Alma Mater Studiorum · Università di Bologna

SCUOLA DI SCIENZE
Corso di Laurea in Informatica

An analysis of the current limitation and future directions of AI applied to the legal domain based on a SLR and early prototyping

Relatore:
Chiar.mo prof.
Giancarlo Succi

Presentata da:
Alessandro Ravveduto

IV Sessione
Anno Accademico 2023/2024

Contents

Chapter 1

Introduction

The objective of the thesis is to conduct an analysis of the current state of artificial intelligence (AI) applied to the legal domain, examining its limitations and improvements over the years. Furthermore, attention will be dedicated to the most promising directions for future enhancements. At the end, we will present a prototype, a proof of concept, for handling the process of resolution between conflicting laws.
As highlighted in multiple papers, the legal domain poses unique challenges for AI applications. It demands advanced reasoning capabilities, the ability to interpret complex language structures, and precise decision-making based on legal precedents and contextual understanding. The nature of legal texts demands robust and adaptable models capable of capturing both semantic nuances and logical structures. Working in the legal domain could drive advances in deep learning techniques due to the difficulty faced and be a stimulating challenge for researchers. In particular, we will focus on the topics of legal reasoning and information retrieval. Regarding the latter, we will also consider its subdomains, such as information extraction and legal information retrieval.

Chapter 2

Prerequisites

2.1 About SLR

A Systematic Literature Review (SLR) is a detailed analysis carried out on a topic. Usually, it involves the use of different papers in order to tackle down a problem and identify different shades of it.
The main task of an SLR is to examine an argument going through different phases. In particular, as shown in figure 3.1, the three main phases are:

1. Plan review;
2. Conduct review;
3. Document review.

Therefore, an SLR has not the objective of resolving a problem and does not have to come up with new solutions, even though it can suggest the most promising ways. In conclusion, in chapter 3 we present more details on the SLR and what we did to analyze the current limitation and future directions of AI applied in the legal domain.

2.2 About the use of AI in Legal

The application of AI in the legal domain has become a reality that cannot be ignored anymore. Legal professionals are gradually using several types of AI, data analytics tools and smart virtual assistants to optimize their work, reducing time spent for

time-consuming tasks. As reported in [1] the development of AI models has seen an increase in interest lately and they are designed to accomplish several tasks, for example: classification of documents, application of complex regulations, suggestion or prediction of the outcome of cases, detection or anticipation of illegal behavior, evaluation of legal evidence, analysis of sets of legal cases.

2.3 LLM

Large Language Models (LLMs) are advanced neural networks based on deep learning techniques. This type of model is characterized by the application of transformers, introduced in [18], where the neural network can process entire sequences of data simultaneously. The main innovation is the self-attention mechanism which allows to connect tokens together in order to uncover their meaning. Transformers typically consist of stacked encoder and decoder layers; the encoder builds context-aware representations of the input, while the decoder generates output conditioned on these representations and previously generated tokens.

2.4 Information retrieval

Information retrieval (IR) has been a field of study since the 1950s, and with the advent of web searches in the 1990s its interest has experienced a significant increase. In general, we can define IR as finding resources that satisfy an information need from a large collection. Specifically, we can assert that IR is based on unstructured documents, namely texts in natural language with a partial structure (e.g. title and paragraph). For this reason IR systems differ from Data Retrieval Systems (e.g. a DBMS) which use database schema. Consequently, another difference between IR and DBMS is how they search for information. In particular, IR systems retrieve document by a set of keywords in natural language while a DBMS uses a query language, such as SQL and relational algebra, based on a formal grammar. Considering the ambiguity of natural language, a relevance notion is needed instead of exact matching, and its an important part of the foundation for a retrieval model.
As described in [5], an IR system retrieves information in two phases:

1. Retrieval phase: from a large collection of documents, an initial set of relevant documents is retrieved. There are different types of retrieval techniques, such

[1] Based on works of Scientific Unit Director Prof. A. Rotolo Scientific and executive representatives and Dott.ssa C. Valentini, Dott. G. Contissa `https://centri.unibo.it/alma-ai/en/scientific-units/ai-for-law-and-governance` and The British Institute of International and Comparative Law `https://www.biicl.org/documents/170_use_of_artificial_intelligence_in_legal_practice_final.pdf`

as:

- Conventional retrieval methods: these are based on traditional term-matching methods. Query augmentation, document augmentation and lexical dependency are examples of this category;
- Sparse retrieval methods: these use sparse vectors where only a few elements are nonzero. The two main ways are neural weighting schemes and sparse representation learning;
- Dense retrieval methods: the conventional design for these models adopt the dual-encoder architecture;
- Hybrid retrieval methods: they combine different representations, architectures and techniques in order to benefit from of the strengths of diverse approaches, such as word embeddings, contextualized representations, attention mechanisms and traditional ranking algorithms.

2. Ranking phase: the retrieved documents are re-ranked using sophisticated ranking models to improve accuracy and provide high-quality search results. There are two large families:

   - Learning to rank (LTR): comprises methods that are strictly dependent on manually created features and that pay attention to statistical attributes like document lengths and term frequencies.
   - Deep learning based ranking models: they use neural networks to capture semantic relationships between queries and documents. In this group are present attention based models which gained popularity lately.

2.4.1 Information extraction

Information extraction (IE) goes a step further than IR. Instead of just retrieving documents, it analyzes them in order to extract specific pieces of information needed to answer the query. Thus, the IE's output is structured data (e.g. names, dates, etc.), unlike IR which output is a list of resources that match the query. The variety of possible applications for IE are endless, it can range from a web bot as assistance for an online service to a system for retrieving data in several areas like finance, healthcare, scientific and many others.

2.4.2 Legal information retrieval

Considering the significant amount of Electronically Stored Information (ESI), the importance of a system capable of retrieving information has increased drastically.

In particular, a growing field of legal computer science is Legal Information Retrieval (LIR), it is the application of IR to legal texts, including legislation and case law. As stated in [14], LIR relies on both quantitative and qualitative information and a system of that type must satisfy the following features: volume (in terms of number of documents), document's size, the structure and heterogeneity of each type of document, legal hierarchy, temporal aspects, importance of quotations, and many others.
The main characteristics a LIR system should own:

1. Semantic understanding
2. Robustness with respect to different inputs
3. Robustness to the varying corpus of documents
4. Robustness as the number of documents increases
5. Robustness to input errors
6. Sensitivity to context

Furthermore, it should obtain the intended information according to the semantic meaning of the documents gathered, independently of the different ways in which a user might describe their information needs.

2.5 Ontology

The term "ontology" refers to the discipline dedicated to the study and reasoning about beings and their properties. In particular, it defines entities and relationships between them and the environment, creating a representation of a specific domain. An ontology allows to share knowledge between systems and people, making it useful for interoperability problems where heterogeneous parties are involved. One of the most evident uses of ontology is the semantic web. A large ontology has several drawbacks: frequent errors in a long chain of subsumptions, hierarchy errors, and omission of relationships. In conclusion, an ontology is an abstract model in which each class (a type) could be linked to one another through a relationship, and each class owns different attributes.
The knowledge graph (KG) is an instance of an ontology. The latter one represents the backbone, the general structure, which the individuals from a dataset will follow in order to create an actual representation of the given world. The Resource Description Framework (RDF) can be used to create ontologies and to represent knowledge in a graph model. RDF is a framework for modeling and exchanging data on the web.

It provides a set of specifications for representing data in the form of triples, which consist of a subject, a predicate and an object. Instead, Web Ontology Language (OWL) is a language designed for use by applications that need to process the content of information, not for the only purpose of presenting information to humans, so it allows to create ontologies on the web. It is based on RDF and provides a more expressive way of describing concepts and relationships by providing additional vocabulary along with a formal semantics, for example, it includes a set of constructs for representing classes, properties, and relationships between classes. Thus, OWL facilitates machine interpretability of web content.[2] Also, to interrogate the KG, the SPARQL language is used.

