-trained language model used in this experiment was the base version of BERTimbau (SOUZA; NOGUEIRA; LOTUFO, 2020), the first pre-trained BERT model for Brazilian Portuguese, trained from scratch on the brWaC Corpus (WAGNER FILHO et al., 2018).

In this experiment, we used the same two judicial text corpora than the previous experiment, and also fine-tuning with only a few part of each corpus.

The result of the experiment generated two new models: bert-base-portuguese-cased-finetuned-tcu-acordaos[19] and bert-base-portuguese-cased-finetuned-peticoes[20]. Table 34 five different input texts in Portuguese with a [MASK] token and respective responses from the Huggingface hosted API for both the new models and the bert-base-portuguese-cased model. Each response contains the top five [MASK] token prediction scores. The responses shows that the fine-tuning in fact adapted the context of each model to the new corpus, even though the fine-tuning was done with little data.

For the same input text, the inferences always show the same result, so that is possible to confirm that models have different weights due to fine tuning. As the model's response is just a masked word, differentiating the results is a little more difficult than in the previous experiment, however, with the scores it is possible to see that there are important differences in the models.

The bert-base-portuguese-cased responses generated more generic tokens, with no specific context to be noticed, but apparently better results than the generic gpt2-small-portuguese model from the previous experiment. This could be due to the size of each model, base versus small, and the technique used, respectively from scratch and finetuning from the English model. It is possible to see that the bert-base-portuguese-cased-finetuned-tcu-acordaos responses contained more weight in tokens from the Court of Accounts texts. The same way, the bert-base-portuguese-cased-finetuned-peticoes responses contained more weight in tokens common in lawyers' texts. Therefore, this preliminary analysis also confirms that the context adaptation of a language model works, even with fine-tuning with little data.

[19] https://huggingface.co/Luciano/bert-base-portuguese-cased-finetuned-tcu-acordaos.
[20] https://huggingface.co/Luciano/bert-base-portuguese-cased-finetuned-peticoes.

Table 34: Comparison of responses of fill-mask experiment from the Huggingface hosted inference API

Input text	**bert-base-portuguese-cased model**		**bert-base-portuguese-cased-finetuned-tcu-acordaos**		**bert-base-portuguese-cased-finetuned-peticoes**	
	[MASK] prediction	**Score**	**[MASK] prediction**	**Score**	**[MASK] prediction**	**Score**
O Tribunal de [MASK]	Justiça	0.672	Contas	0.901	Justiça	0.749
	Contas	0.251	Justiça	0.064	Contas	0.166
	origem	0.006	origem	0.002	origem	0.022
	justiça	0.005	Recursos	0.002	justiça	0.006
	.	0.005	:	0.002	contas	0.002
José Carlos Ribeiro, [MASK], casado,	brasileiro	0.546	brasileiro	0.224	brasileiro	0.715
	advogado	0.063	Brasileiro	0.078	advogado	0.062
	Brasileiro	0.038	advogado	0.071	Brasileiro	0.031
	médico	0.026	médico	0.067	médico	0.015
	empresário	0.022	engenheiro	0.030	empresário	0.012
o valor da [MASK]	multa	0.140	multa	0.469	multa	0.132
	taxa	0.066	dívida	0.068	inscrição	0.079
	inscrição	0.059	inscrição	0.033	taxa	0.043
	parcela	0.043	obra	0.031	parcela	0.033
	compra	0.024	parcela	0.024	franquia	0.033
prestação de [MASK]	serviços	0.575	contas	0.967	serviços	0.460
	contas	0.328	serviços	0.023	contas	0.428
	serviço	0.066	serviço	0.004	serviço	0.085
	informações	0.007	Contas	0.002	informações	0.005
	assistência	0.004	informações	0.001	Serviços	0.004
por intermédio de seus [MASK] signatários, nos autos do processo	respectivos	0.553	respectivos	0.501	advogados	0.731
	advogados	0.234	advogados	0.160	respectivos	0.141
	próprios	0.036	novos	0.061	próprios	0.015
	principais	0.015	demais	0.032	órgãos	0.010
	órgãos	0.011	órgãos	0.031	membros	0.006

Source: Elaborated by the author.

5.5 Finetuning and datasets

During this doctorate, we have trained a total of 62 models, using different strategies and datasets. Some of these models were the result of published works and others were generated from experiments not yet published. All trained models are publicly available on Hugging Face[21]. Some of these experiments are described below.

The goal of all these trainings was to experiment with different fine-tuning strategies and evaluate the fine-tuned models generated, as well as to test the ability to train models using the infrastructure available in the Google Colaboratory Pro version. From this work, it was expected to identify opportunities for work to be carried out.

Table 35 lists models generated from PLMs and LLMs full fine-tuning experiments. As described in section 2.9, full fine-tuning of transformer-based PLMs or LLMs involves training the entire model, including all layers and parameters, in an intensive process that requires substantial computational resources.

Considering the complexity and cost involved in full fine tuning of pre-trained models, especially LLMs, strategies have emerged to deal with this, called Parameter Efficient Fine-Tuning (PEFT) (XU et al., 2023), as described in section 2.9.

One of the techniques that is gaining prominence is LoRA, which uses low-rank decomposition method, which we used in several experiments during this doctorate. Table 36 lists models generated from PLM and LLM parameter-efficient fine-tuning experiments with LoRA (HU et al., 2021) technique.

Another low-rank decomposition method is QLoRA (DETTMERS et al., 2023), which quantizes a model to 4-bits and then trains it with LoRA. In Table 37, we list the models generated from PLM and LLM parameter-efficient fine-tuning experiments with QLoRA (DETTMERS et al., 2023) technique.

Table 38 lists models generated from others technique of parameter-efficient fine-tuning experiments. The methods we used are $(IA)^3$ (LIU et al., 2022) and the soft prompt-based fine-tuning methods Prompt-tuning (LESTER; AL-RFOU; CONSTANT, 2021), Prefix-tuning (LI; LIANG, 2021) and P-tuning (LIU et al., 2023b), more detailed in section 2.9.

To carry out the experiments during this doctorate, in addition to the publicly available datasets, as mentioned, some public and private datasets were generated, as shown in Table 39.

5.6 General analysis of experiments

During the period of completion of this doctorate, several academic experiments were carried out, together with the development of real applications of AI applied to Law were developed for the judiciary. The experiments carried out served to validate the proposed methodology in an iterative process of continuous improvement, that is, from the execution of the experiments,

[21] https://huggingface.co/Luciano

Table 35: Full fine-tuning models

Pre-trained model	Finetuned model (model names start with "Luciano/")
bert-base-portuguese-cased	bert-base-portuguese-cased-finetuned-peticoes
bert-base-portuguese-cased	bert-base-portuguese-cased-finetuned-tcu-acordaos
bert-base-multilingual-cased	bert-base-multilingual-cased-finetuned-lener_br (cont.)[a]
bert-base-multilingual-cased	bert-base-multilingual-cased-finetuned-lener_br
bertimbau-base	bertimbau-base-lener_br
bertimbau-base	bertimbau-large-lener_br
bertimbau-base	bertimbau-base-finetuned-lener-br-finetuned- (cont.)[b]
bertimbau-base	bertimbau-base-finetuned-brazilian_court_ (cont.)[c]
bertimbau-base	bertimbau-base-finetuned-brazilian_court_ (cont.)[d]
bertimbau-base	bertimbau-base-finetuned-brazilian_court_decisions
bertimbau-base	bertimbau-base-finetuned-brazilian_court_ (cont.)[e]
bertimbau-base	bertimbau-base-finetuned-lener-br-finetuned- (cont.)[f]
bertimbau-base	bertimbau-base-finetuned-lener-br
bertimbau-base	bertimbau-base-finetuned-lener-br-finetuned (cont.)[g]
bertimbau-base	bertimbau-base-lener-br-finetuned-lener-br
bertimbau-base	bertimbau-base-finetuned-lener-br-finetuned (cont.)[h]
bloomz-560m	bloomz-560m-lener_br
gpt2-small-portuguese	gpt2-small-portuguese-finetuned-tcu-acordaos
gpt2-small-portuguese	gpt2-small-portuguese-finetuned-peticoes
Llama-2-7b	Llama-2-7b-chat-hf-miniguanaco
Llama-2-7b	Llama-2-7b-chat-peticoes-sfttrainer
Llama-2-7b	Llama-2-7b-chat-hf-dolly-mini
Llama-2-7b	llama-2-7b-guanaco-dolly-mini
RoBERTo	OscarRoBERTo_pt
xlm-roberta-large	xlm-roberta-large-finetuned-lener_br
xlm-roberta-base	xlm-roberta-base-finetuned-lener-br
xlm-roberta-large	xlm-roberta-large-finetuned-lener_br-finetuned-lener-br
xlm-roberta-large	xlm-roberta-large-finetuned-lener-br
xlm-roberta-base	xlm-roberta-base-finetuned-lener_br-finetuned-lener-br
xlm-roberta-base	xlm-roberta-base-finetuned-lener_br

Source: Elaborated by the author.

[a] bert-base-multilingual-cased-finetuned-lener_br-finetuned-lener-br
[b] bertimbau-base-finetuned-lener-br-finetuned-peticoes-grupo_competencia
[c] bertimbau-base-finetuned-brazilian_court_decisions_bt8_ep15
[d] bertimbau-base-finetuned-brazilian_court_decisions_bt4_ep5
[e] bertimbau-base-finetuned-brazilian_court_decisions_bt16_ep15
[f] bertimbau-base-finetuned-lener-br-finetuned-brazilian_court_decisions
[g] bertimbau-base-finetuned-lener-br-finetuned-peticoes-assuntos
[h] bertimbau-base-finetuned-lener-br-finetuned-peticoes-competencia

Table 36: LoRA fine-tuning models

Pre-trained model	Finetuned model
bertimbau-base	Luciano/lora-bertimbau-base-lener_br
bertimbau-large	Luciano/lora-bertimbau-large-lener_br
bloom-560m	Luciano/lora-bloom-560m-lener_br
bloomz-560m	Luciano/lora-bloomz-3b-mt-lener_br
bloomz-560m	Luciano/lora-bloomz-1b7-mt-lener_br
bloomz-560m	Luciano/lora-bloomz-560m-lener_br
flan-t5-large	Luciano/SEQ_2_SEQ_LM_financial_phrasebank (cont)[a]
flan-t5-xxl	Luciano/SEQ_2_SEQ_LM_samsum_philschmid (cont)[b]
lora-bloomz-7b1-mt	Luciano/lora-bloomz-7b1-mt-lener_br
Llama-2-7b	Luciano/lora-4bit-Llama-2-13b-hf-lener_br
Llama-2-7b	Luciano/lora-4bit-Llama-2-7b-hf-lener_br
Llama-2-7b	Luciano/lora-4bit-Llama-2-7b-chat-hf-lener_br
Llama-2-7b	Luciano/lora-llama-7b-hf-lener_br
Llama-2-7b	Luciano/lora-8bit-Llama-2-7b-hf-lener_br
mt0-large	Luciano/financial_phrasebank_bigscience_mt0-large (cont)[c]
opt-6.7b	Luciano/CAUSAL_LM_english_quotes (cont)[d]
roberta-large	Luciano/mrpc_roberta-large_LORA
xlm-roberta-base	Luciano/lora-xlm-roberta-base-lener_br

Source: Elaborated by the author.

[a] Luciano/SEQ_2_SEQ_LM_financial_phrasebank_google_flan-t5-large_LORA
[b] Luciano/SEQ_2_SEQ_LM_samsum_philschmid_flan-t5-xxl-sharded-fp16_LORA
[c] Luciano/financial_phrasebank_bigscience_mt0-large_LORA_SEQ_2_SEQ_LM
[d] Luciano/CAUSAL_LM_english_quotes_facebook_opt-6.7b_LORA

Table 37: QLoRA fine-tuning models

Pre-trained model	Finetuned model
bloomz-560m	Luciano/qlora-bloomz-560m-lener_br
bloom-560m	Luciano/qlora-bloom-560m-lener_br
bloomz-7b1-mt	Luciano/qlora-bloomz-7b1-mt-lener_br

Source: Elaborated by the author.

Table 38: Others PEFT fine-tuning models

Pre-trained model	Method	Finetuned model
bloomz-560m	$(IA)^{3a}$	Luciano/ia3-bloomz-560m-lener_br
bloomz-560m	Prompt-tuning	Luciano/twitter_complaints_bigscience (cont)[b]
bloomz-560m	Prefix-tuning	Luciano/twitter_complaints_bigscience (cont)[c]
Llama-2-7b	Adapter	Luciano/Llama-2-7b-chat-hf-miniguanaco-adapter
Llama-2-7b	Adapter	Luciano/Llama-2-7b-chat-hf-miniguanaco_adapter
t5-large	Prefix-tuning	Luciano/financial_phrasebank_t5-large (cont)[d]
roberta-large	P-tuning[e]	Luciano/mrpc_roberta-large_P_TUNING
roberta-large	Prompt-tuning[f]	Luciano/mrpc_roberta-large_PROMPT_TUNING
roberta-large	Prefix-tuning[g]	Luciano/mrpc_roberta-large_PREFIX_TUNING

Source: Elaborated by the author.

[a](LIU et al., 2022)
[b]Luciano/twitter_complaints_bigscience_bloomz-560m_PROMPT_TUNING_CAUSAL_LM
[c]Luciano/twitter_complaints_bigscience_bloomz-560m_PREFIX_TUNING_CAUSAL_LM
[d]Luciano/financial_phrasebank_t5-large_PREFIX_TUNING_SEQ_2_SEQ_LM
[e](LIU et al., 2023b)
[f](LESTER; AL-RFOU; CONSTANT, 2021)
[g](LI; LIANG, 2021)

Table 39: Generated datasets

Dataset	Public/Private
Luciano/lener_br_text_to_lm	Public
Luciano/peticoes	Private
Luciano/peticoes_no_split	Private
Luciano/victor_lrec_2020_medium	Public
Luciano/victor_lrec_2020_small	Public

Source: Elaborated by the author.

adjustments were made to the methodology based on the lessons learned.

This section describes mainly the experiments carried out in the academic context. The exception is the experiment described in section 5.2, which is a published work that describes a real AI and Law application developed for the system for processing legal proceedings in the Court of Justice of Rio Grande do Sul. This work proposes using generative AI to automatic generate sentence reports for legal proceedings to support the work of judges in their core activity. We developed a LLM *playground* feature to prototype prompts and to support analysis of electronic court process documents and a Sentence Report Generator to automate the sentence report generation for legal cases. This work combined the academic and the real world. The academic support was important to propose and carry out the evaluation of the quality of the generated sentence reports. The result guided changes in the prompts that were being used for an improved version of them.

The article published and described in section 5.1 was the result of work that defined a new state-of-the-art for Portuguese Legal NER. However, we believe we could improve the results by adding a previous step of fine-tuning a general language model with an intradomain context-specific corpus, as described in step two of the proposed methodology.

In addition to the articles published, several other experiments not yet published were carried out, including mainly the fine-tuning of PLMs and LLMs, using the most varied techniques.

The results of experiments also showed the importance of unlabeled data in the quality of the models, since the fine-tuning did indeed adapt the context of each model to the new corpus, even though the fine-tuning was done with little data. From that, we intend to run more experiments with variations of this data, to better understand its impact and allow us to reach more concrete conclusions about its importance and indication of use.

6 CONCLUSION

"The best way to predict the future is to invent it."
Alan Kay

Artificial Intelligence (AI) has progressively gained relevance in contemporary society since the concept was first introduced over six decades ago. Today, AI is widely employed across numerous domains, serving either as a tool to augment human capabilities or as a foundational element in automating tasks traditionally performed by humans.

The legal domain began to explore AI applications shortly after the emergence of the field. In recent years, the use of AI in both the Judiciary and legal practice has become increasingly common, primarily as a means to enhance productivity and support decision-making processes. Within the Judiciary, where the number of lawsuits continues to grow and human resources cannot always expand at the same pace, AI is seen as a key enabler to maintaining service capacity and efficiency under these constraints.

Among AI subfields, Natural Language Processing (NLP) plays a particularly crucial role in legal applications due to the inherently textual nature of legal procedures. Lawsuits are predominantly composed of textual documents authored by lawyers, judges, and court officials, which makes NLP especially suitable for tasks such as document classification, information extraction, legal summarization, and decision support.

Recent advances in NLP, especially those based on deep learning and neural networks, have significantly transformed how legal texts are processed. Contextual word embeddings generated by pre-trained neural language models, particularly those based on transformer architectures with attention mechanisms, have led to state-of-the-art performance across a wide range of NLP tasks.

More recently, the advent of Large Language Models (LLMs), such as ChatGPT and other instruction-following conversational models, has opened new horizons for NLP. These decoder-only transformer models have demonstrated impressive capabilities in generative tasks, including text generation, summarization, and question answering, solidifying their role in what is now known as Generative AI.

In this context, this thesis proposes a structured methodology for the development of AI applications in the legal domain, grounded in recent advances in NLP, specifically transformer-based architectures, pre-trained language models, and transfer learning. The methodology aims to bridge the gap between cutting-edge AI research and practical development in courts and law firms.

The proposed methodology defines a high-level process that spans from data acquisition and model preparation to contextual adaptation and application-level deployment. It emphasizes collaboration between the Business Team (legal experts) and the IT Team (technical developers), acknowledging that unlike traditional software development, where the feasibility

and outcome of the solution are generally predictable, AI projects involve exploratory and iterative approaches. In these cases, the most suitable technical solution is not known a priori, and there is no guarantee it will meet the accuracy or reliability expectations required by the legal domain.

A key contribution of the methodology lies in its ability to centralize, document, and formalize best practices for building NLP-based legal AI applications. It provides practical guidance on architecture, components, datasets, evaluation metrics, and development workflows, offering legal and technical teams a reproducible and validated path toward high-quality solutions. Importantly, it also includes reusable components such as pre-trained models and curated datasets in Portuguese, tailored to the legal context.