[2]Based on `https://graph.build/resources/ontology` and `https://www.w3.org/TR/owl-features/`

Chapter 3

Research method

In this chapter, we describe the research methodology employed to ensure a rigorous, systematic and unbiased evaluation of the literature in the field. The Systematic Literature Review (SLR) follows overall the process proposed by [9], [10], [1] in order to ensure that the research is comprehensive, credible and reproducible. The process is visible in figure 3.1.

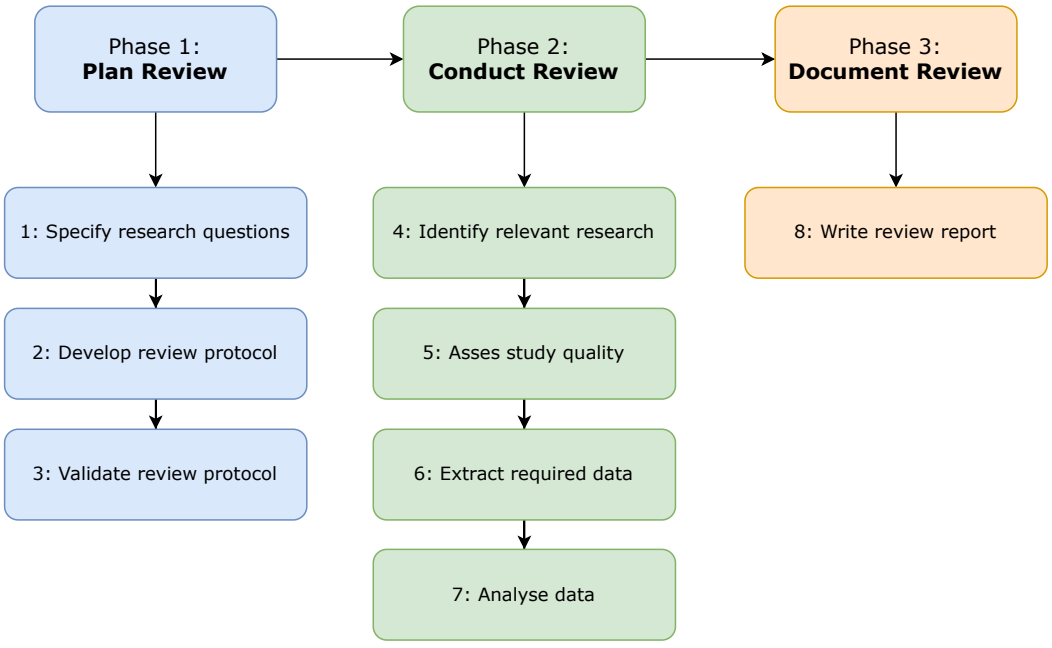

Figure 3.1: SLR Process (adapted from figure 1 in section *2. Systematic literature review* of [1])

3.1 Related works

In [14] is provided an overview about the state-of-the-art of artificial intelligence approaches for legal domain, focusing on Legal Information Retrieval systems using Natural Language Processing (NLP), Machine Learning (ML) and Knowledge Extraction (KE) techniques. The research is dated 2022, so it does not take into account the new discoveries in technologies and strategies that improve the problem of Information Retrieval in the legal field.

3.2 Research questions

One of the most critical steps in conducting a SLR is formulating a clear research question (or a set of questions) that will guide and shape the review process. To set the direction for our SLR, we first establish its main objective by taking into account the following key elements:

- **Purpose:** analyze and characterize
- **Viewpoint:** software developers
- **Issue:** reasoning capabilities
- **Object:** understanding systems for reasoning within the legal domain

These elements lead us to the following research questions (RQ).

- **RQ1** What are the current limitations of LLMs in performing logical reasoning tasks in the context of legal documents ?
- **RQ2** What are the current limits of LLMs in performing information retrieval in legal documents, and how to improve them ?
  - **RQ2.1** How can semantic understanding and context awareness be improved in legal text retrieval systems ?
- **RQ3** What benchmarks and datasets are available for evaluating the logical reasoning and information retrieval capabilities of LLMs?
  - **RQ3.1** How effective are these benchmarks in capturing the complexities of real-world legal reasoning and information retrieval?
- **RQ4** How do logical inconsistencies affect reasoning and information retrieval in the legal domain, and how are these conflicts identified and handled by the systems ?

- **RQ4.1** How are conflicts and ambiguities traditionally resolved by legal practitioners ?
- **RQ4.2** Which techniques can be used to detect and resolve such logical inconsistencies ?

The first research question RQ1 focuses on identifying the limitations LLMs encounter when performing logical reasoning tasks in the legal domain. This question also highlights broader challenges in reasoning tasks beyond the legal context. RQ2 delves specifically into information retrieval, with a sub-question aimed at exploring potential improvements in this area. RQ3 examines existing datasets that are used to evaluate the reasoning capabilities of LLMs, while RQ3.1 assesses whether these datasets effectively capture the complexities of real-world legal reasoning and information retrieval. Finally, RQ4 addresses the issue of logical inconsistencies in the legal domain and investigates how these conflicts are identified and managed, particularly by legal practitioners, and the techniques used to resolve them.

3.3 Study selection

In this section, we will describe the criteria used for including and excluding papers, in particular, using some requirements to filter all the possible results obtained during the search phase. Then, we will use some criteria to rank them and determine the average quality of the works we will examine.

3.3.1 Inclusion and exclusion criteria

In order to narrow down papers relevant to our research questions we defined a set of Exclusion Criteria (EC) as well as Inclusion Criteria (IC). A paper has to fulfill all the IC and not meet any of the EC to be included in our review.

- **IC1:** the work is written in English
- **IC2:** the paper is related to artificial intelligence and reasoning capabilities in the legal domain
- **IC3:** the paper answers at least at one of the research questions
- **IC4:** the work is published in IEEE, Scopus or Web of Science
- **IC5:** the work addresses a software development problem
- **IC6:** the paper contains a benchmark or some form of evaluation

The exclusion criteria that we used in this SLR are as follows:

- **EC1:** the work is not written in English
- **EC2:** the paper does not answer on any of the research questions
- **EC3:** the work did not satisfy one or more of the inclusion criteria stated above
- **EC4:** the work is similar to others produced later by the same authors
- **EC5:** the work can be classified as grey literature
- **EC6:** the paper has not open access

3.3.2 Quality criteria

To ensure the quality of the studies, following the process proposed in [2] we prepared a list of questions with relative scores (0, 0.5, 1) which could be used as a reliable indicator of the quality of the reviewed paper. These questions are:

1. Is there a clear statement of the aims of the research?
    - 1 point if the motivation of the research was clearly stated;
    - 0.5 points if the motivation was provided, but could be further elaborated;
    - 0 points if the motivation was hard to identify or if it was not mentioned.
2. Were there any major issues or limitations mentioned in the authors' research process and results that could affect the effectiveness of the system's reasoning capabilities?
    - 1 point if the author did not mention any difficulty in the research process;
    - 0.5 points if minor issues were encountered during the research process;
    - 0 points if the the author mentioned some significant complication that affected the research process.
3. Does the study provide concrete experiments of the provided system on some valuable dataset?
    - 1 point if the dataset is a commonly used dataset for deductive logical reasoning;
    - 0.5 points if the benchmark was provided, but it is not clear the scenario of evaluation;

- 0 points if the system was not tested.

4. Was the proposed reasoning system objectively evaluated?

   - 1 point if the authors conducted a fair and unbiased review of their system or if they performed a critical analysis of its results;
   - 0.5 points if the authors performed an analysis of their system but such an analysis was partially biased or is not clear or critical enough;
   - 0 points if the authors did not conduct a fair and unbiased analysis or if the results were not critically analysed.