The methodology was validated through a series of experiments and real-world implementations, including applications developed and tested at the Court of Justice of Rio Grande do Sul. These experiments confirmed the viability, coherence, and applicability of the proposed approach.

Some aspects of the methodology, particularly the optional components in the extension layer, remain open for future work. These include the integration of explainable AI mechanisms, the use of judicial ontologies to enhance semantic representation, and the development of novel application-level evaluation metrics specifically designed for legal AI systems.

Overall, this work advances the state of the art in AI and Law by providing a methodological foundation that supports the development of effective, transparent, and context-aware applications, contributing to the responsible and scalable use of AI in the legal system.

6.1 Contributions

This thesis presents several scientific and practical contributions to the field of Artificial Intelligence and Law, summarized as follows:

a) A general and structured methodology for developing state-of-the-art NLP applications in the legal domain, grounded in transformer-based architectures, pre-trained language models, and transfer learning techniques;

b) A peer-reviewed paper accepted at the International Conference on Computational Processing of Portuguese (PROPOR 2022), describing an experiment on fine-tuning a Portuguese language model for a Named Entity Recognition (NER) task involving legal entities;

c) A peer-reviewed article (currently under major revision) reporting the development of an AI system for generating judgment reports, including a domain-specific evaluation method and an application-level dataset to support the assessment of generated outputs;

d) A systematic mapping study on AI-based decision support systems for judges, identifying existing approaches, gaps, and research opportunities in the literature;

e) Two fine-tuned language models for the legal NER task in Brazilian Portuguese, which established new state-of-the-art results on the LeNER-Br dataset, along with four additional fine-tuned models for the broader judicial context, made publicly available to the research community;

f) Two additional experiments (not yet published) on fine-tuning Portuguese language models using different legal corpora to explore contextual adaptation for judicial tasks;

g) Several unpublished experiments employing Parameter-Efficient Fine-Tuning (PEFT) techniques with pre-trained language models (PLMs) and large language models (LLMs) for judicial NLP applications;

h) A legal NER application, publicly released, to demonstrate and facilitate the application of fine-tuned models in real use cases;

i) A web-based platform developed to support human evaluation of AI-generated judgment reports, also made available to the community;

j) Practical collaborations and validation activities involving judges and legal specialists from the Court of Justice of the State of Rio Grande do Sul, reinforcing the applied and real-world relevance of this work beyond academic boundaries.

6.2 Limitations

The main limitations of this work are related to technical constraints, especially those involving computational capacity for training and inference with Pretrained Language Models (PLMs) and Large Language Models (LLMs). These resource limitations restricted the ability to explore larger models, perform extensive fine-tuning, and execute more comprehensive experiments.

Additionally, some methodological advances and experiments initially planned could not be carried out due to time constraints. As a result, certain components of the proposed methodology, particularly those within the extensions layer, were not fully explored or validated. These elements remain as avenues for future work.

6.3 Future Work

Future work will focus on expanding and refining the proposed methodology, particularly by exploring the components of the extensions layer: explainable AI, judicial ontologies, and a novel application-level evaluation metric tailored to AI and Law systems.

Explainable AI, in addition to its academic relevance, is a formal requirement for AI applications within the Brazilian Judiciary, as established by Resolution No. 615/2025 of the National Council of Justice (CNJ, 2025). This resolution defines a regulatory framework for the responsible use of AI in the justice system, encompassing recent developments such as generative models. It sets forth principles of ethics, transparency, auditability, and human oversight, and explicitly requires that AI systems used in judicial contexts must provide, whenever technically feasible, clear and understandable explanations to enable human auditing and interpretation of outputs. This provision highlights the importance of explainability not only as a technical requirement, but also as a legal and ethical obligation in judicial applications of AI.

Another promising research direction is the incorporation of judicial ontologies to enhance semantic understanding and domain alignment of NLP models. Ontologies can improve data representation and support more robust, interpretable, and context-aware AI systems.

In addition, future research will focus on defining and applying a domain-specific, application-level evaluation metric that goes beyond conventional model-centric metrics such as accuracy and F1-score. This metric aims to evaluate the practical effectiveness of AI solutions within real-world legal workflows, considering preprocessing and postprocessing steps, as well as end-user expectations.

We also intend to further explore lightweight LLM adaptation strategies such as prompt engineering techniques (SAHOO et al., 2024) and Retrieval-Augmented Generation (RAG) (GAO et al., 2024), which offer lower-cost alternatives to full model training. While more computationally intensive, fine-tuning techniques—particularly those based on adapters and Parameter-Efficient Fine-Tuning (PEFT) strategies (XU et al., 2023)—will continue to be investigated as potentially viable options for tailoring models to legal contexts.

Finally, a new avenue of research will include the integration of Knowledge Graphs[1] (YANG et al., 2024) into RAG pipelines. This integration may enhance the contextual grounding and factual consistency of generative legal applications.

[1] A knowledge graph captures structured information about entities and their relationships within a specific domain, organizing data as nodes and edges in a graph format.

REFERENCES

ADADI, A.; BERRADA, M. Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI). **IEEE Access**, [S.l.], v. 6, p. 52138–52160, 2018. Conference Name: IEEE Access.

AKBIK, A.; BLYTHE, D.; VOLLGRAF, R. Contextual String Embeddings for Sequence Labeling. In: International Conference ON Computational Linguistics, 27., 2018, Santa Fe, New Mexico, USA. **Proceedings...** Association for Computational Linguistics, 2018. p. 1638–1649.

ALETRAS, N.; TSARAPATSANIS, D.; PREOţIUC-PIETRO, D.; LAMPOS, V. Predicting judicial decisions of the European Court of Human Rights: a Natural Language Processing perspective. **PeerJ Computer Science**, [S.l.], v. 2, p. e93, Oct. 2016. Publisher: PeerJ Inc.

AQUINO, I. V. d.; SANTOS, M. M. d.; DORNELES, C. F.; CARVALHO, J. T. Extracting Information from Brazilian Legal Documents with Retrieval Augmented Generation. In: SIMPóSIO Brasileiro DE Banco DE Dados (SBBD), 2024. **Anais...** SBC, 2024. p. 280–287. ISSN: 0000-0000.

ARAUJO, D. A. de; RIGO, S. J.; BARBOSA, J. L. V. Ontology-based information extraction for juridical events with case studies in Brazilian legal realm. **Artificial Intelligence and Law**, [S.l.], v. 25, n. 4, p. 379–396, Dec. 2017.

ARAUJO, P. H. Luz de; CAMPOS, T. E. de; OLIVEIRA, R. R. R. de; STAUFFER, M.; COUTO, S.; BERMEJO, P. LeNER-Br: A Dataset for Named Entity Recognition in Brazilian Legal Text. In: VILLAVICENCIO, A.; MOREIRA, V.; ABAD, A.; CASELI, H.; GAMALLO, P.; RAMISCH, C.; GONçALO OLIVEIRA, H.; PAETZOLD, G. H. (Ed.). **Computational Processing of the Portuguese Language**. Cham: Springer International Publishing, 2018. v. 11122, p. 313–323. Series Title: Lecture Notes in Computer Science.

BAHDANAU, D.; CHO, K.; BENGIO, Y. Neural Machine Translation by Jointly Learning to Align and Translate. **arXiv:1409.0473 [cs, stat]**, [S.l.], May 2016. arXiv: 1409.0473.

BAI, Y.; YING, J.; CAO, Y.; LV, X.; HE, Y.; WANG, X.; YU, J.; ZENG, K.; XIAO, Y.; LYU, H.; ZHANG, J.; LI, J.; HOU, L. Benchmarking Foundation Models with Language-Model-as-an-Examiner. **Advances in Neural Information Processing Systems**, [S.l.], v. 36, p. 78142–78167, Dec. 2023.

BAILEY, J.; BUDGEN, D.; TURNER, M.; KITCHENHAM, B.; BRERETON, P.; LINKMAN, S. Evidence relating to Object-Oriented software design: A survey. In: FIRST International Symposium ON Empirical Software Engineering AND Measurement (ESEM 2007), 2007. **Anais...** [S.l.: s.n.], 2007. p. 482–484. ISSN: 1949-3789.

BANERJEE, S.; LAVIE, A. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In: ACL Workshop ON Intrinsic AND Extrinsic Evaluation Measures FOR Machine Translation AND/OR Summarization, 2005, Ann Arbor, Michigan. **Proceedings...** Association for Computational Linguistics, 2005. p. 65–72.

BAPTISTA, I. d.; COSTA, P. R. d. O impacto da inovação no Poder Judiciário: um ensaio teórico. **Brazilian Journal of Development**, [S.l.], v. 5, n. 8, p. 12445–12465, Aug. 2019.

BARN, B.; BARAT, S.; CLARK, T. Conducting Systematic Literature Reviews and Systematic Mapping Studies. In: Innovations IN Software Engineering Conference, 10., 2017, Jaipur, India. **Proceedings...** Association for Computing Machinery, 2017. p. 212–213. (ISEC '17).

BARONI, M.; BERNARDINI, S.; FERRARESI, A.; ZANCHETTA, E. The WaCky wide web: a collection of very large linguistically processed web-crawled corpora. **Language Resources and Evaluation**, [S.l.], v. 43, n. 3, p. 209–226, Sept. 2009.

BENCH-CAPON, T.; ARASZKIEWICZ, M.; ASHLEY, K.; ATKINSON, K.; BEX, F.; BORGES, F.; BOURCIER, D.; BOURGINE, P.; CONRAD, J. G.; FRANCESCONI, E.; GORDON, T. F.; GOVERNATORI, G.; LEIDNER, J. L.; LEWIS, D. D.; LOUI, R. P.; MCCARTY, L. T.; PRAKKEN, H.; SCHILDER, F.; SCHWEIGHOFER, E.; THOMPSON, P.; TYRRELL, A.; VERHEIJ, B.; WALTON, D. N.; WYNER, A. Z. A history of AI and Law in 50 papers: 25 years of the international conference on AI and Law. **Artificial Intelligence and Law**, [S.l.], v. 20, n. 3, p. 215–319, Sept. 2012.

BENGIO, Y.; DUCHARME, R.; VINCENT, P.; JANVIN, C. A neural probabilistic language model. **The Journal of Machine Learning Research**, [S.l.], v. 3, n. null, p. 1137–1155, Mar. 2003.

BENGIO, Y.; SCHWENK, H.; SENéCAL, J.-S.; MORIN, F.; GAUVAIN, J.-L. Neural Probabilistic Language Models. In: HOLMES, D. E.; JAIN, L. C. (Ed.). **Innovations in Machine Learning**: Theory and Applications. Berlin, Heidelberg: Springer, 2006. p. 137–186. (Studies in Fuzziness and Soft Computing).

BERGER, A. L.; PIETRA, V. J. D.; PIETRA, S. A. D. A maximum entropy approach to natural language processing. **Computational Linguistics**, [S.l.], v. 22, n. 1, p. 39–71, Mar. 1996.

BERNARDINI, S.; BARONI, M.; EVERT, S. A WaCky introduction. **WaCky**, [S.l.], p. 9–40, 2006. Publisher: Citeseer.

BOJANOWSKI, P.; GRAVE, E.; JOULIN, A.; MIKOLOV, T. Enriching Word Vectors with Subword Information. **Transactions of the Association for Computational Linguistics**, [S.l.], v. 5, p. 135–146, 2017. Place: Cambridge, MA Publisher: MIT Press.

BOMMASANI, R.; HUDSON, D. A.; ADELI, E.; ALTMAN, R.; ARORA, S.; ARX, S. v.; BERNSTEIN, M. S.; BOHG, J.; BOSSELUT, A.; BRUNSKILL, E.; BRYNJOLFSSON, E.; BUCH, S.; CARD, D.; CASTELLON, R.; CHATTERJI, N.; CHEN, A.; CREEL, K.; DAVIS, J. Q.; DEMSZKY, D.; DONAHUE, C.; DOUMBOUYA, M.; DURMUS, E.; ERMON, S.; ETCHEMENDY, J.; ETHAYARAJH, K.; FEI-FEI, L.; FINN, C.; GALE, T.; GILLESPIE, L.; GOEL, K.; GOODMAN, N.; GROSSMAN, S.; GUHA, N.; HASHIMOTO, T.; HENDERSON, P.; HEWITT, J.; HO, D. E.; HONG, J.; HSU, K.; HUANG, J.; ICARD, T.; JAIN, S.; JURAFSKY, D.; KALLURI, P.; KARAMCHETI, S.; KEELING, G.; KHANI, F.; KHATTAB, O.; KOH, P. W.; KRASS, M.; KRISHNA, R.; KUDITIPUDI, R.; KUMAR, A.; LADHAK, F.; LEE, M.; LEE, T.; LESKOVEC, J.; LEVENT, I.; LI, X. L.; LI, X.; MA, T.; MALIK, A.; MANNING, C. D.; MIRCHANDANI, S.; MITCHELL, E.; MUNYIKWA, Z.; NAIR, S.; NARAYAN, A.; NARAYANAN, D.; NEWMAN, B.; NIE, A.; NIEBLES, J. C.;

NILFOROSHAN, H.; NYARKO, J.; OGUT, G.; ORR, L.; PAPADIMITRIOU, I.; PARK, J. S.; PIECH, C.; PORTELANCE, E.; POTTS, C.; RAGHUNATHAN, A.; REICH, R.; REN, H.; RONG, F.; ROOHANI, Y.; RUIZ, C.; RYAN, J.; Ré, C.; SADIGH, D.; SAGAWA, S.; SANTHANAM, K.; SHIH, A.; SRINIVASAN, K.; TAMKIN, A.; TAORI, R.; THOMAS, A. W.; TRAMèR, F.; WANG, R. E.; WANG, W.; WU, B.; WU, J.; WU, Y.; XIE, S. M.; YASUNAGA, M.; YOU, J.; ZAHARIA, M.; ZHANG, M.; ZHANG, T.; ZHANG, X.; ZHANG, Y.; ZHENG, L.; ZHOU, K.; LIANG, P. **On the Opportunities and Risks of Foundation Models**. arXiv:2108.07258.

BONIFACIO, L. H.; VILELA, P. A.; LOBATO, G. R.; FERNANDES, E. R. A Study on the Impact of Intradomain Finetuning of Deep Language Models for Legal Named Entity Recognition in Portuguese. In: INTELLIGENT Systems, 2020, Cham. **Anais...** Springer International Publishing, 2020. p. 648–662. (Lecture Notes in Computer Science).

BRANTING, L. K.; YEH, A.; WEISS, B.; MERKHOFER, E.; BROWN, B. Inducing Predictive Models for Decision Support in Administrative Adjudication. In: AI Approaches TO THE Complexity OF Legal Systems, 2018, Cham. **Anais...** Springer International Publishing, 2018. p. 465–477. (Lecture Notes in Computer Science).

BRASIL. **Lei nº 13.105, de 16 de março de 2015. Código de Processo Civil**. 2015.

BROWN, T. B.; MANN, B.; RYDER, N.; SUBBIAH, M.; KAPLAN, J.; DHARIWAL, P.; NEELAKANTAN, A.; SHYAM, P.; SASTRY, G.; ASKELL, A. Language models are few-shot learners. **arXiv preprint arXiv:2005.14165**, [S.l.], 2020.

BURCHARDT, A. Multidimensional quality metrics: a flexible system for assessing translation quality. In: Translating AND THE Computer 35, 2013, London, UK. **Proceedings...** Aslib, 2013. p. 0–0.

CARMO, D.; PIAU, M.; CAMPIOTTI, I.; NOGUEIRA, R.; LOTUFO, R. PTT5: Pretraining and validating the T5 model on Brazilian Portuguese data. **arXiv:2008.09144 [cs]**, [S.l.], Oct. 2020. arXiv: 2008.09144.

CASTRO, A. P.; CALIXTO, W. P.; GOMES, V. M.; VEIGA, E. F.; SILVA, L. F. A.; CASTRO, L. L. O. P.; BARBOSA, J. L. F.; CAMPOS, P. H. M. Ontology applied in the judicial sentences. In: CHILEAN Conference ON Electrical, Electronics Engineering, Information AND Communication Technologies (CHILECON), 2017., 2017. **Anais...** [S.l.: s.n.], 2017. p. 1–6.

CASTRO, P. V. Quinta de. **Aprendizagem profunda para reconhecimento de entidades nomeadas em domínio jurídico**. 2019. Tese (Doutorado em Ciência da Computação) — , 2019. Accepted: 2020-01-07T11:57:54Z Publisher: Universidade Federal de Goiás.

CASTRO, P. V. Quinta de; SILVA, N. Félix Felipe da; SILVA SOARES, A. da. Portuguese Named Entity Recognition Using LSTM-CRF. In: COMPUTATIONAL Processing OF THE Portuguese Language, 2018, Cham. **Anais...** Springer International Publishing, 2018. p. 83–92. (Lecture Notes in Computer Science).

CHANG, Y.-W.; LIN, C.-J. Feature Ranking Using Linear SVM. In: CAUSATION AND Prediction Challenge, 2008. **Anais...** [S.l.: s.n.], 2008. p. 53–64. ISSN: 1938-7228 Section: Machine Learning.

CHEN, B.; LI, Y.; ZHANG, S.; LIAN, H.; HE, T. A Deep Learning Method for Judicial Decision Support. In: IEEE 19TH International Conference ON Software Quality, Reliability AND Security Companion (QRS-C), 2019., 2019. **Anais...** [S.l.: s.n.], 2019. p. 145–149.