3.4 Selection of sources

A critical component in conducting an SLR is the creation of an effective search strategy. Due to the large number of papers available on the topic, it is unfeasible to review them all, even though this may result in missing some relevant insights. Consequently, we prioritized several major databases of scientific and technological literature: IEEE Xplore, Scopus and Web of Science (WoS).
The query strings used are:

- **IEEE Xplore** - (161 papers initially)

  ("All Metadata":Logical reasoning) AND ("All Metadata":Law) AND ("All Metadata":Information retrieval) OR ("All Metadata":Logical reasoning in Law) considering only conferences, journals and books published by IEEE during: 1997-2025.

- **Scopus** - (347 papers initially)

  (TITLE-ABS-KEY (logical AND reasoning) AND TITLE-ABS-KEY (law) AND TITLE-ABS-KEY (information AND retrieval) OR TITLE-ABS-KEY (logical AND reasoning AND in AND law)) AND PUBYEAR ¿ 1996 AND PUBYEAR ¡ 2025 AND (LIMIT-TO (DOCTYPE , "ar") OR LIMIT-TO (DOCTYPE , "cp") OR LIMIT-TO (DOCTYPE , "ch") OR LIMIT-TO (DOCTYPE , "bk")) AND (LIMIT-TO (LANGUAGE , "English")) AND (LIMIT-TO (SRCTYPE , "j") OR LIMIT-TO (SRCTYPE , "p") OR LIMIT-TO (SRCTYPE , "b") OR LIMIT-TO (SRCTYPE , "k")) AND (LIMIT-TO (PUBSTAGE , "final"))

- **Web of Science** - (332 papers initially)

  (((ALL=(information retrival)) AND ALL=(Law)) AND ALL=(logical reasoning)) OR ALL=(logical reasoning in law) and Article or Book Chapters or

Proceeding Paper or Review Article (Document Types) and English (Languages) and Law or Philosophy or Computer Science Artificial Intelligence or Logic or Computer Science Theory Methods or Computer Science Interdisciplinary Applications or Computer Science Information Systems or Computer Science Software Engineering or Mathematics Applied or Mathematics or Language Linguistics or Linguistics (Web of Science Categories)

The results of the gathering and filtering phase can be seen at `https://docs.google.com/spreadsheets/d/12BLsRam-hAUbYWtYBylsrsolkFZbAAkWPv5U9smuCrc/edit?usp=sharing`

3.4.1 Other sources

We observed that some relevant papers were not included in the databases mentioned above, leading us to incorporate arXiv into our review, which is an open-access repository of electronic preprints and postprints. The primary limitation of using it is that the papers are not peer-reviewed, raising concerns about the reliability and rigor of the research. In order to make up for this limitation the papers will undergo the inclusion and exclusion criteria, then will be evaluated using the quality criteria described before. However, arXiv offers several advantages: the submission process is faster, enabling researchers to remain current with the latest advancements in their field. Additionally, it is increasingly common for researchers to publish on arXiv, as it enhances the visibility of their work. In particular, the papers included in this SLR that were selected from the other sources are:

- [4] from arXiv
- [16] from ACL anthology

3.4.2 Searching results

In this section, we presents the results of our searching phase by following the sequential stages outlined in the review process, as illustrated in Figure 3.2.

1. Stage of Identification: This phase involves the initial gathering of potential studies through systematic database searches. It employs well-defined search strategies using Boolean operators and keywords, aimed at maximizing the amount of relevant publications. In this stage the total number of collected papers were 848.
2. Stage of Screening: Following identification, the process transitions to the screening phase, where titles and abstracts are scrutinized against predefined

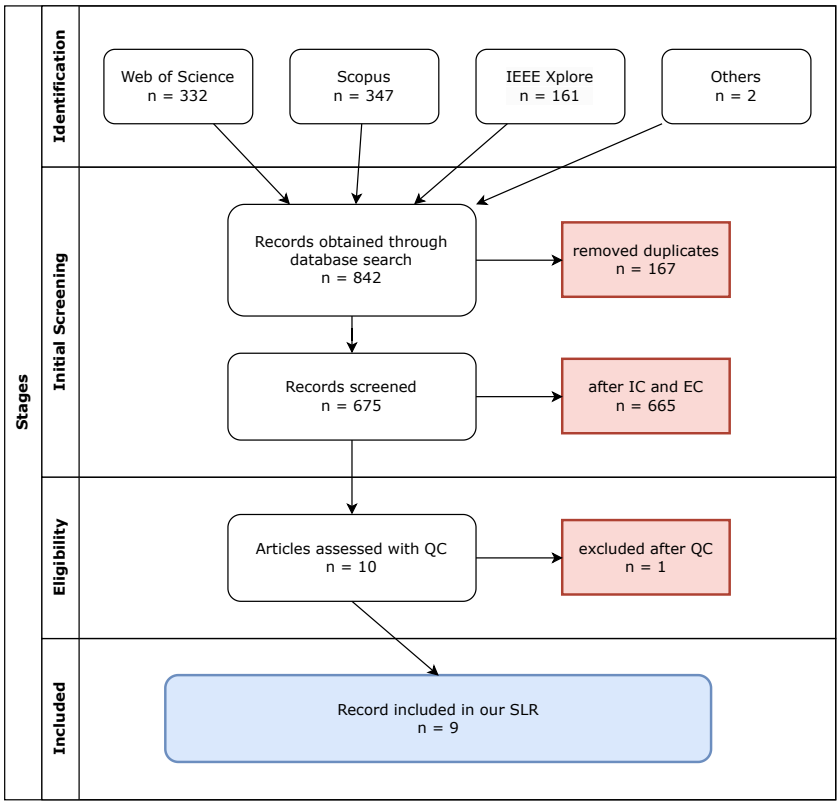

Figure 3.2: SLR Phases and Outcomes of the Review Process (PRISMA-like))

inclusion and exclusion criteria. This step ensures that only studies pertinent to the research question are retained for further evaluation. After this stage, the papers removed were 832.

3. Stage of Eligibility: In this critical stage, full-text articles are reviewed to ascertain their eligibility for inclusion. The flowchart conveys the methodology of assessing each study against rigorous eligibility criteria, detailing reasons for exclusion, thereby enhancing transparency and reproducibility of the review process.

4. Stage of Inclusion: The final stage culminates in the inclusion of studies that satisfy all criteria, thus forming the basis for data extraction and synthesis. The total number of studies included in the final analysis are 9.

3.4.3 Analyze results

We can look deeper at those selected papers analyzing the frequency and the number of their sources. From figures 3.3, 3.4 we can observe that almost half of the collected papers are from Scopus while for the other sources only one paper was used for each.

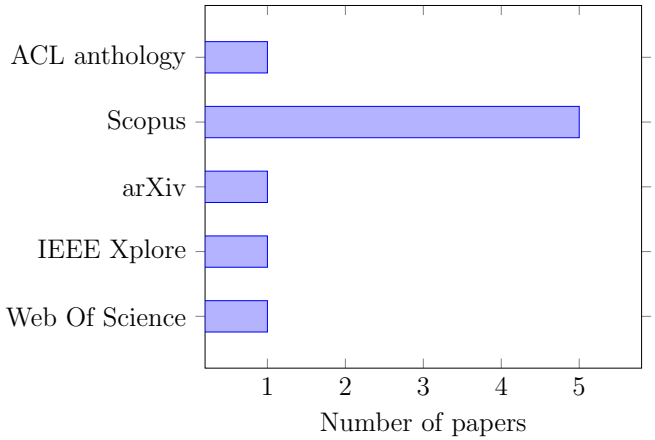

Figure 3.3: Absolute number of occurrences of papers relative to their sources

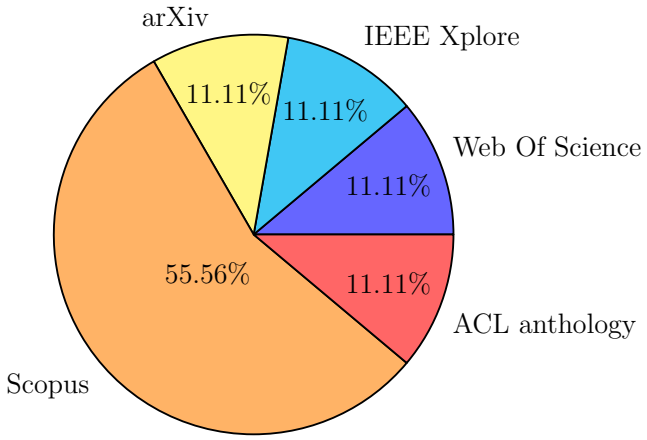

Figure 3.4: Papers distribution by database

Chapter 4

Analysis of the results

4.1 Discussion

As mentioned previously, we gathered a total of 842 studies from different sources, then we screened them with multiple phases until achieving a total of 9 papers. In particular, section 3.4.3 Analyze results shows the number and ratio of papers per source. Moreover, in Appendix A Summary of the consulted papers a more detailed examination of each publication collected and used is presented. In this section, we will discuss the research question proposed in 3.2 Research questions.