CHOWDHERY, A.; NARANG, S.; DEVLIN, J.; BOSMA, M.; MISHRA, G.; ROBERTS, A.; BARHAM, P.; CHUNG, H. W.; SUTTON, C.; GEHRMANN, S.; SCHUH, P.; SHI, K.; TSVYASHCHENKO, S.; MAYNEZ, J.; RAO, A.; BARNES, P.; TAY, Y.; SHAZEER, N.; PRABHAKARAN, V.; REIF, E.; DU, N.; HUTCHINSON, B.; POPE, R.; BRADBURY, J.; AUSTIN, J.; ISARD, M.; GUR-ARI, G.; YIN, P.; DUKE, T.; LEVSKAYA, A.; GHEMAWAT, S.; DEV, S.; MICHALEWSKI, H.; GARCIA, X.; MISRA, V.; ROBINSON, K.; FEDUS, L.; ZHOU, D.; IPPOLITO, D.; LUAN, D.; LIM, H.; ZOPH, B.; SPIRIDONOV, A.; SEPASSI, R.; DOHAN, D.; AGRAWAL, S.; OMERNICK, M.; DAI, A. M.; PILLAI, T. S.; PELLAT, M.; LEWKOWYCZ, A.; MOREIRA, E.; CHILD, R.; POLOZOV, O.; LEE, K.; ZHOU, Z.; WANG, X.; SAETA, B.; DIAZ, M.; FIRAT, O.; CATASTA, M.; WEI, J.; MEIER-HELLSTERN, K.; ECK, D.; DEAN, J.; PETROV, S.; FIEDEL, N. PaLM: scaling language modeling with pathways. **The Journal of Machine Learning Research**, [S.l.], v. 24, n. 1, p. 240:11324–240:11436, Mar. 2024.

CHOWDHURY, G. G. Natural language processing. **Annual Review of Information Science and Technology**, [S.l.], v. 37, n. 1, p. 51–89, 2003. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/aris.1440370103.

CIURLINO, V. H. **BertBR** : a pretrained language model for law texts. 2021. Tese (Doutorado em Ciência da Computação) — Trabalho de Conclusão de Curso (Bacharelado em Engenharia Eletrônica). Universidade de Brasília, Brasília, 2021. Accepted: 2021-06-25T14:37:56Z.

CLARK, P.; COWHEY, I.; ETZIONI, O.; KHOT, T.; SABHARWAL, A.; SCHOENICK, C.; TAFJORD, O. **Think you have Solved Question Answering? Try ARC, the AI2 Reasoning Challenge**. arXiv:1803.05457.

CNJ. **Resolução Nº 215 de 16/12/2015**. 2015.

CNJ. **CNJ abre seleção de projetos para Centro de Inteligência Artificial**. 2019.

CNJ. **Resolução Nº 332 de 21/08/2020**. 2020.

CNJ. **Justiça em números 2021**. Brasília: Conselho Nacional de Justiça, 2021.

CNJ. **Em 15 anos, Justiça recebeu mais de 250 milhões de processos eletrônicos**. 2024.

CNJ. **DataJud - Base Nacional de Dados do Poder Judiciário**. 2024.

CNJ. **Resolução Nº 615 de 11/03/2025**. 2025.

COLLOBERT, R.; WESTON, J.; BOTTOU, L.; KARLEN, M.; KAVUKCUOGLU, K.; KUKSA, P. Natural Language Processing (Almost) from Scratch. **The Journal of Machine Learning Research**, [S.l.], v. 12, n. null, p. 2493–2537, Nov. 2011.

COLLOVINI, S.; SANTOS, J.; CONSOLI, B.; TERRA, J.; VIEIRA, R.; QUARESMA, P.; SOUZA, M.; CLARO, D. B.; GLAUBER, R. a Xavier. **CC: Portuguese named entity recognition and relation extraction tasks at iberlef**, [S.l.], 2019.

CORTES, C.; VAPNIK, V. Support-vector networks. **Machine Learning**, [S.l.], v. 20, n. 3, p. 273–297, Sept. 1995.

CORTES, E. G.; VIANNA, A. L.; MARTINS, M.; RIGO, S.; KUNST, R. LLMs and Translation: different approaches to localization between Brazilian Portuguese and European Portuguese. In: International Conference ON Computational Processing OF Portuguese - Vol. 1, 16., 2024, Santiago de Compostela, Galicia/Spain. **Proceedings...** Association for Computational Lingustics, 2024. p. 45–55.

DAL PONT, T. R.; SABO, I. C.; HüBNER, J. F.; ROVER, A. J. Impact of Text Specificity and Size on Word Embeddings Performance: An Empirical Evaluation in Brazilian Legal Domain. In: INTELLIGENT Systems: 9TH Brazilian Conference, BRACIS 2020, Rio Grande, Brazil, October 20–23, 2020, Proceedings, Part I, 2020, Berlin, Heidelberg. **Anais...** Springer-Verlag, 2020. p. 521–535.

DAS, T.; ROY, A.; MAJUMDAR, A. K. A Study on Legal Knowledge Base Creation Using Artificial Intelligence and Ontology. In: INNOVATIVE Data Communication Technologies AND Application, 2020, Cham. **Anais...** Springer International Publishing, 2020. p. 647–653. (Lecture Notes on Data Engineering and Communications Technologies).

DEROY, A.; GHOSH, K.; GHOSH, S. **How Ready are Pre-trained Abstractive Models and LLMs for Legal Case Judgement Summarization?** arXiv:2306.01248.

DEROY, A.; GHOSH, K.; GHOSH, S. Applicability of large language models and generative models for legal case judgement summarization. **Artificial Intelligence and Law**, [S.l.], July 2024.

DETTMERS, T.; PAGNONI, A.; HOLTZMAN, A.; ZETTLEMOYER, L. **QLoRA**: Efficient Finetuning of Quantized LLMs. arXiv:2305.14314.

DEVLIN, J.; CHANG, M.-W.; LEE, K.; TOUTANOVA, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In: Conference OF THE North American Chapter OF THE Association FOR Computational Linguistics: Human Language Technologies, Volume 1 (Long AND Short Papers), 2019., 2019, Minneapolis, Minnesota. **Proceedings...** Association for Computational Linguistics, 2019. p. 4171–4186.

DICE, D.; KOGAN, A. Optimizing Inference Performance of Transformers on CPUs. **arXiv:2102.06621 [cs]**, [S.l.], Feb. 2021. arXiv: 2102.06621.

DODGE, J.; ILHARCO, G.; SCHWARTZ, R.; FARHADI, A.; HAJISHIRZI, H.; SMITH, N. Fine-Tuning Pretrained Language Models: Weight Initializations, Data Orders, and Early Stopping. **arXiv:2002.06305 [cs]**, [S.l.], Feb. 2020. arXiv: 2002.06305.

DOZIER, C.; KONDADADI, R.; LIGHT, M.; VACHHER, A.; VEERAMACHANENI, S.; WUDALI, R. Named Entity Recognition and Resolution in Legal Text. In: FRANCESCONI, E.; MONTEMAGNI, S.; PETERS, W.; TISCORNIA, D. (Ed.). **Semantic Processing of Legal Texts**: Where the Language of Law Meets the Law of Language. Berlin, Heidelberg: Springer, 2010. p. 27–43. (Lecture Notes in Computer Science).

EL GHOSH, M.; NAJA, H.; ABDULRAB, H.; KHALIL, M. Towards a Legal Rule-Based System Grounded on the Integration of Criminal Domain Ontology and Rules. **Knowledge-Based and Intelligent Information & Engineering Systems: Proceedings of the 21st International Conference, KES-20176-8 September 2017, Marseille, France**, [S.l.], v. 112, p. 632–642, Jan. 2017.

FANG, X.; ZHAO, X. Nonlinear Dimensionality Reduction with Judicial Document Learning. In: IEEE International Conference ON Big Knowledge (ICBK), 2018., 2018. **Anais...** [S.l.: s.n.], 2018. p. 448–455.

FARUQUI, M.; TSVETKOV, Y.; RASTOGI, P.; DYER, C. Problems With Evaluation of Word Embeddings Using Word Similarity Tasks. **arXiv:1605.02276 [cs]**, [S.l.], June 2016. arXiv: 1605.02276.

FAWEI, B.; WYNER, A.; PAN, J. Z.; KOLLINGBAUM, M. Using Legal Ontologies with Rules for Legal Textual Entailment. In: AI Approaches TO THE Complexity OF Legal Systems, 2018, Cham. **Anais...** Springer International Publishing, 2018. p. 317–324. (Lecture Notes in Computer Science).

FEIJO, D.; MOREIRA, V. Summarizing Legal Rulings: Comparative Experiments. In: International Conference ON Recent Advances IN Natural Language Processing (RANLP 2019), 2019, Varna, Bulgaria. **Proceedings...** INCOMA Ltd., 2019. p. 313–322.

FONSECA, E.; SANTOS, L.; CRISCUOLO, M.; ALUISIO, S. ASSIN: Avaliacao de similaridade semantica e inferencia textual. In: COMPUTATIONAL Processing OF THE Portuguese Language-12TH International Conference, Tomar, Portugal, 2016. **Anais...** [S.l.: s.n.], 2016. p. 13–15.

FREITAS, C.; MOTA, C.; SANTOS, D.; OLIVEIRA, H. G.; CARVALHO, P. Second HAREM: Advancing the State of the Art of Named Entity Recognition in Portuguese. In: Seventh International Conference ON Language Resources AND Evaluation (LREC'10), 2010, Valletta, Malta. **Proceedings...** European Language Resources Association (ELRA), 2010.

GALASSI, A.; LIPPI, M.; TORRONI, P. Attention in Natural Language Processing. **IEEE Transactions on Neural Networks and Learning Systems**, [S.l.], v. 32, n. 10, p. 4291–4308, Oct. 2021. Conference Name: IEEE Transactions on Neural Networks and Learning Systems.

GAO, Y.; XIONG, Y.; GAO, X.; JIA, K.; PAN, J.; BI, Y.; DAI, Y.; SUN, J.; WANG, M.; WANG, H. **Retrieval-Augmented Generation for Large Language Models**: A Survey. arXiv:2312.10997.

GARG, S.; VU, T.; MOSCHITTI, A. TANDA: Transfer and Adapt Pre-Trained Transformer Models for Answer Sentence Selection. **Proceedings of the AAAI Conference on Artificial Intelligence**, [S.l.], v. 34, n. 05, p. 7780–7788, Apr. 2020. Number: 05.

GEHRING, J.; AULI, M.; GRANGIER, D.; YARATS, D.; DAUPHIN, Y. N. Convolutional Sequence to Sequence Learning. In: International Conference ON Machine Learning, 34., 2017. **Proceedings...** PMLR, 2017. p. 1243–1252. ISSN: 2640-3498.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. **Deep Learning**. [S.l.]: MIT Press, 2016.

GUILLOU, P. **GPorTuguese-2**: a Language Model for Portuguese text generation (and more NLP tasks...). 2020.

GUNNING, D.; STEFIK, M.; CHOI, J.; MILLER, T.; STUMPF, S.; YANG, G.-Z. XAI—Explainable artificial intelligence. **Science Robotics**, [S.l.], v. 4, n. 37, p. eaay7120, Dec. 2019. Publisher: American Association for the Advancement of Science.

HARTMANN, N.; FONSECA, E.; SHULBY, C.; TREVISO, M.; RODRIGUES, J.; ALUISIO, S. Portuguese word embeddings: Evaluating on word analogies and natural language tasks. **arXiv preprint arXiv:1708.06025**, [S.l.], 2017.

HE, T.-K.; LIAN, H.; QIN, Z.-M.; CHEN, Z.-Y.; LUO, B. PTM: A Topic Model for the Inferring of the Penalty. **Journal of Computer Science and Technology**, [S.l.], v. 33, n. 4, p. 756–767, July 2018.

HE, Y.; CHEN, J.; ANTONYRAJAH, D.; HORROCKS, I. Biomedical ontology alignment with BERT. In: International Workshop ON Ontology Matching, 16., 2021, Virtual conference, October. **Proceedings...** CEUR, 2021. p. 1–12. (CEUR Workshop Proceedings, v. 3063). ISSN: 1613-0073.

HE, Y.; CHEN, J.; ANTONYRAJAH, D.; HORROCKS, I. BERTMap: A BERT-based Ontology Alignment System. **arXiv:2112.02682 [cs]**, [S.l.], Jan. 2022. arXiv: 2112.02682.

HENDRYCKS, D.; BURNS, C.; BASART, S.; ZOU, A.; MAZEIKA, M.; SONG, D.; STEINHARDT, J. **Measuring Massive Multitask Language Understanding**. arXiv:2009.03300.

HOEKSTRA, R.; BREUKER, J.; DI BELLO, M.; BOER, A. The LKIF Core Ontology of Basic Legal Concepts. **LOAIT**, [S.l.], v. 321, p. 43–63, 2007.

HOOVER, B.; STROBELT, H.; GEHRMANN, S. exBERT: A Visual Analysis Tool to Explore Learned Representations in Transformer Models. In: Annual Meeting OF THE Association FOR Computational Linguistics: System Demonstrations, 58., 2020, Online. **Proceedings...** Association for Computational Linguistics, 2020. p. 187–196.

HOWARD, J.; RUDER, S. Universal Language Model Fine-tuning for Text Classification. In: Annual Meeting OF THE Association FOR Computational Linguistics (Volume 1: Long Papers), 56., 2018, Melbourne, Australia. **Proceedings...** Association for Computational Linguistics, 2018. p. 328–339.

HSU, Y.-T.; GARG, S.; LIAO, Y.-H.; CHATSVIORKIN, I. Efficient Inference For Neural Machine Translation. In: SustaiNLP: Workshop ON Simple AND Efficient Natural Language Processing, 2020, Online. **Proceedings...** Association for Computational Linguistics, 2020. p. 48–53.

HU, E. J.; SHEN, Y.; WALLIS, P.; ALLEN-ZHU, Z.; LI, Y.; WANG, S.; WANG, L.; CHEN, W. **LoRA**: Low-Rank Adaptation of Large Language Models. arXiv:2106.09685.

HU, Z.; LI, X.; TU, C.; LIU, Z.; SUN, M. Few-Shot Charge Prediction with Discriminative Legal Attributes. In: International Conference ON Computational Linguistics, 27., 2018, Santa Fe, New Mexico, USA. **Proceedings...** Association for Computational Linguistics, 2018. p. 487–498.

HUANG, H.; QU, Y.; LIU, J.; YANG, M.; ZHAO, T. **An Empirical Study of LLM-as-a-Judge for LLM Evaluation**: Fine-tuned Judge Models are Task-specific Classifiers. arXiv:2403.02839 [cs].

HUANG, S.-J.; ZHOU, Z.-H. Multi-label learning by exploiting label correlations locally. In: Twenty-Sixth AAAI Conference ON Artificial Intelligence, 2012, Toronto, Ontario, Canada. **Proceedings...** AAAI Press, 2012. p. 949–955. (AAAI'12).

HUGGING Face. 2022.

INOVAçãO., R. G. do Sul. Tribunal de Justiça. Comissão de. **Guia de linguagem simples TJRS**. Porto Alegre: Tribunal de Justiça do Estado do Rio Grande do Sul, Departamento de Suporte Operacional, Núcleo de Arte e Controle de Cópias, 2021.

IYYER, M.; WIETING, J.; GIMPEL, K.; ZETTLEMOYER, L. Adversarial Example Generation with Syntactically Controlled Paraphrase Networks. In: Conference OF THE North American Chapter OF THE Association FOR Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), 2018., 2018, New Orleans, Louisiana. **Proceedings...** Association for Computational Linguistics, 2018. p. 1875–1885.

JIAJING, L.; XIAOMING, L.; TAO, M. TML: A General High-Performance Text Mining Language. **Jisuanji Yanjiu yu Fazhan**, [S.l.], v. 52, n. 3, p. 553–560, 2015.

JIANG, A. Q.; SABLAYROLLES, A.; ROUX, A.; MENSCH, A.; SAVARY, B.; BAMFORD, C.; CHAPLOT, D. S.; CASAS, D. d. l.; HANNA, E. B.; BRESSAND, F.; LENGYEL, G.; BOUR, G.; LAMPLE, G.; LAVAUD, L. R.; SAULNIER, L.; LACHAUX, M.-A.; STOCK, P.; SUBRAMANIAN, S.; YANG, S.; ANTONIAK, S.; SCAO, T. L.; GERVET, T.; LAVRIL, T.; WANG, T.; LACROIX, T.; SAYED, W. E. **Mixtral of Experts**. arXiv:2401.04088.

JOULIN, A.; GRAVE, E.; BOJANOWSKI, P.; MIKOLOV, T. Bag of Tricks for Efficient Text Classification. In: Conference OF THE European Chapter OF THE Association FOR Computational Linguistics: Volume 2, Short Papers, 15., 2017, Valencia, Spain. **Proceedings...** Association for Computational Linguistics, 2017. p. 427–431.

JUNYI, S. **Jieba Chinese word segmentation tool**. original-date: 2012-09-29T07:52:01Z.

KALCHBRENNER, N.; GREFENSTETTE, E.; BLUNSOM, P. A Convolutional Neural Network for Modelling Sentences. In: Annual Meeting OF THE Association FOR Computational Linguistics (Volume 1: Long Papers), 52., 2014, Baltimore, Maryland. **Proceedings...** Association for Computational Linguistics, 2014. p. 655–665.

KATZ, D. M.; II, M. J. B.; BLACKMAN, J. A general approach for predicting the behavior of the Supreme Court of the United States. **PLOS ONE**, [S.l.], v. 12, n. 4, p. e0174698, Apr. 2017. Publisher: Public Library of Science.

KAUFMAN, A. R.; KRAFT, P.; SEN, M. Machine Learning , Text Data , and Supreme Court Forecasting. In: ., 2017. **Anais...** [S.l.: s.n.], 2017.

KIM, Y. Convolutional Neural Networks for Sentence Classification. In: Conference ON Empirical Methods IN Natural Language Processing (EMNLP), 2014., 2014, Doha, Qatar. **Proceedings...** Association for Computational Linguistics, 2014. p. 1746–1751.

KIM, Y.; JERNITE, Y.; SONTAG, D.; RUSH, A. M. Character-aware neural language models. In: Thirtieth AAAI Conference ON Artificial Intelligence, 2016, Phoenix, Arizona. **Proceedings...** AAAI Press, 2016. p. 2741–2749. (AAAI'16).

KITCHENHAM, B.; CHARTERS, S. **Guidelines for performing Systematic Literature Reviews in software engineering. EBSE Technical Report EBSE-2007-01**. [S.l.]: Keele University and University of Durham, 2007.

KLüGL, P.; TOEPFER, M.; BECK, P.-D.; FETTE, G.; PUPPE, F. UIMA Ruta: Rapid development of rule-based information extraction applications. **Natural Language Engineering**, [S.l.], v. FirstView, p. 1–40, Oct. 2014.

KOJIMA, T.; GU, S. S.; REID, M.; MATSUO, Y.; IWASAWA, Y. **Large Language Models are Zero-Shot Reasoners**. arXiv:2205.11916 [cs].

KUDO, T. Subword Regularization: Improving Neural Network Translation Models with Multiple Subword Candidates. In: Annual Meeting OF THE Association FOR Computational Linguistics (Volume 1: Long Papers), 56., 2018, Melbourne, Australia. **Proceedings...** Association for Computational Linguistics, 2018. p. 66–75.

KURCHEEVA, G.; RAKHVALOVA, M.; RAKHVALOVA, D.; BAKAEV, M. Mining and Indexing of Legal Natural Language Texts with Domain and Task Ontology. In: ELECTRONIC Governance AND Open Society: Challenges IN Eurasia, 2019, Cham. **Anais...** Springer International Publishing, 2019. p. 123–137. (Communications in Computer and Information Science).

KURSA, M. B.; RUDNICKI, W. R. Feature Selection with the Boruta Package. **Journal of Statistical Software**, [S.l.], v. 36, n. 1, p. 1–13, Sept. 2010. Number: 1.

LABAN, P.; SCHNABEL, T.; BENNETT, P. N.; HEARST, M. A. SummaC: Re-Visiting NLI-based Models for Inconsistency Detection in Summarization. **Transactions of the Association for Computational Linguistics**, [S.l.], v. 10, p. 163–177, Feb. 2022.

LAROCHELLE, H.; HINTON, G. E. Learning to combine foveal glimpses with a third-order Boltzmann machine. In: ADVANCES IN Neural Information Processing Systems, 2010. **Anais...** Curran Associates: Inc., 2010. v. 23.

Lawgorithm. **A evolução da Inteligência Artificial aplicada ao Direito no Brasil**. 2019.

LAWLOR, R. C. What Computers Can Do: Analysis and Prediction of Judicial Decisions. **American Bar Association Journal**, [S.l.], v. 49, n. 4, p. 337–344, 1963. Publisher: American Bar Association.

LEI, M.; GE, J.; LI, Z.; LI, C.; ZHOU, Y.; ZHOU, X.; LUO, B. Automatically Classify Chinese Judgment Documents Utilizing Machine Learning Algorithms. In: DATABASE Systems FOR Advanced Applications, 2017, Cham. **Anais...** Springer International Publishing, 2017. p. 3–17. (Lecture Notes in Computer Science).

LEITER, B. Legal formalism and legal realism: what is the issue. **Legal Theory**, [S.l.], v. 16, p. 111, 2010.

LESTER, B.; AL-RFOU, R.; CONSTANT, N. The Power of Scale for Parameter-Efficient Prompt Tuning. In: Conference ON Empirical Methods IN Natural Language Processing, 2021., 2021, Online and Punta Cana, Dominican Republic. **Proceedings...** Association for Computational Linguistics, 2021. p. 3045–3059.

LEWIS, M.; LIU, Y.; GOYAL, N.; GHAZVININEJAD, M.; MOHAMED, A.; LEVY, O.; STOYANOV, V.; ZETTLEMOYER, L. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. **arXiv:1910.13461 [cs, stat]**, [S.l.], Oct. 2019. arXiv: 1910.13461.

LI, D.; ZHANG, Y.; PENG, H.; CHEN, L.; BROCKETT, C.; SUN, M.-T.; DOLAN, B. Contextualized Perturbation for Textual Adversarial Attack. In: Conference OF THE North American Chapter OF THE Association FOR Computational Linguistics: Human Language Technologies, 2021., 2021, Online. **Proceedings...** Association for Computational Linguistics, 2021. p. 5053–5069.

LI, J.; ZHANG, G.; YU, L.; MENG, T. Research and Design on Cognitive Computing Framework for Predicting Judicial Decisions. **Journal of Signal Processing Systems**, [S.l.], v. 91, n. 10, p. 1159–1167, Oct. 2019.

LI, X. L.; LIANG, P. Prefix-Tuning: Optimizing Continuous Prompts for Generation. In: Annual Meeting OF THE Association FOR Computational Linguistics AND THE 11TH International Joint Conference ON Natural Language Processing (Volume 1: Long Papers), 59., 2021, Online. **Proceedings...** Association for Computational Linguistics, 2021. p. 4582–4597.

LI, X.; ZHANG, T.; DUBOIS, Y.; TAORI, R.; GULRAJANI, I.; GUESTRIN, C.; LIANG, P.; HASHIMOTO, T. B. **AlpacaEval**: An Automatic Evaluator of Instruction-following Models. original-date: 2023-05-25T09:35:28Z.

LI, Z.; DING, X.; LIU, T. Story Ending Prediction by Transferable BERT. In: Twenty-Eighth International Joint Conference ON Artificial Intelligence, 2019. **Proceedings...** [S.l.: s.n.], 2019. p. 1800–1806.

LIN, C.-H.; CHENG, P.-J. **Legal Documents Drafting with Fine-Tuned Pre-Trained Large Language Model**. arXiv:2406.04202.

LIN, C.-Y. ROUGE: A Package for Automatic Evaluation of Summaries. In: TEXT Summarization Branches Out, 2004, Barcelona, Spain. **Anais...** Association for Computational Linguistics, 2004. p. 74–81.

LING, W.; DYER, C.; BLACK, A. W.; TRANCOSO, I. Two/Too Simple Adaptations of Word2Vec for Syntax Problems. In: Conference OF THE North American Chapter OF THE Association FOR Computational Linguistics: Human Language Technologies, 2015., 2015, Denver, Colorado. **Proceedings...** Association for Computational Linguistics, 2015. p. 1299–1304.

LINS, A. A.; CARVALHO, C. S.; BOMFIM, F. D. C. J.; CARVALHO BENTES, D. de; PINHEIRO, V. CLSJUR.BR - A Model for Abstractive Summarization of Legal Documents in Portuguese Language based on Contrastive Learning. In: International Conference ON Computational Processing OF Portuguese - Vol. 1, 16., 2024, Santiago de Compostela, Galicia/Spain. **Proceedings...** Association for Computational Lingustics, 2024. p. 321–331.

LIU, C.-L.; HSIEH, C.-D. Exploring Phrase-Based Classification of Judicial Documents for Criminal Charges in Chinese. In: FOUNDATIONS OF Intelligent Systems, 2006, Berlin, Heidelberg. **Anais...** Springer, 2006. p. 681–690. (Lecture Notes in Computer Science).

LIU, H.; PERL, Y.; GELLER, J. Concept placement using BERT trained by transforming and summarizing biomedical ontology structure. **Journal of Biomedical Informatics**, [S.l.], v. 112, p. 103607, Dec. 2020.

LIU, H.; TAM, D.; MUQEETH, M.; MOHTA, J.; HUANG, T.; BANSAL, M.; RAFFEL, C. **Few-Shot Parameter-Efficient Fine-Tuning is Better and Cheaper than In-Context Learning**. arXiv:2205.05638.

LIU, P.; QIU, X.; HUANG, X. Recurrent neural network for text classification with multi-task learning. In: Twenty-Fifth International Joint Conference ON Artificial Intelligence, 2016, New York, New York, USA. **Proceedings...** AAAI Press, 2016. p. 2873–2879. (IJCAI'16).

LIU, P.; YUAN, W.; FU, J.; JIANG, Z.; HAYASHI, H.; NEUBIG, G. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. **ACM Computing Surveys**, [S.l.], v. 55, n. 9, p. 195:1–195:35, Jan. 2023.

LIU, Q.; KUSNER, M. J.; BLUNSOM, P. A Survey on Contextual Embeddings. **arXiv:2003.07278 [cs]**, [S.l.], Apr. 2020. arXiv: 2003.07278.

LIU, X.; ZHENG, Y.; DU, Z.; DING, M.; QIAN, Y.; YANG, Z.; TANG, J. **GPT Understands, Too**. arXiv:2103.10385.

LIU, Y.; LIU, P. SimCLS: A Simple Framework for Contrastive Learning of Abstractive Summarization. In: Annual Meeting OF THE Association FOR Computational Linguistics AND THE 11TH International Joint Conference ON Natural Language Processing (Volume 2: Short Papers), 59., 2021, Online. **Proceedings...** Association for Computational Linguistics, 2021. p. 1065–1072.

LIU, Z.; CHEN, H. A predictive performance comparison of machine learning models for judicial cases. In: IEEE Symposium Series ON Computational Intelligence (SSCI), 2017., 2017. **Anais...** [S.l.: s.n.], 2017. p. 1–6.

LUNDBERG, S. M.; LEE, S.-I. A Unified Approach to Interpreting Model Predictions. In: GUYON, I.; LUXBURG, U. V.; BENGIO, S.; WALLACH, H.; FERGUS, R.; VISHWANATHAN, S.; GARNETT, R. (Ed.). **Advances in Neural Information Processing Systems 30**. [S.l.]: Curran Associates, Inc., 2017. p. 4765–4774.

LUO, B.; FENG, Y.; XU, J.; ZHANG, X.; ZHAO, D. Learning to Predict Charges for Criminal Cases with Legal Basis. In: Conference ON Empirical Methods IN Natural Language Processing, 2017., 2017, Copenhagen, Denmark. **Proceedings...** Association for Computational Linguistics, 2017. p. 2727–2736.

MA, X.; HOVY, E. End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF. In: Annual Meeting OF THE Association FOR Computational Linguistics (Volume 1: Long Papers), 54., 2016, Berlin, Germany. **Proceedings...** Association for Computational Linguistics, 2016. p. 1064–1074.

MADAAN, N.; PADHI, I.; PANWAR, N.; SAHA, D. Generate your counterfactuals: Towards controlled counterfactual generation for text. In: AAAI Conference ON Artificial Intelligence, 2021. **Proceedings...** [S.l.: s.n.], 2021. v. 35, p. 13516–13524. Issue: 15.

MAHFOUZ, T.; KANDIL, A.; DAVLYATOV, S. Identification of latent legal knowledge in differing site condition (DSC) litigations. **Automation in Construction**, [S.l.], v. 94, p. 104–111, Oct. 2018.

MARANGUNIć, N.; GRANIć, A. Technology acceptance model: a literature review from 1986 to 2013. **Universal Access in the Information Society**, [S.l.], v. 14, n. 1, p. 81–95, Mar. 2015.

MARCHEGGIANI, D.; BASTINGS, J.; TITOV, I. Exploiting Semantics in Neural Machine Translation with Graph Convolutional Networks. In: Conference OF THE North American Chapter OF THE Association FOR Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers), 2018., 2018, New Orleans, Louisiana. **Proceedings...** Association for Computational Linguistics, 2018. p. 486–492.

MARIANA, V. R. **The Multidimensional Quality Metric (MQM) framework**: A new framework for translation quality assessment. [S.l.]: Brigham Young University, 2014.

MARQUES, M. R. S.; BIANCO, T.; ROODNEJAD, M.; BADUEL, T.; BERROU, C. Machine learning for explaining and ranking the most influential matters of law. In: Seventeenth International Conference ON Artificial Intelligence AND Law, 2019, Montreal, QC, Canada. **Proceedings...** Association for Computing Machinery, 2019. p. 239–243. (ICAIL '19).

MARR, B. **How AI And Machine Learning Are Transforming Law Firms And The Legal Sector**. 2018.

MARSHALL, C.; BRERETON, P. Tools to Support Systematic Literature Reviews in Software Engineering: A Mapping Study. In: ACM / IEEE International Symposium ON Empirical Software Engineering AND Measurement, 2013., 2013. **Anais...** [S.l.: s.n.], 2013. p. 296–299. ISSN: 1949-3789.

MARTINEZ, A. R. Natural language processing. **WIREs Computational Statistics**, [S.l.], v. 2, n. 3, p. 352–357, 2010. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/wics.76.

MATHUR, N.; BALDWIN, T.; COHN, T. Tangled up in BLEU: Reevaluating the Evaluation of Automatic Machine Translation Evaluation Metrics. In: Annual Meeting OF THE Association FOR Computational Linguistics, 58., 2020, Online. **Proceedings...** Association for Computational Linguistics, 2020. p. 4984–4997.

MEDVEDEVA, M.; VOLS, M.; WIELING, M. Using machine learning to predict decisions of the European Court of Human Rights. **Artificial Intelligence and Law**, [S.l.], June 2019.

MENDONçA JúNIOR, C.; MACEDO, H.; BISPO, T.; SANTOS, F.; SILVA, N.; BARBOSA, L. Paramopama: a Brazilian-Portuguese corpus for named entity recognition. In: XII Encontro Nacional DE Inteligência Artificial E Computacional (ENIAC), 2015. **Anais...** [S.l.: s.n.], 2015.

MENEZES, D. S.; SAVARESE, P.; MILIDIú, R. L. Building a Massive Corpus for Named Entity Recognition using Free Open Data Sources. **arXiv:1908.05758 [cs]**, [S.l.], Aug. 2019. arXiv: 1908.05758.

METSKER, O.; TROFIMOV, E.; GRECHISHCHEVA, S. Natural Language Processing of Russian Court Decisions for Digital Indicators Mapping for Oversight Process Control Efficiency: Disobeying a Police Officer Case. In: ELECTRONIC Governance AND Open Society: Challenges IN Eurasia, 2020, Cham. **Anais...** Springer International Publishing, 2020. p. 295–307. (Communications in Computer and Information Science).

MIKOLOV, T.; CHEN, K.; CORRADO, G.; DEAN, J. Efficient Estimation of Word Representations in Vector Space. **arXiv:1301.3781 [cs]**, [S.l.], Sept. 2013. arXiv: 1301.3781.

MOHIT, B. Named Entity Recognition. In: ZITOUNI, I. (Ed.). **Natural Language Processing of Semitic Languages**. Berlin, Heidelberg: Springer, 2014. p. 221–245. (Theory and Applications of Natural Language Processing).

MOK, W. Y.; MOK, J. R. Classification of Breach of Contract Court Decision Sentences. In: Third Workshop ON Automated Semantic Analysis OF Information IN Legal Texts, 2019, Montreal, QC. **Proceedings...** CEUR, 2019. (CEUR Workshop Proceedings, v. 2385). ISSN: 1613-0073.

MUNKHDALAI, T.; FARUQUI, M.; GOPAL, S. **Leave No Context Behind**: Efficient Infinite Context Transformers with Infini-attention. arXiv:2404.07143.

NAGPAL, A.; GABRANI, G. Python for Data Analytics, Scientific and Technical Applications. In: Amity International Conference ON Artificial Intelligence (AICAI), 2019., 2019. **Anais...** [S.l.: s.n.], 2019. p. 140–145.

NEUTEL, S.; BOER, M. H. de. Towards Automatic Ontology Alignment using BERT. In: AAAI Spring Symposium: Combining Machine Learning WITH Knowledge Engineering, 2021. **Anais...** [S.l.: s.n.], 2021.

NOTHMAN, J.; RINGLAND, N.; RADFORD, W.; MURPHY, T.; CURRAN, J. R. Learning multilingual named entity recognition from Wikipedia. **Artificial Intelligence**, [S.l.], v. 194, p. 151–175, Jan. 2013.

NOY, N. F.; MCGUINNESS, D. L. **Ontology Development 101**: A Guide to Creating Your First Ontology. [S.l.]: Stanford Knowledge Systems Laboratory, 2001. Technical Report.

OLIVEIRA, L. E. S. e.; PETERS, A. C.; SILVA, A. M. P. da; GEBELUCA, C. P.; GUMIEL, Y. B.; CINTHO, L. M. M.; CARVALHO, D. R.; AL HASAN, S.; MORO, C. M. C. SemClinBr - a multi-institutional and multi-specialty semantically annotated corpus for Portuguese clinical NLP tasks. **Journal of Biomedical Semantics**, [S.l.], v. 13, n. 1, p. 13, May 2022.

OPENAI. **openai/evals**. original-date: 2023-01-23T20:51:04Z.