RQ1: What are the current limitations of LLMs in performing logical reasoning tasks in the context of legal documents ?

Logical reasoning is an hard task that requires complex abilities, especially understanding in a given context both explicit and implicit facts, extrapolating new information and deriving conclusions by following logical steps. In light of this, as reported in [20], such a challenge should not be treated as a traditional reading comprehension problem. An interesting approach for legal reasoning is the further decomposition of this task in sub-problems like in [4], precisely: issue spotting, rule recall, rule application, rule conclusion, interpretation, and rhetorical understanding. A primary obstacle highlighted in papers [19, 16] is that the pre-trained models like BERT and RoBERTa show strong results in semantic matching. Since they rely mainly on contextual semantics, they do not capture the symbolic logic necessary for inference in logical reasoning tasks. Those are present in the LR-LSAT dataset introduced in [19] - the various datasets will be discussed in more detail in RQ3. A key component for a reasonable argumentation is logical consistency. In particular, [11] brings attention to the inadequateness of GPT models in regards to the quality of symbolic logic reasoning, the inability to handle negation effectively, and the

strictly dependence on superficial patterns. A new approach proposed in several works like [6] and many others, consists in using LLMs for parsing the sentences of a given text in First Order Logic (FOL) expression. This is done in order to work with them and enhance the logical reasoning process. The bottleneck of this method is acknowledge by the parsing ability of the model, since wrong expressions can compromise the entire reasoning process. Thus, as noted by Wang et al. [19] in section *C. Results and Analysis*, a significant challenge for logical reasoning is to automatically extract the logical elementary units, and then identify the logical relationships between them in an unsupervised manner. In [3] is underlined how Deep Neural Networks (DNNs) based models are data thirsty, thus making their training more difficult in low resources environment, for instance when electronic files cannot provide some information due to their confidentiality. As mentioned in the previous work and in [16], another challenge for most models nowadays is the lack of explicability, which is crucial for legal practitioners who have to understand why a system took some decision instead of others, in order to apply that result in a real life scenario. This brings the professionals of the legal field to not tolerate the concept of "black box" for justifying an answer from a "smart" system and to shift their attention on models which provide an explanation of their reasoning steps.

RQ2: What are the current limits of LLMs in performing information retrieval in legal documents, and how to improve them ?

2.4 Information retrieval describes the task of Information Retrieval (IR) and its definition. Based on that and on the observation in [16], we can assert that a specialization of an IR task applied in the legal domain is Legal Case Retrieval (LCR), which retrieves relevant cases from a query. Moreover, another subclass of IR is Information Extraction (IE), which has some differences - in 2.4.1 is reported in greater depth the explanation of IE and its distinctions from IR.
In section *3.1 Information Extraction from Legal Texts* of [8] are described the three main methods for IE from legal texts, namely:

- Rule-based information extraction method: the key idea of this approach is to define a domain ontology or a large number of structured rules tailored to the features of legal texts. It is clear that the biggest disadvantage of this method is its low generalization capability. An example of this is present in [6] where the SARA ontology's limited expressiveness affects the ability to retrieve temporal and event-specific information;

- Similar-case-based information extraction method: the fundamental concept relies on the definition of similar-case rules to measure similarity (early models

used euclidean distance). This method is very effective but lacks of explanaibility;

- Machine-learning-based information extraction method: this technique can be split up in two groups. In particular: End-to-end deep-learning and Machine-learning methods, the first one uses directly deep neural networks where there is no explicit information-extraction layer. The second one, extracts the structure of text-information according to the legal elements, then through a knowledge graph, labels them and constructs the supervised-learning-based model. Nevertheless, both methods have poor interpretability for the judicial-trial process.

The study [6] finds, in section *5.2 - Error analysis*, that LLMs struggle with the extraction of legal facts due to the lack of arguments, having high precision but low recall and occasional over predictions. Potential solutions are: adopting a richer ontology and enhancing training datasets to ensure that extracted KBEs align more accurately with the case description. Translating legal cases into structured formats automatically, such as Prolog, could facilitate the scalability and the application of AI in real-world legal scenarios.
Reading comprehension (RC) is the ability to read a text and fully understand the meaning of it, identifying relevant information. If we add the assignment to retrieve some specific data, we have the IE task. For this reason, we will take in account the section *VI. READING COMPREHENSION* of [19]. Although the proposed model P-DUMA - more details can be found in RQ2.1 - achieved outstanding performance with a score that could allow acceptance by the top 30 law schools, the researchers, through error analysis, have underlined some major problems:

- The length of the context prevents effective encoding, due to the truncation of essential information for answer prediction;
- Some comparative questions require comparative learning abilities that have not yet been taken into account;
- The shortage of common sense knowledge.

The authors JIN and HE in [8], remind that in a judicial trial, the primary data source to aid in sentence decision making are electronic files, which include electronic documents and related electronic data produced before or during the case acceptance process. In addition, a judgment document is a legal document in which the process and result of the case's trial are logged. The judgment documents are more complete and provide more information; consequently, they are useful for training models. Specifically, in section *4.1 Legal-Fact Extraction and Verification from Electronic Case Files* is shown both how the basic unit of information in legal texts is the legal

fact, and how fundamental it is. This approach could bring better performance in improving context-awareness and accuracy of information retrieval systems but challenges remain, including data privacy constraints, lack of structured training datasets, and complexity of mapping extracted facts to relevant statutes.
The framework NS-LCR introduced in [16], supports the baseline model by providing three external modules, in particular the law-level modules use the law in FOL format, enhancing the comprehension between relationships. An obvious limitation is the manual extraction of the predicates from each law.
To push beyond the current boundaries, the most compelling approaches directions are the application of logic and the semantic context enhancement.

RQ2.1 How can semantic understanding and context awareness be improved in legal text retrieval systems ?

Understanding the context allow us to interact in certain ways and make different choices. At the base of every interaction with the environment around us, there is the comprehension and interpretation of what is happening. The improvement in semantic understanding by legal text retrieval systems can be a game changer for the LIR task. As human beings, we have an innate knowledge of the concept of similarity and we are able to develop it with experience. This does not apply to machines, therefore it is mandatory to explore this area of research.
In [6], the focus is on the extraction of events and their arguments, translating semantic content into a structured format. In the paragraph on Semantic representations in *2. Related work*, the paper has reported several formalisms with that purpose. It also states that previous work have relied on ad-hoc ontologies for semantic representation, depending on the application domain. For a better semantic understanding, the authors of the paper proposed a representation of entities and events with a direct reference to the text of the case description using structured span objects $span(Value, Start_{index}, End_{index})$ where $Value$ is a string or integer whose purpose is to represent the event or person and serves also the Prolog program; $Start_{index}$ and End_{index} provides an anchor in the text for the entity or event. Furthermore, they made the ontology more consistent. The events which are instantaneous, from the perspective of statutory reasoning like birth, death and income, are allowed to have a start argument but no end. In general, a more expressive ontology could allow a straightforward translation between language and logical form. Also, the usage of explicit KBEs helps with the interpretability of the model because it makes evident which facts are used to perform statutory reasoning.
In [19] is presented a system for complex reasoning, it faces three tasks of LSAT and for each one proposes a unique strategy, in particular:

- For analytical reasoning combines a symbolic, neural and neural-symbolic models;

- For logical reasoning uses a neural-symbolic model called LReasoner which utilize a logic-driven context extension framework and data augmentation algorithm. It was introduced in the previous paper [20];
- For reading comprehension they used the P-DUMA module, a neural model based with an additional Dual Multi-head Co-Attention module between the pre-trained encoder model and the classification layer.

A great example of how to improve the context awareness is the LReasoner model, it extracts the elements from the context as logical symbols and identifies the logical relationships between them creating logical expression that will be used for inferring new ones and choosing the option that is closest to the answer.
Similarly, NS-LCR in [16], is a neural-symbolic framework that leverages the power of FOL over direct embedding encoding and combines the output of different elements to provide the final answer, prioritizing the neural retrieval module when the prediction has a high confidence. Otherwise, the symbolic module has a higher priority to have a better accuracy.
In conclusion, we highlight that ontologies and knowledge graphs 2.5, are really useful tools in information retrieval systems. For instance, in [6] the extraction of information is guided by the ontology. In section *4 Model*, they defined the problem of IE as building a tree of depth 2 or less where each node is a span in the case description and each edge has a label with an element from the ontology.

RQ3 What benchmarks and datasets are available for evaluating the logical reasoning and information retrieval capabilities of LLMs?

As outlined in section *A. Challenges in Logical Reasoning* of [19], at the time of the study (2022) several logical reasoning benchmarks were introduced, such as ReClor and LogiQA, which [20] uses for conducting experiments. The first one is built upon standardized exams including GMAT (Graduate Management Admission Test) and LSAT (Law School Admission Test), the second one comes from the National Civil Servants Examination of China and is professionally translated into an English version. The LSAT dataset used in [19] is composed of 90 exams, each of them contains 100 questions, of which half relates to logical reasoning, the remaining part is divided into reading comprehension and analytical reasoning questions. For reading comprehension, many datasets have been developed and studied, for instance: SQuAD, MCTest and RACE even though the RC-LSAT [19] is more challenging due to its longer sequences and difficulty to grasp.
An exhaustive study conducted by Guha et al. [4] presents a collaboratively constructed legal reasoning benchmark consisting of 162 tasks that cover six different types of legal reasoning. These are: issue-spotting, rule-recall, rule-application, rule-conclusion, interpretation and rhetorical-understanding. As reported in *4.2*

Dimensions of variation, each task in LegalBench contains a number of samples that range from a minimum of 50 and has an average size of 563. It is worth to mention that, as seen in *Table 13: Task Statistics*, both the number of samples and the mean sample length (in words) differs considerably between tasks, for instance the "privacy_policy_qa" task has the highest number of samples (10931) but the mean sample length is just 41.1, instead the "sara_numeric" task has 100 samples with the biggest average length of 12222.1 words. The LegalBench repository[1] contains all the tasks and the associated datasets, making it extremely useful for evaluating LLMs in the legal domain. Furthermore, as pointed out in *5.2 Performance Trends*, predictably, there are some choices about architecture, pretraining data, and others that influence the type of reasoning ability and fit better for a specific task.
The problem of determining if, given the facts of a case, a law applies to it is an important skill for legal professionals and a system which could accelerate this process would come in handy. A useful dataset for this kind of job is the Statutory Reasoning Assessment (SARA) dataset introduced in 2020. We found its application in [4, 6], even though in the latter one is proposed an updated version with a different representation for entities and events.
The LogiLaw dataset introduced in [11] is based on the COLIEE dataset (it contains legal questions and related articles) enriched with both generated Prolog code and corresponding verification results to capture the underlying logical reasoning required to answer the questions appropriately. The aim of that work is to improve the training and evaluation of LLMs in legal reasoning and related tasks.

RQ3.1 How effective are these benchmarks in capturing the complexities of real-world legal reasoning and information retrieval?

The benchmarks we have reviewed so far are high-quality and most of them focus on addressing a specific problem. For example, the SARA dataset is specialized in retrieving information instead of evaluating logical reasoning as LR-SAT. Both of them capture different shades of real-world scenarios, but it is evident that if we take into account only one dataset as representative for the entire legal domain, we will commit a mistake. The legal documents may differ by numerous factors, such as the context of application or the country in which they are valid; this makes it difficult to build a comprehensive dataset for law in general. Furthermore, we can observe that the benchmarks are usually less verbally complex than real-life legal documents. Usually, the latter ones are longer and have a more elaborate text structure typical of technical texts. The authors of [6] suggest that future datasets should incorporate more varied legal cases, longer texts, and richer semantic annotations to better reflect the challenges faced in real-world legal information

[1] `https://github.com/HazyResearch/legalbench/`

retrieval. The special structure of legal texts affects also the prompting techniques. In fact, another evidence of the gap between the language "spoke" by the LLMs and legal practitioners stands in the prompting strategies. In [4] section *5.4 Prompt engineering strategies*, it is highlighted that prompting for legal tasks could require a different strategy than general domain tasks due to the lower frequency of legal terms in general domain training corpora. In order to overcome this problem, practitioners must provide additional background information. This remarks that LLMs are used to a simpler language structure; through preliminary experiments, the authors found out that, on four out of five tasks, the plain-language prompt significantly outperforms the technical language prompt. Moreover, it is crucial to choose the most appropriate benchmark in order to properly evaluate an LLM in a specific task. In this way, we ensure a greater similarity between the sample and the real case of application. In addition, the definition of tasks in LegalBench leverages the IRAC (Issue, Rule, Application and Conclusion) framework used by practicing lawyers to face the legal reasoning. We believe that some of these benchmarks are really close in capturing the complexities of real-world legal reasoning, such as the task "ucc_v_common_law" which evaluates an LLM's ability to determine whether a particular contract is covered by the Uniform Commercial Code (UCC) or the common law, given information about the contract. Nevertheless, the benchmarks took into consideration do not include tasks in which the output could be right or wrong at some degrees. In [11] is proposed the Reinforcement Learning from Logical Feedback (RLLF) approach for promoting accurate logical reasoning and reducing human feedback biases. In conclusion, even though some benchmarks are close to every day applications, we can assert that the entire legal domain is not completely covered yet; thus new benchmarks and datasets should be developed to achieve further improvements in both legal reasoning and information retrieval.

RQ4: How do logical inconsistencies affect reasoning and information retrieval in the legal domain, and how are these conflicts identified and handled by the systems ?

When talking about inconsistencies we have to point out a distinction between the ones originated from the nature of law and logical inconsistencies which arise from incomplete symbolic representations and ambiguous contexts.
In [23] is defined what constitutes a conflict between legal rules and explains how these conflicts can arise through different types of "attack" relations in structured argumentation. There are three kinds of attack relation:

1. Undermining Attack: it targets the premises of an argument. Essentially, it challenges the foundational assumptions or facts that support the argument, as a consequence, it cannot be treated as an attack on a legal rule.

2. Undercutting Attack: it challenges the inference step of an argument, essentially questioning the validity of the rule or reasoning that connects the premises to the conclusion. Furthermore, this type of attack introduces exceptions to the rule being applied. For example, consider the rule: "If a vehicle enters a park, it is not allowed". An undercutting attack could argue: "If the vehicle is an ambulance on an emergency, this rule does not apply".

3. Rebutting Attack: it establishes a contradictory or contrary conclusion that directly opposes the conclusion of the attacked argument. By way of example suppose that an argument concludes "Vehicles are prohibited in the park". A rebutting attack could argue "Vehicles are allowed in the park if they are part of a public service".