OPENAI; ACHIAM, J.; ADLER, S.; AGARWAL, S.; AHMAD, L.; AKKAYA, I.; ALEMAN, F. L.; ALMEIDA, D.; ALTENSCHMIDT, J.; ALTMAN, S.; ANADKAT, S.; AVILA, R.; BABUSCHKIN, I.; BALAJI, S.; BALCOM, V.; BALTESCU, P.; BAO, H.; BAVARIAN, M.; BELGUM, J.; BELLO, I.; BERDINE, J.; BERNADETT-SHAPIRO, G.; BERNER, C.; BOGDONOFF, L.; BOIKO, O.; BOYD, M.; BRAKMAN, A.-L.; BROCKMAN, G.; BROOKS, T.; BRUNDAGE, M.; BUTTON, K.; CAI, T.; CAMPBELL, R.; CANN, A.; CAREY, B.; CARLSON, C.; CARMICHAEL, R.; CHAN, B.; CHANG, C.; CHANTZIS, F.; CHEN, D.; CHEN, S.; CHEN, R.; CHEN, J.; CHEN, M.; CHESS, B.; CHO, C.; CHU, C.; CHUNG, H. W.; CUMMINGS, D.; CURRIER, J.; DAI, Y.; DECAREAUX, C.; DEGRY, T.; DEUTSCH, N.; DEVILLE, D.; DHAR, A.; DOHAN, D.; DOWLING, S.; DUNNING, S.; ECOFFET, A.; ELETI, A.; ELOUNDOU, T.; FARHI, D.; FEDUS, L.; FELIX, N.; FISHMAN, S. P.; FORTE, J.; FULFORD, I.; GAO, L.; GEORGES, E.; GIBSON, C.; GOEL, V.; GOGINENI, T.; GOH, G.; GONTIJO-LOPES, R.; GORDON, J.; GRAFSTEIN, M.; GRAY, S.; GREENE, R.; GROSS, J.; GU, S. S.; GUO, Y.; HALLACY, C.; HAN, J.; HARRIS, J.; HE, Y.; HEATON, M.; HEIDECKE, J.; HESSE, C.; HICKEY, A.; HICKEY, W.; HOESCHELE, P.;

HOUGHTON, B.; HSU, K.; HU, S.; HU, X.; HUIZINGA, J.; JAIN, S.; JAIN, S.; JANG, J.; JIANG, A.; JIANG, R.; JIN, H.; JIN, D.; JOMOTO, S.; JONN, B.; JUN, H.; KAFTAN, T.; KAISER, L.; KAMALI, A.; KANITSCHEIDER, I.; KESKAR, N. S.; KHAN, T.; KILPATRICK, L.; KIM, J. W.; KIM, C.; KIM, Y.; KIRCHNER, J. H.; KIROS, J.; KNIGHT, M.; KOKOTAJLO, D.; KONDRACIUK, L.; KONDRICH, A.; KONSTANTINIDIS, A.; KOSIC, K.; KRUEGER, G.; KUO, V.; LAMPE, M.; LAN, I.; LEE, T.; LEIKE, J.; LEUNG, J.; LEVY, D.; LI, C. M.; LIM, R.; LIN, M.; LIN, S.; LITWIN, M.; LOPEZ, T.; LOWE, R.; LUE, P.; MAKANJU, A.; MALFACINI, K.; MANNING, S.; MARKOV, T.; MARKOVSKI, Y.; MARTIN, B.; MAYER, K.; MAYNE, A.; MCGREW, B.; MCKINNEY, S. M.; MCLEAVEY, C.; MCMILLAN, P.; MCNEIL, J.; MEDINA, D.; MEHTA, A.; MENICK, J.; METZ, L.; MISHCHENKO, A.; MISHKIN, P.; MONACO, V.; MORIKAWA, E.; MOSSING, D.; MU, T.; MURATI, M.; MURK, O.; MéLY, D.; NAIR, A.; NAKANO, R.; NAYAK, R.; NEELAKANTAN, A.; NGO, R.; NOH, H.; OUYANG, L.; O'KEEFE, C.; PACHOCKI, J.; PAINO, A.; PALERMO, J.; PANTULIANO, A.; PARASCANDOLO, G.; PARISH, J.; PARPARITA, E.; PASSOS, A.; PAVLOV, M.; PENG, A.; PERELMAN, A.; PERES, F. d. A. B.; PETROV, M.; PINTO, H. P. d. O.; MICHAEL; POKORNY; POKRASS, M.; PONG, V. H.; POWELL, T.; POWER, A.; POWER, B.; PROEHL, E.; PURI, R.; RADFORD, A.; RAE, J.; RAMESH, A.; RAYMOND, C.; REAL, F.; RIMBACH, K.; ROSS, C.; ROTSTED, B.; ROUSSEZ, H.; RYDER, N.; SALTARELLI, M.; SANDERS, T.; SANTURKAR, S.; SASTRY, G.; SCHMIDT, H.; SCHNURR, D.; SCHULMAN, J.; SELSAM, D.; SHEPPARD, K.; SHERBAKOV, T.; SHIEH, J.; SHOKER, S.; SHYAM, P.; SIDOR, S.; SIGLER, E.; SIMENS, M.; SITKIN, J.; SLAMA, K.; SOHL, I.; SOKOLOWSKY, B.; SONG, Y.; STAUDACHER, N.; SUCH, F. P.; SUMMERS, N.; SUTSKEVER, I.; TANG, J.; TEZAK, N.; THOMPSON, M. B.; TILLET, P.; TOOTOONCHIAN, A.; TSENG, E.; TUGGLE, P.; TURLEY, N.; TWOREK, J.; URIBE, J. F. C.; VALLONE, A.; VIJAYVERGIYA, A.; VOSS, C.; WAINWRIGHT, C.; WANG, J. J.; WANG, A.; WANG, B.; WARD, J.; WEI, J.; WEINMANN, C. J.; WELIHINDA, A.; WELINDER, P.; WENG, J.; WENG, L.; WIETHOFF, M.; WILLNER, D.; WINTER, C.; WOLRICH, S.; WONG, H.; WORKMAN, L.; WU, S.; WU, J.; WU, M.; XIAO, K.; XU, T.; YOO, S.; YU, K.; YUAN, Q.; ZAREMBA, W.; ZELLERS, R.; ZHANG, C.; ZHANG, M.; ZHAO, S.; ZHENG, T.; ZHUANG, J.; ZHUK, W.; ZOPH, B. **GPT-4 Technical Report**. arXiv:2303.08774 [cs].

OTTER, D. W.; MEDINA, J. R.; KALITA, J. K. A Survey of the Usages of Deep Learning for Natural Language Processing. **IEEE Transactions on Neural Networks and Learning Systems**, [S.l.], v. 32, n. 2, p. 604–624, Feb. 2021. Conference Name: IEEE Transactions on Neural Networks and Learning Systems.

PAN, S. J.; YANG, Q. A Survey on Transfer Learning. **IEEE Transactions on Knowledge and Data Engineering**, [S.l.], v. 22, n. 10, p. 1345–1359, Oct. 2010. Conference Name: IEEE Transactions on Knowledge and Data Engineering.

PAPINENI, K.; ROUKOS, S.; WARD, T.; ZHU, W.-J. BLEU: a method for automatic evaluation of machine translation. In: Annual Meeting ON Association FOR Computational Linguistics, 40., 2002, USA. **Proceedings...** Association for Computational Linguistics, 2002. p. 311–318. (ACL '02).

PATTERSON, D.; GONZALEZ, J.; LE, Q.; LIANG, C.; MUNGUIA, L.-M.; ROTHCHILD, D.; SO, D.; TEXIER, M.; DEAN, J. Carbon Emissions and Large Neural Network Training. **arXiv:2104.10350 [cs]**, [S.l.], Apr. 2021. arXiv: 2104.10350.

PEIXOTO, F. H. Projeto Victor: relato do desenvolvimento da Inteligência Artificial na Repercussão Geral do Supremo Tribunal Federal. **Revista Brasileira de Inteligência Artificial e Direito - RBIAD**, [S.l.], v. 1, n. 1, p. 1–22, July 2020. Number: 1.

PENNINGTON, J.; SOCHER, R.; MANNING, C. GloVe: Global Vectors for Word Representation. In: Conference ON Empirical Methods IN Natural Language Processing (EMNLP), 2014., 2014, Doha, Qatar. **Proceedings...** Association for Computational Linguistics, 2014. p. 1532–1543.

PETERS, M. E.; NEUMANN, M.; IYYER, M.; GARDNER, M.; CLARK, C.; LEE, K.; ZETTLEMOYER, L. Deep Contextualized Word Representations. In: Conference OF THE North American Chapter OF THE Association FOR Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), 2018., 2018, New Orleans, Louisiana. **Proceedings...** Association for Computational Linguistics, 2018. p. 2227–2237.

PETERSEN, K.; FELDT, R.; MUJTABA, S.; MATTSSON, M. Systematic Mapping Studies in Software Engineering. **Proceedings of the 12th International Conference on Evaluation and Assessment in Software Engineering**, [S.l.], v. 17, June 2008.

PETERSEN, K.; VAKKALANKA, S.; KUZNIARZ, L. Guidelines for conducting systematic mapping studies in software engineering: An update. **Information and Software Technology**, [S.l.], v. 64, p. 1–18, Aug. 2015.

PHANG, J.; FéVRY, T.; BOWMAN, S. R. Sentence Encoders on STILTs: Supplementary Training on Intermediate Labeled-data Tasks. **arXiv:1811.01088 [cs]**, [S.l.], Feb. 2019. arXiv: 1811.01088.

PIRES, R.; ABONIZIO, H.; ALMEIDA, T. S.; NOGUEIRA, R. **Sabiá**: Portuguese Large Language Models. arXiv:2304.07880.

POSNER, R. A. Legal Formalism, Legal Realism, and the Interpretation of Statutes and the Constitution. **Case Western Reserve Law Review**, [S.l.], v. 37, p. 179, 1986.

PRUTHI, D.; BANSAL, R.; DHINGRA, B.; SOARES, L. B.; COLLINS, M.; LIPTON, Z. C.; NEUBIG, G.; COHEN, W. W. Evaluating Explanations: How much do explanations from the teacher aid students? **arXiv:2012.00893 [cs]**, [S.l.], Dec. 2021. arXiv: 2012.00893.

PUSHP, P. K.; SRIVASTAVA, M. M. Train Once, Test Anywhere: Zero-Shot Learning for Text Classification. **arXiv:1712.05972 [cs]**, [S.l.], Dec. 2017. arXiv: 1712.05972.

QIN, L.; CHEN, Q.; ZHOU, Y.; CHEN, Z.; LI, Y.; LIAO, L.; LI, M.; CHE, W.; YU, P. S. **Multilingual Large Language Model**: A Survey of Resources, Taxonomy and Frontiers. arXiv:2404.04925.

QIU, X.; SUN, T.; XU, Y.; SHAO, Y.; DAI, N.; HUANG, X. Pre-trained models for natural language processing: A survey. **Science China Technological Sciences**, [S.l.], v. 63, n. 10, p. 1872–1897, Oct. 2020.

RADFORD, A.; NARASIMHAN, K.; SALIMANS, T.; SUTSKEVER, I. Improving language understanding by generative pre-training. **OpenAI blog**, [S.l.], 2018.

RADFORD, A.; WU, J.; CHILD, R.; LUAN, D.; AMODEI, D.; SUTSKEVER, I. Language models are unsupervised multitask learners. **OpenAI blog**, [S.l.], v. 1, n. 8, p. 9, 2019.

RAFFEL, C.; SHAZEER, N.; ROBERTS, A.; LEE, K.; NARANG, S.; MATENA, M.; ZHOU, Y.; LI, W.; LIU, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. **arXiv preprint arXiv:1910.10683**, [S.l.], 2019.

REAL, L.; FONSECA, E.; GONçALO OLIVEIRA, H. The ASSIN 2 Shared Task: A Quick Overview. In: COMPUTATIONAL Processing OF THE Portuguese Language: 14TH International Conference, PROPOR 2020, Evora, Portugal, March 2–4, 2020, Proceedings, 2020, Berlin, Heidelberg. **Anais...** Springer-Verlag, 2020. p. 406–412.

RIBEIRO, M. T.; SINGH, S.; GUESTRIN, C. **"Why Should I Trust You?"**: Explaining the Predictions of Any Classifier. arXiv:1602.04938 [cs, stat].

RIBEIRO, M. T.; SINGH, S.; GUESTRIN, C. Semantically Equivalent Adversarial Rules for Debugging NLP models. In: Annual Meeting OF THE Association FOR Computational Linguistics (Volume 1: Long Papers), 56., 2018, Melbourne, Australia. **Proceedings...** Association for Computational Linguistics, 2018. p. 856–865.

RISSLAND, E. L.; ASHLEY, K. D.; LOUI, R. AI and Law: A fruitful synergy. **Artificial Intelligence**, [S.l.], v. 150, n. 1-2, p. 1–15, Nov. 2003.

ROBALDO, L.; VILLATA, S.; WYNER, A.; GRABMAIR, M. Introduction for artificial intelligence and law: special issue "natural language processing for legal texts". **Artificial Intelligence and Law**, [S.l.], v. 27, n. 2, p. 113–115, June 2019.

RODRIGUES, R. C.; RODRIGUES, J.; CASTRO, P. V. Q. de; SILVA, N. F. F. da; SOARES, A. Portuguese Language Models and Word Embeddings: Evaluating on Semantic Similarity Tasks. In: COMPUTATIONAL Processing OF THE Portuguese Language, 2020, Cham. **Anais...** Springer International Publishing, 2020. p. 239–248. (Lecture Notes in Computer Science).

ROSS, A.; MARASOVIć, A.; PETERS, M. E. Explaining nlp models via minimal contrastive editing (mice). **arXiv preprint arXiv:2012.13985**, [S.l.], 2020.

RUDER, S. **On word embeddings - Part 1**. 2016.

RUDER, S. **NLP's ImageNet moment has arrived**. 2018.

RUDER, S. **Neural Transfer Learning for Natural Language Processing**. 2019. PhD thesis — National University of Ireland, Galway, 2019.

SAHOO, P.; SINGH, A. K.; SAHA, S.; JAIN, V.; MONDAL, S.; CHADHA, A. **A Systematic Survey of Prompt Engineering in Large Language Models**: Techniques and Applications. arXiv:2402.07927.

SAIAS, J.; QUARESMA, P. Semantic enrichment of a web legal information retrieval system. In: IN Trevor Bench-Capon, Aspassia Daskalopulu, AND Radboud Winkels, EDITORS, Legal Knowledge AND Information Systems. IOS, 2002. **Anais...** Press, 2002. p. 11–20.

SAIAS, J.; QUARESMA, P. A Methodology to Create Legal Ontologies in a Logic Programming Information Retrieval System. In: BENJAMINS, V. R.; CASANOVAS, P.; BREUKER, J.; GANGEMI, A. (Ed.). **Law and the Semantic Web**: Legal Ontologies, Methodologies, Legal Information Retrieval, and Applications. Berlin, Heidelberg: Springer, 2005. p. 185–200. (Lecture Notes in Computer Science).

SAKIYAMA, K.; NOGUEIRA, R.; ROMERO, R. A. F. Automated Keyphrase Generation for Brazilian Legal Information Retrieval. In: International Joint Conference ON Neural Networks (IJCNN), 2023., 2023. **Anais...** [S.l.: s.n.], 2023. p. 1–8. ISSN: 2161-4407.

SANH, V.; DEBUT, L.; CHAUMOND, J.; WOLF, T. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. **arXiv:1910.01108 [cs]**, [S.l.], 2019. arXiv: 1910.01108.

SANTOS, C. dos; GUIMARãES, V. Boosting Named Entity Recognition with Neural Character Embeddings. In: Fifth Named Entity Workshop, 2015, Beijing, China. **Proceedings...** Association for Computational Linguistics, 2015. p. 25–33.

SANTOS, D.; CARDOSO, N. A Golden Resource for Named Entity Recognition in Portuguese. In: COMPUTATIONAL Processing OF THE Portuguese Language, 2006, Berlin, Heidelberg. **Anais...** Springer, 2006. p. 69–79. (Lecture Notes in Computer Science).

SANTOS, J.; CONSOLI, B.; SANTOS, C. dos; TERRA, J.; COLLONINI, S.; VIEIRA, R. Assessing the Impact of Contextual Embeddings for Portuguese Named Entity Recognition. In: Brazilian Conference ON Intelligent Systems (BRACIS), 2019., 2019. **Anais...** [S.l.: s.n.], 2019. p. 437–442. ISSN: 2643-6264.

SAUNDERS, P. **Developing legal talent | Deloitte UK**. 2016.

SCHMIDHUBER, J. Deep learning in neural networks: An overview. **Neural Networks**, [S.l.], v. 61, p. 85–117, Jan. 2015.

SCHNEIDER, E. T. R.; SOUZA, J. V. A. de; KNAFOU, J.; OLIVEIRA, L. E. S. e.; COPARA, J.; GUMIEL, Y. B.; OLIVEIRA, L. F. A. d.; PARAISO, E. C.; TEODORO, D.; BARRA, C. M. C. M. BioBERTpt - A Portuguese Neural Language Model for Clinical Named Entity Recognition. In: Clinical Natural Language Processing Workshop, 3., 2020, Online. **Proceedings...** Association for Computational Linguistics, 2020. p. 65–72.

SCORDATO, M. R. **Book Review of Brian Tamanaha's Beyond the Formalist-Realist Divide**: The Role of Politics in Judging. Rochester, NY: Social Science Research Network, 2012. SSRN Scholarly Paper. (ID 2002156).

SELLAM, T.; DAS, D.; PARIKH, A. BLEURT: Learning Robust Metrics for Text Generation. In: Annual Meeting OF THE Association FOR Computational Linguistics, 58., 2020, Online. **Proceedings...** Association for Computational Linguistics, 2020. p. 7881–7892.

SENNRICH, R.; HADDOW, B.; BIRCH, A. Neural Machine Translation of Rare Words with Subword Units. In: Annual Meeting OF THE Association FOR Computational Linguistics (Volume 1: Long Papers), 54., 2016, Berlin, Germany. **Proceedings...** Association for Computational Linguistics, 2016. p. 1715–1725.

SHAIKH, R. A.; SAHU, T. P.; ANAND, V. Predicting Outcomes of Legal Cases based on Legal Factors using Classifiers. **International Conference on Computational Intelligence and Data Science**, [S.l.], v. 167, p. 2393–2402, Jan. 2020.

SHANAHAN, M. Talking about Large Language Models. **Communications of the ACM**, [S.l.], v. 67, n. 2, p. 68–79, Jan. 2024.

SHEIK, R.; NIRMALA, S. J. Deep Learning Techniques for Legal Text Summarization. In: IEEE 8TH Uttar Pradesh Section International Conference ON Electrical, Electronics AND Computer Engineering (UPCON), 2021., 2021. **Anais...** [S.l.: s.n.], 2021. p. 1–5. ISSN: 2687-7767.

SHEN, Y.; SUN, J.; LI, X.; ZHANG, L.; LI, Y.; SHEN, X. Legal Article-Aware End-To-End Memory Network for Charge Prediction. In: International Conference ON Computer Science AND Application Engineering, 2., 2018, Hohhot, China. **Proceedings...** Association for Computing Machinery, 2018. p. 1–5. (CSAE '18).