RQ4.1: How are conflicts and ambiguities traditionally resolved by legal practitioners ?

Despite we collected hundreds of papers; we found only one of them which delves into conflict management between legal rules. In particular, the work we will use for analyzing this research question is [23] by Zurek published in 2016. Even though this study is not recent, it proposes a formal model of the mechanism of conflict recognition along with three different mechanisms for solving them. The author explains how legal practitioners traditionally resolve conflicts and ambiguities using well-established doctrines in statutory interpretation by formalizing three main legal principles. It is essential to underline the distinction between legal principles and legal rules: the first one may be applied with a certain range, while the second one is applicable or not. Consequently, when there is a conflict among multiple rules, only one can be used. As [23] focus only on the resolution of conflicts between legal rules, we will not treat contradictory legal principles.
In section *Methods of conflict resolving between legal rules*, the four primary methods of resolving conflicts between legal rules from the theory of law are shown. They are presented in order of priority and do not have equal power: if the stronger one does not solve the problem, then it is possible to use the weaker one.

1. *Lex superior derogat legi inferiori*: the concept is based on the hierarchical structure of law. In case a rule is implied in a conflict and stays in a higher position in the hierarchy, then the latter takes precedence over the rule from lower level (e.g., in Italian law, the constitution has a higher priority than regional regulation). As a consequence, legal practitioners prioritize rules depending on their source of authority.

2. *Lex posterior derogat legi priori*: it is crucial to assess when a legal act was established; newer legal acts prevail over older ones. For this reason, profes-

sional in the legal domain rely on legislative history to resolve such conflicts. Furthermore, it is important to remember that this method can be applied only if both conflict acts have the same rank and specificity.

3. *Lex specialis derogat legi generali*: in this method the idea of specification is applied, particularly, a specific act overrides a general regulation. By analyzing the scope of conflicting legal rules, this mechanism allows for conflicts to be resolved between rules within the same legal act.

4. Argument from social importance: it is the weakest way for resolving conflicts and is the most controversial one because it is based on the distinction between axiological contexts of conflicting norms. Namely, a norm may be more significant from the point of view of social importance, and this should be applied instead of the less significant norm. Due to the uncertainty of interpretation and evaluation of social importance, it is desirable to avoid its usage.

RQ4.2 Which techniques can be used to detect and resolve such logical inconsistencies ?

In [23] are presented three mechanisms for resolving conflict which follow the approach described in RQ4.1 from legal practitioners. Before delving into these models, we have to point out some assumption the author made: for the purpose of creating simple model, they used propositional logic but more expressive logics and both interpretation or inference mechanisms could also be suitable. In addition, they assumed a language $\mathcal{L}$ which contains a set of operators $OP = \{\neg, \sim, \supset, \vee, \wedge, \twoheadrightarrow\}$ and potentially more - the symbols $\sim, \twoheadrightarrow, \supset$ represent respectively negation as failure, a binary connective which stands for a defeasible legal rule, a classical (material) implication used in commonsense rules. Considering the set of propositional atoms called facts $F = \{f_1, f_2, \dots\}$, they defined the formula of a regale rule as:
$r_n : Conditions \twoheadrightarrow Conclusion$ where:

- n is the name of the rule;
- $Conditions$ is a (possibly empty) antecedent formula in the form of $\{c1\ func\ c2\ func\ \dots\}$, where c_n are atomic conditions, each one can be a positive fact or a negated one by $\neg$ or $\sim$ or both of them, and $func \in \{\wedge, \vee\}$;
- $Conclusion$ is a non-empty rule in the form: $Conclusion = (lx \wedge ly \wedge \dots)$, where: lx, ly are atomic conclusions which can be positive or negative facts.

To wrap up the notions necessary to explain the proposed models in [23] we have to introduce $K \bullet Conditions$, which indicates that the knowledge base K satisfies the

condition of a given legal rule. As a result, we have to introduce the following rule $r_n : Conditions \twoheadrightarrow Conclusion \wedge K \bullet Conditions \Rightarrow conclusion$.
Considering that:

- $ACT = \{act_1, act_2, \dots\} \in K_l$ is the set of legal rules. To denote that a legal rule is taken from act we use $r_l \in act_x$
- $Prem(\cdot)$ is a function which returns the premises of an argument;
- A, B are the arguments built on conflicting legal rules;
- If it is possible to deduct an order between the arguments A and B on the basis of a principle, then the first one wins.

Let us follow the formalization of the models (each one refers to a subsection in section *IV. Model of conflict resolution* in [23]):

- *Lex superior derogat legi inferiori*: if r_n and r_m are recognized as conflicting rules, in order to resolve the conflict it is possible to apply:
  $lexSuperior : (r_n \in act_k) \wedge (r_m \in act_l) \wedge (H(act_k) = hch_x) \wedge (H(act_l) = hch_y) \wedge (hch_x >_{hch} hch_y) \wedge (r_n \in Prem(A)) \wedge (r_m \in Prem(B)) \Rightarrow A \succeq B$
  where, the primary symbols:
  - $HCH = \{hch_1, hch_2, \dots\} \in K_n$ is the set of levels of a hierarchy;
  - $H : ACT \rightarrow HCH$ is the function which assign to a given legal act a hierarchy level;
- *Lex posterior derogat legi priori*: if r_n and r_m are legal rules in conflict that cannot be resolved by lexSuperior or lexSpecialist, it is possible to apply:
  $lexPosterior : (r_n \in act_k) \wedge (r_m \in act_l) \wedge (D(act_k) = date_k) \wedge (D(act_l) = date_l) \wedge (date_k >_{time} date_l) \wedge (r_n \in Prem(A)) \wedge (r_m \in Prem(B)) \Rightarrow A \succeq B$
  where:
  - $DATE = \{hch_1, hch_2, \dots\} \in K_n$ is the set of all dates of issue of all legal acts;
  - $D : ACT \rightarrow DATE$ is the function which assign to a given legal act a the date of issue;
  - $date_m >_{time} date_n$ indicates that $date_m$ was earlier than $date_n$.
- *Lex specialis derogat legi generali*: the biggest difference between the first two and *Lex specialis*, is that the latter is based on common sense knowledge instead of legal knowledge taken from statutes. In particular:

Even though the proposed model from Zurek does not allow the recognition of all general-specific relations between rules (it is impossible to predict all possible real-life cases). Instead, it analyzes the antecedents of conflicting rules to discover whether the condition of subsumption is fulfilled.

In general, when it is possible to define the specificity relation between legal rules, represented by $>_{spec}$, it can be applied the following:
$lexSpecialis : (r_n >_{spec} r_m) \wedge (r_n \in Prem(A)) \wedge (r_m \in Prem(B)) \Rightarrow A \succeq B.$

In determined situations, the $>_{spec}$ can be obtained through inferences, for example:

- $subsumingRule : (r_1 : Conditions1 \twoheadrightarrow Conclusion1) \wedge (r_2 : (Conditions1) \vee (Conditions1a) \twoheadrightarrow Conclusion2) \wedge (Conditions1 \neq Conditions1a) \Rightarrow r_1 >_{spec} r_2.$
- $restrinctingRule : (r_1 : Conditions1 \twoheadrightarrow Conclusion1) \wedge (r_2 : (Conditions1) \wedge (Conditions1a) \twoheadrightarrow Conclusion2) \wedge (Conditions1 \neq Conditions1a) \Rightarrow r_2 >_{spec} r_1;$

4.2 Summary

In RQ1 are highlighted the intrinsic challenges that LLMs face in performing legal reasoning tasks, which hinder the inference of logical steps. The problem of explainability is also a main concern. In RQ2 and its sub-question we discuss the limitations in performing IR by LLMs and some promising solutions. For example, a richer ontology and better understanding of the context could improve the performance of the system. The answer of RQ3 showcases several benchmarks and datasets used to evaluate logical reasoning tasks. Finally, the RQ4 deals with the conflict resolution from the point of legal practitioners and also showcases the techniques for solving logical inconsistencies.