SHOEYBI, M.; PATWARY, M.; PURI, R.; LEGRESLEY, P.; CASPER, J.; CATANZARO, B. Megatron-LM: Training Multi-Billion Parameter Language Models Using Model Parallelism. **arXiv:1909.08053 [cs]**, [S.l.], Mar. 2020. arXiv: 1909.08053.

SHUKLA, A.; BHATTACHARYA, P.; PODDAR, S.; MUKHERJEE, R.; GHOSH, K.; GOYAL, P.; GHOSH, S. Legal Case Document Summarization: Extractive and Abstractive Methods and their Evaluation. In: Conference OF THE Asia-Pacific Chapter OF THE Association FOR Computational Linguistics AND THE 12TH International Joint Conference ON Natural Language Processing (Volume 1: Long Papers), 2., 2022, Online only. **Proceedings...** Association for Computational Linguistics, 2022. p. 1048–1064.

SILLS, A. **ROSS and Watson tackle the law**. 2016.

SIM, Y.; ROUTLEDGE, B.; SMITH, N. A. The Utility of Text: The Case of Amicus Briefs and the Supreme Court. **arXiv:1409.7985 [cs]**, [S.l.], Nov. 2014. arXiv: 1409.7985.

SINGLA, P.; DOMINGOS, P. Discriminative training of Markov logic networks. In: Artificial INTELLIGENCE - Volume 2, 20., 2005, Pittsburgh, Pennsylvania. **Proceedings...** AAAI Press, 2005. p. 868–873. (AAAI'05).

SMITH, B.; WELTY, C. FOIS introduction: Ontology—towards a new synthesis. In: Formal Ontology IN Information Systems - Volume 2001, 2001, New York, NY, USA. **Proceedings...** Association for Computing Machinery, 2001. p. .3–.9. (FOIS '01).

SOCHER, R.; PERELYGIN, A.; WU, J.; CHUANG, J.; MANNING, C. D.; NG, A.; POTTS, C. Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank. In: Conference ON Empirical Methods IN Natural Language Processing, 2013., 2013, Seattle, Washington, USA. **Proceedings...** Association for Computational Linguistics, 2013. p. 1631–1642.

SOUZA, F.; NOGUEIRA, R.; LOTUFO, R. BERTimbau: Pretrained BERT Models for Brazilian Portuguese. In: INTELLIGENT Systems, 2020, Cham. **Anais...** Springer International Publishing, 2020. p. 403–417. (Lecture Notes in Computer Science).

SPEROTTO, F. A.; BELCHIOR, M.; AGUIAR, M. S. de. Ontology-Based Legal System in Multi-agents Systems. In: ADVANCES IN Soft Computing, 2019, Cham. **Anais...** Springer International Publishing, 2019. p. 507–521. (Lecture Notes in Computer Science).

STRUBELL, E.; GANESH, A.; MCCALLUM, A. Energy and Policy Considerations for Deep Learning in NLP. **arXiv:1906.02243 [cs]**, [S.l.], June 2019. arXiv: 1906.02243.

SUKHBAATAR, S.; SZLAM, a.; WESTON, J.; FERGUS, R. End-To-End Memory Networks. In: CORTES, C.; LAWRENCE, N. D.; LEE, D. D.; SUGIYAMA, M.; GARNETT, R. (Ed.). **Advances in Neural Information Processing Systems 28**. [S.l.]: Curran Associates, Inc., 2015. p. 2440–2448.

SULEA, O.-M.; ZAMPIERI, M.; MALMASI, S.; VELA, M.; DINU, L. P.; GENABITH, J. van. Exploring the Use of Text Classification in the Legal Domain. **arXiv:1710.09306 [cs]**, [S.l.], Oct. 2017. arXiv: 1710.09306.

SUN, C.; QIU, X.; XU, Y.; HUANG, X. How to Fine-Tune BERT for Text Classification? In: CHINESE Computational Linguistics: 18TH China National Conference, CCL 2019, Kunming, China, October 18–20, 2019, Proceedings, 2019, Berlin, Heidelberg. **Anais...** Springer-Verlag, 2019. p. 194–206.

SURDEN, H. **Artificial Intelligence and Law**: An Overview. Rochester, NY: Social Science Research Network, 2019. SSRN Scholarly Paper. (ID 3411869).

SUTOR, P.; ALOIMONOS, Y.; FERMULLER, C.; SUMMERS-STAY, D. Metaconcepts: Isolating Context in Word Embeddings. In: IEEE Conference ON Multimedia Information Processing AND Retrieval (MIPR), 2019., 2019. **Anais...** [S.l.: s.n.], 2019. p. 544–549.

SUTSKEVER, I.; VINYALS, O.; LE, Q. V. Sequence to Sequence Learning with Neural Networks. In: ADVANCES IN Neural Information Processing Systems, 2014. **Anais...** Curran Associates: Inc., 2014. v. 27.

SUáREZ, P. J. O.; SAGOT, B.; ROMARY, L. Asynchronous Pipeline for Processing Huge Corpora on Medium to Low Resource Infrastructures. In: ., 2019. **Anais...** Leibniz-Institut für Deutsche Sprache, 2019.

TAI, K. S.; SOCHER, R.; MANNING, C. D. Improved Semantic Representations From Tree-Structured Long Short-Term Memory Networks. In: Annual Meeting OF THE Association FOR Computational Linguistics AND THE 7TH International Joint Conference ON Natural Language Processing (Volume 1: Long Papers), 53., 2015, Beijing, China. **Proceedings...** Association for Computational Linguistics, 2015. p. 1556–1566.

TAYLOR, R.; KARDAS, M.; CUCURULL, G.; SCIALOM, T.; HARTSHORN, A.; SARAVIA, E.; POULTON, A.; KERKEZ, V.; STOJNIC, R. **Galactica**: A Large Language Model for Science. arXiv:2211.09085 [cs, stat].

TENNEY, I.; DAS, D.; PAVLICK, E. BERT Rediscovers the Classical NLP Pipeline. In: Annual Meeting OF THE Association FOR Computational Linguistics, 57., 2019, Florence, Italy. **Proceedings...** Association for Computational Linguistics, 2019. p. 4593–4601.

TJONG KIM SANG, E. F. Introduction to the CoNLL-2002 Shared Task: Language-Independent Named Entity Recognition. In: COLING-02: The 6TH Conference ON Natural Language Learning 2002 (CoNLL-2002), 2002. **Anais...** [S.l.: s.n.], 2002.

TJONG KIM SANG, E. F.; DE MEULDER, F. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In: Seventh Conference ON Natural Language Learning AT HLT-NAACL 2003, 2003. **Proceedings...** [S.l.: s.n.], 2003. p. 142–147.

TOUVRON, H.; LAVRIL, T.; IZACARD, G.; MARTINET, X.; LACHAUX, M.-A.; LACROIX, T.; ROZIèRE, B.; GOYAL, N.; HAMBRO, E.; AZHAR, F.; RODRIGUEZ, A.; JOULIN, A.; GRAVE, E.; LAMPLE, G. **LLaMA**: Open and Efficient Foundation Language Models. arXiv:2302.13971 [cs].

TURIAN, J.; RATINOV, L.-A.; BENGIO, Y. Word Representations: A Simple and General Method for Semi-Supervised Learning. In: Annual Meeting OF THE Association FOR Computational Linguistics, 48., 2010, Uppsala, Sweden. **Proceedings...** Association for Computational Linguistics, 2010. p. 384–394.

VARGAS FEIJó, D. de; MOREIRA, V. P. RulingBR: A Summarization Dataset for Legal Texts. In: COMPUTATIONAL Processing OF THE Portuguese Language, 2018, Cham. **Anais...** Springer International Publishing, 2018. p. 255–264.

VASWANI, A.; SHAZEER, N.; PARMAR, N.; USZKOREIT, J.; JONES, L.; GOMEZ, A. N.; KAISER, L.; POLOSUKHIN, I. Attention is all you need. In: International Conference ON Neural Information Processing Systems, 31., 2017, Red Hook, NY, USA. **Proceedings...** Curran Associates Inc., 2017. p. 6000–6010. (NIPS'17).

VATSAL, S.; DUBEY, H. **A Survey of Prompt Engineering Methods in Large Language Models for Different NLP Tasks**. arXiv:2407.12994.

VERGARA, J. R.; ESTéVEZ, P. A. A Review of Feature Selection Methods Based on Mutual Information. **Neural Computing and Applications**, [S.l.], v. 24, n. 1, p. 175–186, Jan. 2014. arXiv: 1509.07577.

VIRTUCIO, M. B. L.; ABOROT, J. A.; ABONITA, J. K. C.; AVIñANTE, R. S.; COPINO, R. J. B.; NEVERIDA, M. P.; OSIANA, V. O.; PERAMO, E. C.; SYJUCO, J. G.; TAN, G. B. A. Predicting Decisions of the Philippine Supreme Court Using Natural Language Processing and Machine Learning. In: IEEE 42ND Annual Computer Software AND Applications Conference (COMPSAC), 2018., 2018. **Anais...** [S.l.: s.n.], 2018. v. 02, p. 130–135. ISSN: 0730-3157.

VISENTIN, A.; NARDOTTO, A.; O'SULLIVAN, B. Predicting Judicial Decisions: A Statistically Rigorous Approach and a New Ensemble Classifier. In: IEEE 31ST International Conference ON Tools WITH Artificial Intelligence (ICTAI), 2019., 2019. **Anais...** [S.l.: s.n.], 2019. p. 1820–1824. ISSN: 2375-0197.

WAGNER FILHO, J. A.; WILKENS, R.; IDIART, M.; VILLAVICENCIO, A. The brWaC Corpus: A New Open Resource for Brazilian Portuguese. In: Eleventh International Conference ON Language Resources AND Evaluation (LREC 2018), 2018, Miyazaki, Japan. **Proceedings...** European Language Resources Association (ELRA), 2018.

WALKER, V. R. A Default-Logic Paradigm for Legal Fact-Finding. **Jurimetrics**, [S.l.], v. 47, n. 2, p. 193–243, 2007. Publisher: American Bar Association.

WALKER, V. R.; HAN, J. H.; NI, X.; YOSEDA, K. Semantic types for computational legal reasoning: propositional connectives and sentence roles in the veterans' claims dataset. In: International Conference ON Articial Intelligence AND Law, 16., 2017, London, United Kingdom. **Proceedings...** Association for Computing Machinery, 2017. p. 217–226. (ICAIL '17).

WALTL, B.; BONCZEK, G.; SCEPANKOVA, E.; MATTHES, F. Semantic types of legal norms in German laws: classification and analysis using local linear explanations. **Artificial Intelligence and Law**, [S.l.], v. 27, n. 1, p. 43–71, Mar. 2019.

WANG, Y.; WANG, L.; LI, Y.; HE, D.; LIU, T.-Y.; CHEN, W. A Theoretical Analysis of NDCG Type Ranking Measures. **arXiv:1304.6480 [cs, stat]**, [S.l.], Apr. 2013. arXiv: 1304.6480.

WANG, Y.; YAO, Q.; KWOK, J. T.; NI, L. M. Generalizing from a Few Examples: A Survey on Few-shot Learning. **ACM Computing Surveys**, [S.l.], v. 53, n. 3, p. 63:1–63:34, June 2020.

WEBLEY, L. Qualitative Approaches to Empirical Legal Research. In: **The Oxford Handbook of Empirical Legal Research**. [S.l.: s.n.], 2010. p. 927–950. Journal Abbreviation: The Oxford Handbook of Empirical Legal Research.

WEI, J.; WANG, X.; SCHUURMANS, D.; BOSMA, M.; ICHTER, B.; XIA, F.; CHI, E.; LE, Q. V.; ZHOU, D. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. **Advances in Neural Information Processing Systems**, [S.l.], v. 35, p. 24824–24837, Dec. 2022.

WESTERMANN, H.; WALKER, V. R.; ASHLEY, K. D.; BENYEKHLEF, K. Using Factors to Predict and Analyze Landlord-Tenant Decisions to Increase Access to Justice. In: Seventeenth International Conference ON Artificial Intelligence AND Law, 2019, Montreal, QC, Canada. **Proceedings...** Association for Computing Machinery, 2019. p. 133–142. (ICAIL '19).

WIEGREFFE, S.; PINTER, Y. Attention is not not Explanation. In: Conference ON Empirical Methods IN Natural Language Processing AND THE 9TH International Joint Conference ON Natural Language Processing (EMNLP-IJCNLP), 2019., 2019, Hong Kong, China. **Proceedings...** Association for Computational Linguistics, 2019. p. 11–20.

WILLIAN SOUSA, A.; FABRO, M. Iudicium Textum Dataset Uma Base de Textos Jurídicos para NLP. In: Dataset Showcase Workshop AT SBBD (Brazilian Symposium ON Databases), 2., 2019, Fortaleza, Brazil. **Anais...** [S.l.: s.n.], 2019.

WOHLIN, C. Guidelines for snowballing in systematic literature studies and a replication in software engineering. In: International Conference ON Evaluation AND Assessment IN Software Engineering, 18., 2014, New York, NY, USA. **Proceedings...** Association for Computing Machinery, 2014. p. 1–10. (EASE '14).

WOLF, T.; DEBUT, L.; SANH, V.; CHAUMOND, J.; DELANGUE, C.; MOI, A.; CISTAC, P.; RAULT, T.; LOUF, R.; FUNTOWICZ, M.; DAVISON, J.; SHLEIFER, S.; PLATEN, P. von; MA, C.; JERNITE, Y.; PLU, J.; XU, C.; LE SCAO, T.; GUGGER, S.; DRAME, M.; LHOEST, Q.; RUSH, A. Transformers: State-of-the-Art Natural Language Processing. In: Conference ON Empirical Methods IN Natural Language Processing: System Demonstrations, 2020., 2020, Online. **Proceedings...** Association for Computational Linguistics, 2020. p. 38–45.

WORKSHOP, B.; SCAO, T. L.; FAN, A.; AKIKI, C.; PAVLICK, E.; ILIć, S.; HESSLOW, D.; CASTAGNé, R.; LUCCIONI, A. S.; YVON, F.; GALLé, M.; TOW, J.; RUSH, A. M.; BIDERMAN, S.; WEBSON, A.; AMMANAMANCHI, P. S.; WANG, T.; SAGOT, B.; MUENNIGHOFF, N.; MORAL, A. V. d.; RUWASE, O.; BAWDEN, R.; BEKMAN, S.; MCMILLAN-MAJOR, A.; BELTAGY, I.; NGUYEN, H.; SAULNIER, L.; TAN, S.; SUAREZ,

P. O.; SANH, V.; LAURENçON, H.; JERNITE, Y.; LAUNAY, J.; MITCHELL, M.; RAFFEL, C.; GOKASLAN, A.; SIMHI, A.; SOROA, A.; AJI, A. F.; ALFASSY, A.; ROGERS, A.; NITZAV, A. K.; XU, C.; MOU, C.; EMEZUE, C.; KLAMM, C.; LEONG, C.; STRIEN, D. v.; ADELANI, D. I.; RADEV, D.; PONFERRADA, E. G.; LEVKOVIZH, E.; KIM, E.; NATAN, E. B.; TONI, F. D.; DUPONT, G.; KRUSZEWSKI, G.; PISTILLI, G.; ELSAHAR, H.; BENYAMINA, H.; TRAN, H.; YU, I.; ABDULMUMIN, I.; JOHNSON, I.; GONZALEZ-DIOS, I.; ROSA, J. d. l.; CHIM, J.; DODGE, J.; ZHU, J.; CHANG, J.; FROHBERG, J.; TOBING, J.; BHATTACHARJEE, J.; ALMUBARAK, K.; CHEN, K.; LO, K.; WERRA, L. V.; WEBER, L.; PHAN, L.; ALLAL, L. B.; TANGUY, L.; DEY, M.; MUñOZ, M. R.; MASOUD, M.; GRANDURY, M.; ŠAšKO, M.; HUANG, M.; COAVOUX, M.; SINGH, M.; JIANG, M. T.-J.; VU, M. C.; JAUHAR, M. A.; GHALEB, M.; SUBRAMANI, N.; KASSNER, N.; KHAMIS, N.; NGUYEN, O.; ESPEJEL, O.; GIBERT, O. d.; VILLEGAS, P.; HENDERSON, P.; COLOMBO, P.; AMUOK, P.; LHOEST, Q.; HARLIMAN, R.; BOMMASANI, R.; LóPEZ, R. L.; RIBEIRO, R.; OSEI, S.; PYYSALO, S.; NAGEL, S.; BOSE, S.; MUHAMMAD, S. H.; SHARMA, S.; LONGPRE, S.; NIKPOOR, S.; SILBERBERG, S.; PAI, S.; ZINK, S.; TORRENT, T. T.; SCHICK, T.; THRUSH, T.; DANCHEV, V.; NIKOULINA, V.; LAIPPALA, V.; LEPERCQ, V.; PRABHU, V.; ALYAFEAI, Z.; TALAT, Z.; RAJA, A.; HEINZERLING, B.; SI, C.; TAşAR, D. E.; SALESKY, E.; MIELKE, S. J.; LEE, W. Y.; SHARMA, A.; SANTILLI, A.; CHAFFIN, A.; STIEGLER, A.; DATTA, D.; SZCZECHLA, E.; CHHABLANI, G.; WANG, H.; PANDEY, H.; STROBELT, H.; FRIES, J. A.; ROZEN, J.; GAO, L.; SUTAWIKA, L.; BARI, M. S.; AL-SHAIBANI, M. S.; MANICA, M.; NAYAK, N.; TEEHAN, R.; ALBANIE, S.; SHEN, S.; BEN-DAVID, S.; BACH, S. H.; KIM, T.; BERS, T.; FEVRY, T.; NEERAJ, T.; THAKKER, U.; RAUNAK, V.; TANG, X.; YONG, Z.-X.; SUN, Z.; BRODY, S.; URI, Y.; TOJARIEH, H.; ROBERTS, A.; CHUNG, H. W.; TAE, J.; PHANG, J.; PRESS, O.; LI, C.; NARAYANAN, D.; BOURFOUNE, H.; CASPER, J.; RASLEY, J.; RYABININ, M.; MISHRA, M.; ZHANG, M.; SHOEYBI, M.; PEYROUNETTE, M.; PATRY, N.; TAZI, N.; SANSEVIERO, O.; PLATEN, P. v.; CORNETTE, P.; LAVALLéE, P. F.; LACROIX, R.; RAJBHANDARI, S.; GANDHI, S.; SMITH, S.; REQUENA, S.; PATIL, S.; DETTMERS, T.; BARUWA, A.; SINGH, A.; CHEVELEVA, A.; LIGOZAT, A.-L.; SUBRAMONIAN, A.; NéVéOL, A.; LOVERING, C.; GARRETTE, D.; TUNUGUNTLA, D.; REITER, E.; TAKTASHEVA, E.; VOLOSHINA, E.; BOGDANOV, E.; WINATA, G. I.; SCHOELKOPF, H.; KALO, J.-C.; NOVIKOVA, J.; FORDE, J. Z.; CLIVE, J.; KASAI, J.; KAWAMURA, K.; HAZAN, L.; CARPUAT, M.; CLINCIU, M.; KIM, N.; CHENG, N.; SERIKOV, O.; ANTVERG, O.; WAL, O. v. d.; ZHANG, R.; ZHANG, R.; GEHRMANN, S.; MIRKIN, S.; PAIS, S.; SHAVRINA, T.; SCIALOM, T.; YUN, T.; LIMISIEWICZ, T.; RIESER, V.; PROTASOV, V.; MIKHAILOV, V.; PRUKSACHATKUN, Y.; BELINKOV, Y.; BAMBERGER, Z.; KASNER, Z.; RUEDA, A.; PESTANA, A.; FEIZPOUR, A.; KHAN, A.; FARANAK, A.; SANTOS, A.; HEVIA, A.; UNLDREAJ, A.; AGHAGOL, A.; ABDOLLAHI, A.; TAMMOUR, A.; HAJIHOSSEINI, A.; BEHROOZI, B.; AJIBADE, B.; SAXENA, B.; FERRANDIS, C. M.; MCDUFF, D.; CONTRACTOR, D.; LANSKY, D.; DAVID, D.; KIELA, D.; NGUYEN, D. A.; TAN, E.; BAYLOR, E.; OZOANI, E.; MIRZA, F.; ONONIWU, F.; REZANEJAD, H.; JONES, H.; BHATTACHARYA, I.; SOLAIMAN, I.; SEDENKO, I.; NEJADGHOLI, I.; PASSMORE, J.; SELTZER, J.; SANZ, J. B.; DUTRA, L.; SAMAGAIO, M.; ELBADRI, M.; MIESKES, M.; GERCHICK, M.; AKINLOLU, M.; MCKENNA, M.; QIU, M.; GHAURI, M.; BURYNOK, M.; ABRAR, N.; RAJANI, N.; ELKOTT, N.; FAHMY, N.; SAMUEL, O.; AN, R.; KROMANN, R.; HAO, R.; ALIZADEH, S.; SHUBBER, S.; WANG, S.; ROY, S.; VIGUIER, S.; LE, T.; OYEBADE, T.; LE, T.; YANG, Y.; NGUYEN, Z.; KASHYAP, A. R.;