Chapter 5

Limitations, Threats to Validity and Review Assessment

5.1 Limitations

In this section we will analyze the potential obstacles that could have affected our review in its objectiveness. In the selection of sources we have chosen to use a limited amount of databases, specifically: IEEE Xplore, Scopus and Web of Science. Nevertheless, we integrated some papers from other sources and we provided good reasons in 3.4.1. Although our database selection could have been refined, we are confident that it was broad enough to guarantee the academic integrity of our findings. In addition, one may object that the inclusion criteria (the work is written in English) could limit our research capabilities. However, considering that the majority of publications in our field are written in English, the requirement we adopted in this SLR is neither unusual nor uncommon.

5.2 Threats to Validity

Next, we address potential biases that may pose some doubt on the validity of our work [22]. In order to avoid biases we evaluate each paper through the use of quality criteria to ensure high quality of the materials. In addition, for bias like the inclusion criteria we endeavored to establish the most general and appropriate criteria for our topic.

5.3 Review Assessment

The final phase to evaluate the quality of the SLR is to reflect on the work presented through a series of questions that will act as a benchmark.

Are the review's inclusion and exclusion criteria described and appropriate? The inclusion and exclusion criteria were clearly outlined in our protocol and are aligned with the best norms in our field. Thus, we firmly believe that the criteria used are appropriate for our research topic.

Did the reviewers assess the quality/validity of the included studies? To ensure high standard material, we set several quality criteria 3.3.2 and evaluated each study, achieving an average of 3,67 out of 4, which is assessed in the electronic sheets we shared at the end of 3.4.

Does the search process cover all possible relevant papers? Considering that the number of new papers published every day is high, we have to underline that the risk of missing new studies cannot be overlooked. However, we can affirm that the papers gathered are representative of the field.

Chapter 6

Prototype

The purpose of this prototype is to create a small-scale system, a proof-of-concept, for conflict resolution based on research questions RQ4.1 and RQ4.2. The idea is to have a system that has legal knowledge, it can be asked about some topic and it will retrieve relevant documents. If several laws are involved and come to different conclusions, the system will be able to choose which law has to be enforced first.

6.1 Structure of Cheshire Cat AI

The foundation of this prototype is based on an open source Italian project, the Cheshire Cat AI framework [1], which allows the build of AI agents on top of LLMs. In [17] the structure of the framework is described in greater detail. The framework is composed of the following elements:

- LLM and Embedder: both elements have a central role for the chatbot, the first one actually generates the responses, while the latter one converts input text into vector representations. By employing a Factory design pattern, the framework remains agnostic about which specific LLM or Embedder is used, allowing dynamic selection based on user needs and available resources;
- Vector Database: it is essential for storing and retrieving information. In Cheshire Cat AI, it is used an open source vector database, Qdrant[2], that adopts the HNSW (Hierarchical Navigable Small World) algorithm to search for similar vectors;
- Rabbit hole: it accomplishes the task of ingesting documents and storing them in the declarative memory. It is possible to interact with it either through its endpoint, the GUI or a Python script;

[1] `https://github.com/cheshire-cat-ai`
[2] `https://qdrant.tech/qdrant-vector-database/`

- Long Term Memory: LTM is the framework's persistent memory and is divided into three parts:
  - Episodic memory, stores all previous user interactions (questions and responses);
  - Declarative memory, holds all information ingested via the Rabbit Hole, which can be recalled to influence the prompt;
  - Procedural Memory, contains the available tools and instructions (i.e. how to activate them).

  From each of the these memories information are extracted and saved in the Working Memory, which, in addition to these data, will also contain the history of the conversation;
- Mad Hatter: it is the plugin manager, it is responsible for loading, prioritizing and executing plugins;
- White Rabbit: is a component that deals with scheduling computations to be performed at exact time. These include events to be triggered only once or periodically (using *cronjobs*);
- Agent Manager: as stated in the documentation of the Cheshire Cat AI[3], the Agent Manager manages the execution of language models chains, namely, it is a pipeline that takes one or more input variables, it formats them in a prompt, submits the prompt to a language model and, optionally, parses the output.

6.2 Prototype's structure

The prototype is essentially a plugin built on top of the Cheshire Cat AI framework. Addresses legal conflicts by selecting, through a series of principles, the most appropriate law from a set of retrieved legal documents. The core of the prototype is the HandleConflict class, which orchestrates the decision-making process when the user produces a query. [4] Its workflow follows these steps:

1. Lex Superior: it checks the hierarchy and filters out laws with lower priority. If a single law remains, it is immediately chosen;
2. Lex Posterior: if more than one law shares the highest hierarchy, the newest law will be chosen by comparing the dates.

[3] `https://cheshire-cat-ai.github.io/docs/framework/cat-components/cheshire_cat/agent`

[4] The plugin's code is published at `https://github.com/sc-ale/ThesisPrototype`

3. Lex Specialis: if both previous rules fail to single out one law, the plugin constructs a prompt for the LLM that includes all candidate laws along with legal reasoning guidelines. The prompt instructs the model to choose the most specific law based on principles such as restricting and subsuming rules. The LLM response, expected in JSON format with 'index' and 'motivation' fields, is parsed and used to determine the final result.

In order to facilitate the work with attributes like hierarchy and date, they are included as metadata during the parsing process of the document. Furthermore, several hooks are being employed for integrating the HandleConflict class with the Cat, and customizing some aspects, like the threshold of the declarative memory when retrieving documents. The Figure 6.1 represents the prototype's flowchart when the user sends a message. In particular, after saving the user's question in the working memory, the embedder converts the input string into a vector that will be saved in the episodic memory and then retrieves all related data. The hook "after cat recalls memory" will be fired, calling the HandleConflict object if at least one document is retrieved. Subsequent to the application of one or more principles, it will remain only one law, this one will be set up in the Cat's memory and used for creating the output returned with the "agent fast reply" hook.
This implementation not only showcases the flexibility of the framework but also illustrates a practical application of integrating domain-specific knowledge into a chatbot system.

6.2.1 Implementation choices

- We have chosen to use made up document with a simple form "ID - Hierarchy - Date - Content" instead of real-world documents because a more sophisticated parser would have been necessary otherwise;

- Regarding the documents, we defined a custom priority hierarchy $Constitution > Regional > Municipal$ due to the fact that we are only showcasing a prototype, thus in an actual application there would be some consideration to take into account such as the country and the related legal system;

- In the hook "rabbithole_instantiates_splitter" the default text splitter is overridden with a custom function which returns the entire text as a single document. Since the texts are short, applying the chunking technique would not be useful.