PALASCIANO, A.; CALLAHAN, A.; SHUKLA, A.; MIRANDA-ESCALADA, A.; SINGH, A.; BEILHARZ, B.; WANG, B.; BRITO, C.; ZHOU, C.; JAIN, C.; XU, C.; FOURRIER, C.; PERIñáN, D. L.; MOLANO, D.; YU, D.; MANJAVACAS, E.; BARTH, F.; FUHRIMANN, F.; ALTAY, G.; BAYRAK, G.; BURNS, G.; VRABEC, H. U.; BELLO, I.; DASH, I.; KANG, J.; GIORGI, J.; GOLDE, J.; POSADA, J. D.; SIVARAMAN, K. R.; BULCHANDANI, L.; LIU, L.; SHINZATO, L.; BYKHOVETZ, M. H. d.; TAKEUCHI, M.; PàMIES, M.; CASTILLO, M. A.; NEZHURINA, M.; SäNGER, M.; SAMWALD, M.; CULLAN, M.; WEINBERG, M.; WOLF, M. D.; MIHALJCIC, M.; LIU, M.; FREIDANK, M.; KANG, M.; SEELAM, N.; DAHLBERG, N.; BROAD, N. M.; MUELLNER, N.; FUNG, P.; HALLER, P.; CHANDRASEKHAR, R.; EISENBERG, R.; MARTIN, R.; CANALLI, R.; SU, R.; SU, R.; CAHYAWIJAYA, S.; GARDA, S.; DESHMUKH, S. S.; MISHRA, S.; KIBLAWI, S.; OTT, S.; SANG-AROONSIRI, S.; KUMAR, S.; SCHWETER, S.; BHARATI, S.; LAUD, T.; GIGANT, T.; KAINUMA, T.; KUSA, W.; LABRAK, Y.; BAJAJ, Y. S.; VENKATRAMAN, Y.; XU, Y.; XU, Y.; XU, Y.; TAN, Z.; XIE, Z.; YE, Z.; BRAS, M.; BELKADA, Y.; WOLF, T. **BLOOM**: A 176B-Parameter Open-Access Multilingual Language Model. arXiv:2211.05100.

WU, T.; RIBEIRO, M. T.; HEER, J.; WELD, D. Polyjuice: Generating Counterfactuals for Explaining, Evaluating, and Improving Models. In: Annual Meeting OF THE Association FOR Computational Linguistics AND THE 11TH International Joint Conference ON Natural Language Processing (Volume 1: Long Papers), 59., 2021, Online. **Proceedings...** Association for Computational Linguistics, 2021. p. 6707–6723.

XIAO, C.; ZHONG, H.; GUO, Z.; TU, C.; LIU, Z.; SUN, M.; FENG, Y.; HAN, X.; HU, Z.; WANG, H.; XU, J. CAIL2018: A Large-Scale Legal Dataset for Judgment Prediction. **arXiv:1807.02478 [cs]**, [S.l.], July 2018. arXiv: 1807.02478.

XU, L.; XIE, H.; QIN, S.-Z. J.; TAO, X.; WANG, F. L. **Parameter-Efficient Fine-Tuning Methods for Pretrained Language Models**: A Critical Review and Assessment. arXiv:2312.12148.

XUE, L.; CONSTANT, N.; ROBERTS, A.; KALE, M.; AL-RFOU, R.; SIDDHANT, A.; BARUA, A.; RAFFEL, C. **mT5**: A massively multilingual pre-trained text-to-text transformer. arXiv:2010.11934.

YADAV, P.; TAM, D.; CHOSHEN, L.; RAFFEL, C.; BANSAL, M. **TIES-Merging**: Resolving Interference When Merging Models. arXiv:2306.01708.

YANG, L.; CHEN, H.; LI, Z.; DING, X.; WU, X. **Give Us the Facts**: Enhancing Large Language Models with Knowledge Graphs for Fact-aware Language Modeling. arXiv:2306.11489.

YANG, Z.; YANG, D.; DYER, C.; HE, X.; SMOLA, A.; HOVY, E. Hierarchical Attention Networks for Document Classification. In: Conference OF THE North American Chapter OF THE Association FOR Computational Linguistics: Human Language Technologies, 2016., 2016, San Diego, California. **Proceedings...** Association for Computational Linguistics, 2016. p. 1480–1489.

YU, L.; YU, B.; YU, H.; HUANG, F.; LI, Y. **Language Models are Super Mario**: Absorbing Abilities from Homologous Models as a Free Lunch. arXiv:2311.03099.

ZELLERS, R.; HOLTZMAN, A.; BISK, Y.; FARHADI, A.; CHOI, Y. HellaSwag: Can a Machine Really Finish Your Sentence? In: Annual Meeting OF THE Association FOR

Computational Linguistics, 57., 2019, Florence, Italy. **Proceedings...** Association for Computational Linguistics, 2019. p. 4791–4800.

ZENG, D.; LIU, K.; LAI, S.; ZHOU, G.; ZHAO, J. Relation Classification via Convolutional Deep Neural Network. In: COLING 2014, THE 25TH International Conference ON Computational Linguistics: Technical Papers, 2014, Dublin, Ireland. **Proceedings...** Dublin City University and Association for Computational Linguistics, 2014. p. 2335–2344.

ZHAI, F.; POTDAR, S.; XIANG, B.; ZHOU, B. Neural Models for Sequence Chunking. In: THIRTY-First AAAI Conference ON Artificial Intelligence, 2017. **Anais...** [S.l.: s.n.], 2017.

ZHANG, H.; BABAR, M. A.; TELL, P. Identifying relevant studies in software engineering. **Information and Software Technology**, [S.l.], v. 53, n. 6, p. 625–637, June 2011.

ZHANG, H.; WANG, X.; TAN, H.; LI, R. Applying Data Discretization to DPCNN for Law Article Prediction. In: NATURAL Language Processing AND Chinese Computing, 2019, Cham. **Anais...** Springer International Publishing, 2019. p. 459–470. (Lecture Notes in Computer Science).

ZHANG, M.-L.; ZHOU, Z.-H. ML-KNN: A lazy learning approach to multi-label learning. **Pattern Recognition**, [S.l.], v. 40, n. 7, p. 2038–2048, July 2007.

ZHANG, S.; YAN, G.; LI, Y.; LIU, J. Evaluation of Judicial Imprisonment Term Prediction Model Based on Text Mutation. In: IEEE 19TH International Conference ON Software Quality, Reliability AND Security Companion (QRS-C), 2019., 2019. **Anais...** [S.l.: s.n.], 2019. p. 62–65.

ZHANG, T.; KISHORE, V.; WU, F.; WEINBERGER, K. Q.; ARTZI, Y. **BERTScore**: Evaluating Text Generation with BERT. arXiv:1904.09675 [cs].

ZHANG, Z.; ZHANG, A.; LI, M.; SMOLA, A. **Automatic Chain of Thought Prompting in Large Language Models**. arXiv:2210.03493 [cs].

ZHAO, W. X.; ZHOU, K.; LI, J.; TANG, T.; WANG, X.; HOU, Y.; MIN, Y.; ZHANG, B.; ZHANG, J.; DONG, Z.; DU, Y.; YANG, C.; CHEN, Y.; CHEN, Z.; JIANG, J.; REN, R.; LI, Y.; TANG, X.; LIU, Z.; LIU, P.; NIE, J.-Y.; WEN, J.-R. **A Survey of Large Language Models**. arXiv:2303.18223 [cs].

ZHENG, L.; CHIANG, W.-L.; SHENG, Y.; ZHUANG, S.; WU, Z.; ZHUANG, Y.; LIN, Z.; LI, Z.; LI, D.; XING, E.; ZHANG, H.; GONZALEZ, J. E.; STOICA, I. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. **Advances in Neural Information Processing Systems**, [S.l.], v. 36, p. 46595–46623, Dec. 2023.

ZHOU, Y.; MURESANU, A. I.; HAN, Z.; PASTER, K.; PITIS, S.; CHAN, H.; BA, J. **Large Language Models Are Human-Level Prompt Engineers**. arXiv:2211.01910 [cs].

ÖZçIFT, A.; AKARSU, K.; YUMUK, F.; SöYLEMEZ, C. Advancing natural language processing (NLP) applications of morphologically rich languages with bidirectional encoder representations from transformers (BERT): an empirical case study for Turkish. **Automatika**, [S.l.], v. 62, n. 2, p. 226–238, Apr. 2021.

APPENDIX A PROMPTS USED IN THE AUTOMATED JUDGMENT REPORT GENERATION EXPERIMENT

A.1 Prompts for Judgment Report Generator

This appendix presents the prompt templates employed in the proposed solution. All prompt templates used are listed below, organized by document type and version. Variables such as {TEXTO_DOC}, {TEXTOS_INFERENCIAS_ANTERIORES_CONCATENADOS}, and {TEXTS_PREVIOUS_INFERENCES_CONCATENATED} represent placeholders that are dynamically replaced with actual input during inference [1].

All prompt templates are implemented in Python using a structured prompt-template approach. They can be directly used with language model APIs as shown in the following code example:

Listing A.1: Prompt templates usage example

```
text = "Text from the judicial document..."
prompt =
    PROMPT_TEMPLATES["initial_petition"].substitute(TEXT_DOC=text)
print(prompt)
```

Listing A.2: Prompts common to version 1 and version 2 (Portuguese)

```
from string import Template

PROMPT_TEMPLATES = {
    # Prompts common to version 1 and version 2 (Portuguese)
    "peticao_inicial": Template(
        "Você éum assessor judiciário especializado em análise de
            autos processuais. "
        "Para elaboração da resposta considere somente o texto incluso
            no prompt. "
        "A partir da petição inicial abaixo indique os autores e réus
            da ação proposta. "
        "Faça um resumo destacando os fatos ocorridos, as razões
            jurídicas e os pedidos dos autores.\n\n"
        "${TEXTO_DOC}"
    ),
    "despacho_decisao": Template(
        "Você éum assessor judiciário especializado em análise de
            autos processuais. "
```

[1] All prompts are also available on Github: https://github.com/lucianozanuz/Manuscript-Number-JJIMEI-D-25-00093—Supplementary-repository.

```
14        "Para elaboração da resposta considere somente o texto incluso
              no prompt. "
15        "Faça um resumo da decisão proferida abaixo pelo juiz. "
16        "Utilize o tempo verbal no pretérito pois já houve
              cumprimento.\n\n"
17        "${TEXTO_DOC}"
18    ),
19    "contestacao": Template(
20        "Você éum assessor judiciário especializado em análise de
              autos processuais. "
21        "Para elaboração da resposta considere somente o texto incluso
              no prompt. "
22        "Faça um resumo do documento de contestação abaixo destacando
              as razões jurídicas e os pedidos.\n\n"
23        "${TEXTO_DOC}"
24    ),
25    "replica": Template(
26        "Você éum assessor judiciário especializado em análise de
              autos processuais. "
27        "Para elaboração da resposta considere somente o texto incluso
              no prompt. "
28        "Faça um resumo do documento de réplica abaixo destacando as
              razões jurídicas e os pedidos.\n\n"
29        "${TEXTO_DOC}"
30    ),
31    "resposta": Template(
32        "Você éum assessor judiciário especializado em análise de
              autos processuais. "
33        "Para elaboração da resposta considere somente o texto incluso
              no prompt. "
34        "Faça um resumo do documento abaixo destacando as razões
              jurídicas e os pedidos.\n\n"
35        "${TEXTO_DOC}"
36    ),
37    "termo_audiencia": Template(
38        "Você éum assessor judiciário especializado em análise de
              autos processuais. "
39        "Para elaboração da resposta considere somente o texto incluso
              no prompt. "
40        "Faça um resumo do termo de audiência abaixo. Indique o nome
              das testemunhas ouvidas.\n\n"
```

```
41          "${TEXTO_DOC}"
42      ),
43      "memoriais": Template(
44          "Você éum assessor judiciário especializado em análise de
                autos processuais. "
45          "Para elaboração da resposta considere somente o texto incluso
                no prompt. "
46          "Faça um resumo do documento abaixo destacando as razões
                jurídicas e os pedidos.\n\n"
47          "${TEXTO_DOC}"
48      ),
49      "denuncia": Template(
50          "Você éum assessor judiciário especializado em análise de
                autos processuais. "
51          "Para elaboração da resposta considere somente o texto incluso
                no prompt. "
52          "A partir da denúncia abaixo indique os réus, as vítimas e a
                data do fato típico. "
53          "Faça um resumo destacando os fatos, as razões jurídicas e os
                pedidos.\n\n"
54          "${TEXTO_DOC}"
55      ),
56      "defesa_previa": Template(
57          "Você éum assessor judiciário especializado em análise de
                autos processuais. "
58          "Para elaboração da resposta considere somente o texto incluso
                no prompt. "
59          "Faça um resumo da defesa do réu descrita no documento abaixo,
                destacando as razões jurídicas e os pedidos.\n\n"
60          "${TEXTO_DOC}"
61      ),
62      "alegacoes_finais": Template(
63          "Você éum assessor judiciário especializado em análise de
                autos processuais. "
64          "Para elaboração da resposta considere somente o texto incluso
                no prompt. "
65          "A partir das alegações finais abaixo faça um resumo das
                razões jurídicas. "
66          "Faça também um resumo dos pedidos.\n\n"
67          "${TEXTO_DOC}"
68      ),
```

```
    "outros_documentos": Template(
        "Você éum assessor judiciário especializado em análise de
            autos processuais. "
        "Para elaboração da resposta considere somente o texto incluso
            no prompt. "
        "Faça um resumo do documento abaixo.\n\n"
        "${TEXTO_DOC}"
    ),
}
```

Listing A.3: Prompts common to version 1 and version 2 (English)

```
from string import Template

PROMPT_TEMPLATES = {
    # Prompts common to version 1 and version 2 (English)
    "initial_petition": Template(
        "You are a judicial advisor specialized in analyzing
            procedural records. "
        "To prepare the response, consider only the text included in
            the prompt. "
        "From the initial petition below, indicate the plaintiffs and
            defendants of the proposed action. "
        "Write a summary highlighting the facts that occurred, the
            legal reasons and the plaintiffs' requests.\n\n"
        "${TEXT_DOC}"
    ),
    "order_decision": Template(
        "You are a judicial advisor specialized in analyzing
            procedural records. "
        "To prepare the response, consider only the text included in
            the prompt. "
        "Write a summary of the decision handed down below by the
            judge. "
        "Use the past tense because it has already been complied
            with.\n\n"
        "${TEXT_DOC}"
    ),
    "contestation": Template(
        "You are a judicial advisor specialized in analyzing
            procedural records. "
        "To prepare the response, consider only the text included in
```

```
            the prompt. "
        "Summarize the response document below, highlighting the legal
            reasons and requests.\n\n"
        "${TEXT_DOC}"
    ),
    "reply": Template(
        "You are a judicial advisor specialized in analyzing
            procedural records. "
        "To prepare the response, consider only the text included in
            the prompt. "
        "Summarize the response document below, highlighting the legal
            reasons and requests.\n\n"
        "${TEXT_DOC}"
    ),
    "response": Template(
        "You are a judicial advisor specialized in analyzing
            procedural records. "
        "To prepare the response, consider only the text included in
            the prompt. "
        "Summarize the document below, highlighting the legal reasons
            and requests.\n\n"
        "${TEXT_DOC}"
    ),
    "hearing_instrument": Template(
        "You are a judicial advisor specialized in analyzing
            procedural records. "
        "To prepare the response, consider only the text included in
            the prompt. "
        "Summarize the hearing terms below. Indicate the names of the
            witnesses heard.\n\n"
        "${TEXT_DOC}"
    ),
    "memorials": Template(
        "You are a judicial advisor specialized in analyzing
            procedural records. "
        "To prepare the response, consider only the text included in
            the prompt. "
        "Summarize the document below, highlighting the legal reasons
            and requests.\n\n"
        "${TEXT_DOC}"
    ),
```

```
    "complaint": Template(
        "You are a judicial advisor specialized in analyzing
            procedural records. "
        "To prepare the response, consider only the text included in
            the prompt. "
        "From the complaint below, indicate the defendants, the
            victims, and the date of the typical event. "
        "Make a summary highlighting the facts, the legal reasons, and
            the requests.\n\n"
        "${TEXT_DOC}"
    ),
    "preliminary_defense": Template(
        "You are a judicial advisor specialized in analyzing
            procedural records. "
        "To prepare the response, consider only the text included in
            the prompt. "
        "Make a summary of the defendant's defense described in the
            document below, highlighting the legal reasons and the
            requests.\n\n"
        "${TEXT_DOC}"
    ),
    "final_arguments": Template(
        "You are a judicial advisor specialized in analyzing
            procedural records. "
        "To prepare the response, consider only the text included in
            the prompt. "
        "From the final allegations below, make a summary of the legal
            reasons. "
        "Also make a summary of the requests.\n\n"
        "${TEXT_DOC}"
    ),
    "other_documents": Template(
        "You are a judicial advisor specialized in analyzing
            procedural records. "
        "To prepare the response, consider only the text included in
            the prompt. "
        "Make a summary of the document below.\n\n"
        "${TEXT_DOC}"
    ),
}
```

Listing A.4: Prompt version 1 (Portuguese)

```
from string import Template

PROMPT_TEMPLATES = {
    # Prompt version 1 (Portuguese)
    "relatorio_sentenca_v1_pt": Template(
        "Você éum assessor judiciário especializado em análise de
            autos processuais. "
        "Para elaboração da resposta considere somente o texto incluso
            no prompt. "
        "A partir dos dados abaixo faça o relatório do ocorrido neste
            processo para o início de uma sentença.\n\n"
        "${TEXTOS_INFERENCIAS_ANTERIORES_CONCATENADOS}"
    ),
}
```

Listing A.5: Prompt version 1 (English)

```
from string import Template

PROMPT_TEMPLATES = {
    # Prompt version 1 (English)
    "judgment_report_v1_en": Template(
        "You are a judicial advisor specialized in analyzing court
            records. "
        "To prepare your answer, consider only the text included in
            the prompt. "
        "Using the data below, write a report of what happened in this
            case to begin a judgment.\n\n"
        "${TEXTS_PREVIOUS_INFERENCES_CONCATENATED}"
    ),
}
```

Listing A.6: Prompt version 2 (Portuguese)

```
from string import Template

PROMPT_TEMPLATES = {
    # Prompt version 2 (Portuguese)
    "relatorio_sentenca_v2_pt": Template(
        "Você éum assessor judiciário especializado em análise de
            autos processuais. "
```

```
        "Como resposta, você deverá redigir um relatório de sentença, que éa parte inicial de uma sentença judicial. "
        "Segundo a Lei, o relatório de sentença deve conter os nomes das partes, a identificação do caso, "
        "com a suma do pedido e da contestação, e o registro das principais ocorrências havidas no andamento do processo. "
        "O formato do relatório de sentença deve seguir o modelo do EXEMPLO a seguir. "
        "<EXEMPLO>Trata-se de ação movida por NOME DA PARTE AUTORA OMITIDA contra NOME DA PARTE RÉ OMITIDA. "
        "A parte autora narra que firmou contrato de seguro com NOME DA PESSOA OMITIDA, no qual foram previstas coberturas para incêndio, "
        "explosão e danos elétricos, entre outros. Em 27/11/2021, ocorreram danos em equipamentos (ar condicionado e camara fria) devido a "
        "quedas e oscilações na rede elétrica. Após a regulamentação do sinistro, a autora indenizou o prejuízo decorrente de danos elétricos "
        "no valor de R$5.150,00, descontando o valor da franquia. A seguradora autora postula da concessionária de energia o ressarcimento do valor pago. "
        "Juntou documentos. A parte ré foi citada, apresentando contestação (evento 13, CONT1). Preliminarmente, arguiu cerceamento de defesa. "
        "Afirmou que não foi oportunizado àcompanhia vistoriar os equipamentos. "
        "Teceu comentários sobre a natureza da responsabilidade e sobre a falta de comprovação de danos materiais, tornando-os inexistentes. "
        "Postulou a improcedência da ação. Juntou documentos. Houve réplica (evento 16, RÉPLICA1). "
        "As partes não pediram a produção de outras provas.</EXEMPLO> "
        "Escreva o relatório de sentença considerando o texto abaixo que descreve o resumo do processo.\n\n"
        "${TEXTOS_INFERENCIAS_ANTERIORES_CONCATENADOS}"
    ),
}
```

Listing A.7: Prompt version 2 (English)

```
from string import Template
```

```
2
3  PROMPT_TEMPLATES = {
4      # Prompt version 2 (English)
5      "judgment_report_v2_en": Template(
6          "You are a judicial advisor specialized in analyzing court
               records. "
7          "As a response, you must write a judgment report, which is the
               initial part of a court judgment. "
8          "According to the Law, the judgment report must contain the
               names of the parties, the identification of the case, "
9          "with the summary of the request and the defense, and the
               record of the main occurrences that occurred during the
               progress of the process. "
10         "The format of the judgment report must follow the model in
               the EXAMPLE below. "
11         "<EXAMPLE>This is a lawsuit filed by NAME OF THE PLAINTIFF
               OMITTED against NAME OF THE DEFENDANT OMITTED. "
12         "The plaintiff states that he entered into an insurance
               contract with NAME OF THE PERSON OMITTED, which provided
               coverage for fire, "
13         "explosion and electrical damage, among others. On 11/27/2021,
               damage occurred to equipment (air conditioning and cold
               storage) due to "
14         "drops and fluctuations in the electrical network. After the
               claim was settled, the plaintiff paid compensation for the
               loss resulting from electrical damage "
15         "in the amount of R$5,150.00, less the deductible. The
               plaintiff insurer requested reimbursement of the amount
               paid from the energy concessionaire. "
16         "Documents were enclosed. The defendant was summoned and filed
               a defense (event 13, CONT1). Preliminarily, the defendant
               argued that the defense was restricted. "
17         "The defendant stated that the company was not given the
               opportunity to inspect the equipment. "
18         "The plaintiff made comments about the nature of the liability
               and the lack of proof of material damage, making it
               non-existent. "
19         "The plaintiff requested the dismissal of the action.
               Documents were enclosed. There was a reply (event 16,
               REPLY1). "
20         "The parties did not request the production of other
```

```
        evidence.</EXAMPLE> "
21      "Write the judgment report considering the text below that
            describes the summary of the case.\n\n"
22      "${TEXTS_PREVIOUS_INFERENCES_CONCATENATED}"
23  ),
24  }
```

A.2 Prompts for LLM as a judge

This appendix presents the LLM-as-a-judge prompt templates employed in the proposed solution. The variables such as {REFERENCE_TEXT} and {TARGET_TEXT} represent placeholders that are dynamically replaced with actual input during inference[2].

All prompt templates are implemented in Python using a structured prompt-template approach. They can be directly used with language model APIs as shown in the following code example:

Listing A.8: LLM-as-a-judge prompt templates usage example

```
1 reference_text = "Text from a real judgment report, written by a
      judge..."
2 target_text = "Text from the judgment report generator..."
3 prompt =
      PROMPT_TEMPLATES["llm_as_judge_en"].substitute(REFERENCE_TEXT=reference_text,
      TARGET_TEXT=target_text)
4 print(prompt)
```

Listing A.9: Prompts for LLM as a judge (Portuguese)

```
1  from string import Template
2
3  PROMPT_TEMPLATES = {
4      # Prompt LLM-as-a-Judge (Portuguese)
5      "llm_as_judge_pt": Template(
6          "Utilizando como base o MQM Framework, avalie os textos
               abaixo, comparando os textos contidos nas tags <TEXTO_ALVO>
               com o <TEXTO_REFERENCIA>. "
7          "Considere os seguintes critérios de avaliação baseados no MQM
               Framework. "
8          "Acurácia: Problemas relacionados a quão bem o conteúdo do
               TEXTO_ALVO representa o conteúdo do TEXTO_REFERENCIA,
```

[2]All prompts are also available on Github: https://github.com/lucianozanuz/Manuscript-Number-JJIMEI-D-25-00093—Supplementary-repository.

```
        considerando a terminologia (se o TEXTO_ALVO contém termos
        legais obrigatórios), a omissão e a inclusão de termos
        importantes ou obrigatórios. "
9       "Verdade: Problemas relacionados a se o TEXTO_ALVO contém
        requisitos do Código do Processo Civil (o TEXTO_ALVO deverá
        conter os nomes das partes, a identificação do caso, com a
        suma do pedido e da contestação e o registro das principais
        ocorrências havidas no andamento do processo; o TEXTO_ALVO
        precisa conter o nomes das partes, o tipo de ação, os
        pedidos da inicial, os pedidos da contestação e sobre as
        principais ocorrências; por exemplo, no Procedimento Comum
        Civil deve conter as audiências e os laudos periciais). "
10      "Fluência: Problemas ao estilo (o TEXTO_ALVO éformal, mas
        segue os princípios da linguagem simples), ortografia (o
        TEXTO_ALVO possui ortografia correta), gramática (o
        TEXTO_ALVO possui gramática correta), violações locais (o
        formato de datas, valores, números, telefones e endereços
        do TEXTO_ALVO segue o padrão local do Brasil). "
11      "Retorne apenas o resumo da análise da seguinte forma, em
        formato JSON. "
12      "Acurácia: 5 se contém problemas graves de Acurácia (Exemplo:
        TEXTO_ALVO não se refere ao mesmo processo judicial
        relatado no TEXTO_REFERENCIA); 3 se contém problemas leves
        de Acurácia (Exemplo: TEXTO_ALVO se refere ao mesmo
        processo judicial relatado no TEXTO_REFERENCIA, mas, omite
        termos legais obrigatórios ou inclui termos inadequados
        para um relatório de sentença); 0 se não contém problemas
        de Acurácia (Exemplo: TEXTO_ALVO se refere ao mesmo
        processo judicial relatado no TEXTO_REFERENCIA, não omite
        termos legais obrigatórios e não inclui termos inadequados
        para um relatório de sentença). "
13      "Verdade: 5 se contém problemas graves de Verdade (Exemplo:
        TEXTO_ALVO não contém o nomes das partes, o tipo de ação e
        os pedidos da inicial e da contestação); 3 se contém
        problemas leves de Verdade (Exemplo: TEXTO_ALVO contém o
        nomes das partes, o tipo de ação e os pedidos da inicial,
        mas não descreve adequadamente as principais ocorrências
        como audiências e laudos periciais, se existirem no
        TEXTO_REFERENCIA); 0 se não contém problemas de Verdade
        (Exemplo: TEXTO_ALVO contém o nomes das partes, o tipo de
        ação e os pedidos da inicial, e descreve adequadamente as
```

```
        principais ocorrências como audiências e laudos periciais,
        se existirem no TEXTO_REFERENCIA). "
14      "Fluência: 5 se contém problemas graves de Fluência (Exemplo:
        TEXTO_ALVO contém interrupções, como divisão em seções ou
        subtítulos); 3 se contém problemas leves de Fluência
        (Exemplo: TEXTO_ALVO não contém interrupções, como divisão
        em seções ou subtítulos, mas não segue os princípios da
        linguagem simples ou possui erros de ortografia ou
        gramática ou o formato de datas, valores, números,
        telefones e endereços não segue o padrão local do Brasil);
        0 se não contém problemas de Fluência (Exemplo: TEXTO_ALVO
        não contém interrupções, como divisão em seções ou
        subtítulos, segue os princípios da linguagem simples e não
        possui erros de ortografia e gramática e o formato de
        datas, valores, números, telefones e endereços segue o
        padrão local do Brasil).\n"
15      "<TEXTO_REFERENCIA>${TEXTO_REFERENCIA}<\TEXTO_REFERENCIA>\n"
16      "<TEXTO_ALVO>${TEXTO_ALVO}<\TEXTO_ALVO>"
17  ),
18 }
```

Listing A.10: Prompts for LLM as a judge (English)

```
1  from string import Template
2
3  PROMPT_TEMPLATES = {
4      # Prompt LLM-as-a-Judge (English)
5      "llm_as_judge_en": Template(
6          "Using the MQM Framework as a basis, evaluate the texts below,
           comparing the texts contained in the tags <TARGET_TEXT>
           with the <REFERENCE_TEXT>. "
7          "Consider the following evaluation criteria based on the MQM
           Framework. "
8          "Accuracy: Issues related to how well the content of the
           TARGET_TEXT represents the content of the REFERENCE_TEXT,
           considering the terminology (whether the TARGET_TEXT
           contains mandatory legal terms), the omission and inclusion
           of important or mandatory terms. "
9          "Truth: Problems related to whether the TARGET TEXT meets the
           requirements of the Code of Civil Procedure (the TARGET
           TEXT must contain the names of the parties, the
           identification of the case, with the summary of the request
```

```
        and the defense and the record of the main occurrences that
        occurred during the progress of the process; the TARGET
        TEXT must contain the names of the parties, the type of
        action, the initial requests, the defense requests and the
        main occurrences; for example, in the Code of Civil
        Procedure it must contain the hearings and the expert
        reports). "
10      "Fluency: Problems with style (the TARGET TEXT is formal, but
        follows the principles of plain language), spelling (the
        TARGET TEXT has correct spelling), grammar (the TARGET TEXT
        has correct grammar), local violations (the format of
        dates, values, numbers, telephones and addresses of the
        TARGET TEXT follows the local standard of Brazil). "
11      "Return only the summary of the analysis as follows, in JSON
        format. "
12      "Accuracy: 5 if it contains serious Accuracy problems
        (Example: TARGET_TEXT does not refer to the same lawsuit
        reported in REFERENCE_TEXT); 3 if it contains minor
        Accuracy problems (Example: TARGET_TEXT refers to the same
        lawsuit reported in REFERENCE_TEXT, but omits mandatory
        legal terms or includes terms inappropriate for a judgment
        report); 0 if it contains no Accuracy problems (Example:
        TARGET_TEXT refers to the same lawsuit reported in
        REFERENCE_TEXT, does not omit mandatory legal terms and
        does not include terms inappropriate for a judgment
        report). "
13      "Truth: 5 if it contains serious Truth problems (Example:
        TARGET_TEXT does not contain the names of the parties, the
        type of action and the claims in the initial and defense);
        3 if it contains minor Truth problems (Example: TARGET_TEXT
        contains the names of the parties, the type of action and
        the initial requests, but does not adequately describe the
        main events such as hearings and expert reports, if they
        exist in REFERENCE_TEXT); 0 if it does not contain Truth
        problems (Example: TARGET_TEXT contains the names of the
        parties, the type of action and the initial requests, and
        adequately describes the main events such as hearings and
        expert reports, if they exist in REFERENCE_TEXT). "
14      "Fluency: 5 if it contains serious Fluency problems (Example:
        TARGET_TEXT contains interruptions, such as division into
        sections or subtitles); 3 if it contains minor Fluency
```

```
        problems (Example: TARGET_TEXT does not contain
        interruptions, such as division into sections or subtitles,
        but does not follow the principles of plain language or has
        spelling or grammar errors or the format of dates, amounts,
        numbers, telephones and addresses does not follow the local
        standard in Brazil); 0 if it does not contain Fluency
        problems (Example: TARGET_TEXT does not contain
        interruptions, such as division into sections or subtitles,
        follows the principles of simple language and does not have
        spelling and grammar errors and the format of dates,
        values, numbers, telephones and addresses follows the local
        standard of Brazil).\n"
      "<REFERENCE_TEXT>${REFERENCE_TEXT}<\REFERENCE_TEXT>\n"
      "<TARGET_TEXT>${TARGET_TEXT}<\TARGET_TEXT>"
  ),
}
```

ANNEX A PUBLISHED ARTICLES

Zanuz, L., Rigo, S.J. (2022). **Fostering Judiciary Applications with New Fine-Tuned Models for Legal Named Entity Recognition in Portuguese**. In: Pinheiro, V., et al. Computational Processing of the Portuguese Language. PROPOR 2022. Lecture Notes in Computer Science(), vol 13208. Springer, Cham. https://doi.org/10.1007/978-3-030-98305-5_21

Zanuz, L., Rigo, S.J. (2025). **Transforming the Legal Landscape: an AI-Driven Framework for Judicial Text Processing**. RECIMA21 - Revista Científica Multidisciplinar - ISSN 2675-6218, [S. l.], v. 6, n. 5, p. e656426, 2025. https://doi.org/10.47820/recima21.v6i5.6426.

Zanuz, L., Rigo, S.J., et al. (2025). **Automated Judgment Report Generation in Brazilian Legal Proceedings Using Generative AI**. International Journal of Information Management Data Insights. In Major Revision.