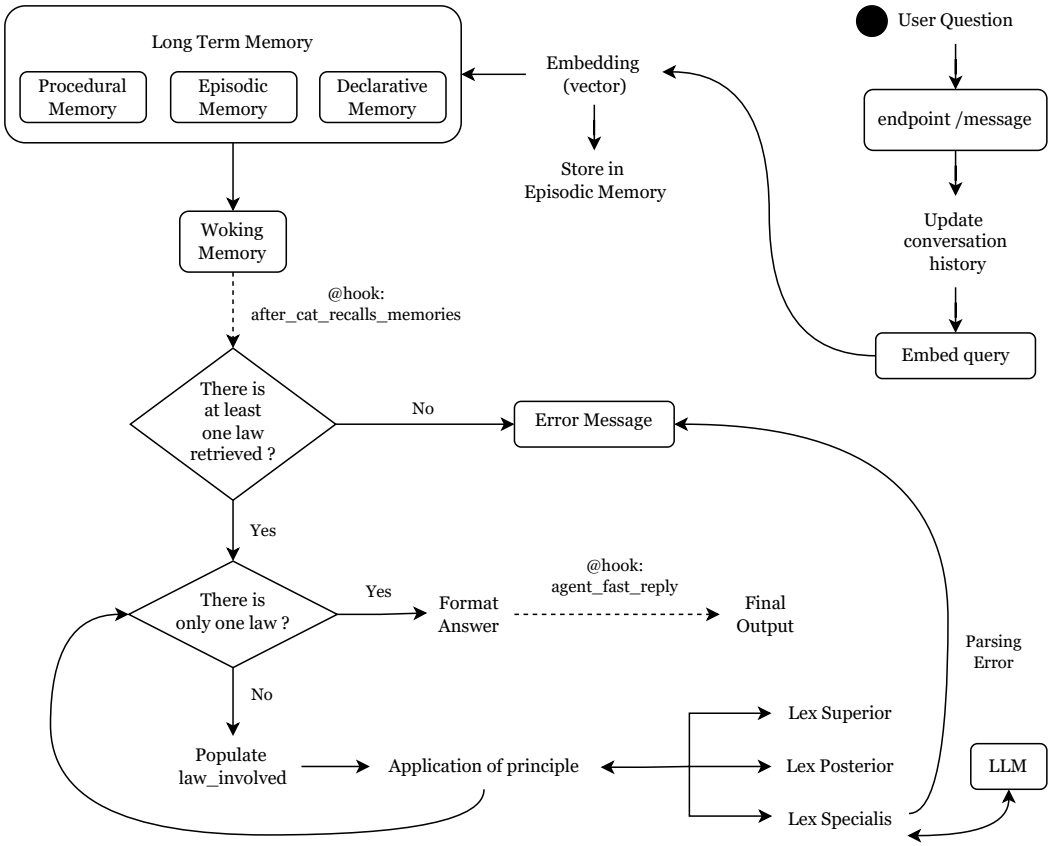

Figure 6.1: Prototype's flowchart

6.3 Examples

In this section, we will glide through some examples of the prototype, where each principle is applied.
Consider the following laws:

- L3 - Constitution - 2006-08-10 - Smoking is prohibited inside all indoor public places.
- L4 - Municipal - 1964-02-24 - Smoking is allowed both in outdoor and indoor public places.

The following screenshot illustrates the application of the lex Superior principle, in this case L3 prevails over L4 due to the higher priority.

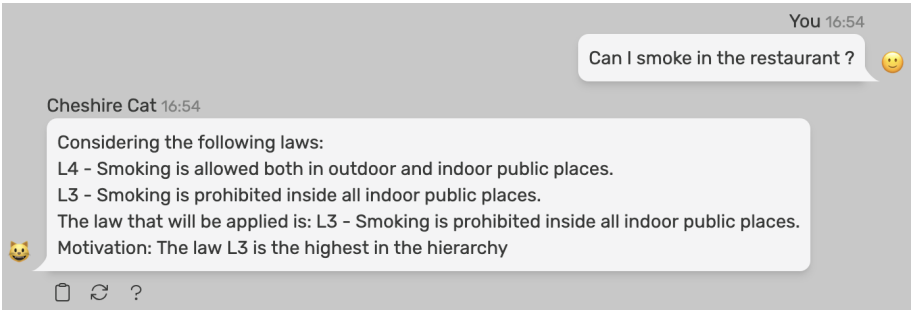

Figure 6.2: Case of Lex Superior application

Consider the following laws:

- L5 - Regional - 2015-07-01 - Construction work is banned in residential areas on weekends.
- L6 - Regional - 2019-05-20 - Public infrastructure projects may proceed on weekends.
- L7 - Regional - 2022-02-13 - Designated urban zones permit weekend construction.

The following screenshot illustrates the application of the lex Posterior principle: in light of the same level in the hierarchy, it is applied the newest law L7.
Consider the following laws:

- L1 - Municipal - 2008-02-15 - No vehicles are allowed in the public park.
- L2 - Municipal - 2008-02-15 - Emergency vehicles are permitted in the public park.

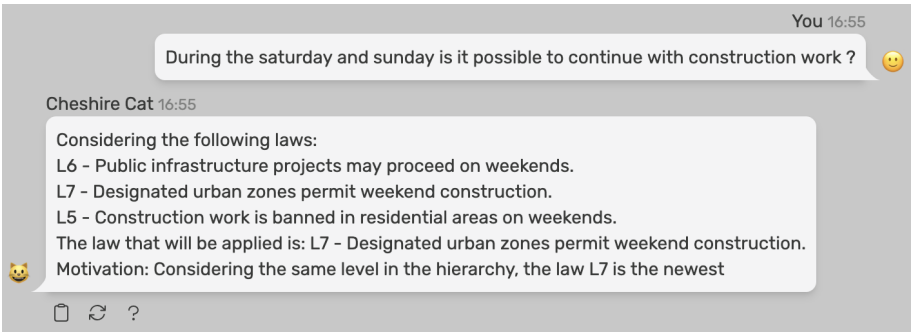

Figure 6.3: Case of Lex Posterior application

The following screenshot illustrates the application of the lex Specialis principle: due to the same priority in the hierarchy and date of release, L2 is applied because more specific in that context.

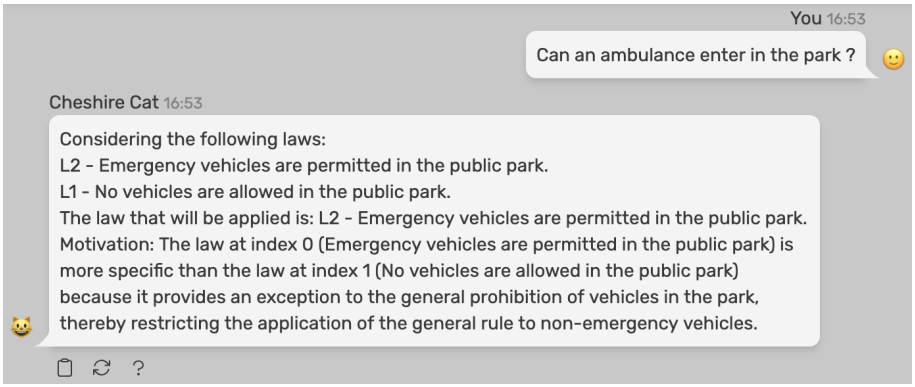

Figure 6.4: Case of Lex Specialis application

In conclusion of this section, we would like to remark that the prototype handles edge cases such as the retrieval of only one law or none of them.

6.4 Limit

We are aware of the limitations of this work but we strongly believe that with enough resources such as a sophisticated parser, an external solver of FOL formulae and other elements, it could assist the legal practitioners relieving some of their workload.

Chapter 7

Conclusion

This work explores the integration of AI in the legal domain, with a particular focus on IR and LR tasks. The study provides a comprehensive overview of the limitations of current AI-driven systems and highlights promising future directions. Through the analysis of the collected papers, we observed that LLMs show strong performance in semantic matching but struggle with symbolic logic reasoning, which is essential for legal interpretation and decision-making. Integrating neural-symbolic approaches seems a more robust solution. In addition, the introduction of symbolic reasoning and structured representations further enhances the accuracy and interpretability of legal AI systems. At the end, we discussed the management of conflicting laws and their resolution, showcasing a prototype based on the Cheshire Cat AI framework, which leverages the potential of RAG techniques, partially implementing the method described in RQ4.

7.1 Future directions

The task of LR represents an hot topic for its potential impact in a wide range of fields. Even though this work is strictly related to the legal domain, it is important to show the latest cutting-edge technologies (at the best of our knowledge) for LR because they could be applied in future to this field. An interesting approach for this problem is to not rely only on LLM in light of their probabilistic nature, but to add symbolic knowledge and logic. Virtuous examples are [12, 13], where LLMs are used for translating the natural language to a structured representation that will be the input for a deterministic solver. In general, the neural-symbolic models are the one with the best trade-off in terms of accuracy and interpretability, which is mandatory for taking in consideration the output, ensuring that the reasoning steps are logically valid. Furthermore, the possibility to see the process allow to understand where it eventually failed and to use self-refinement strategies which use

both the previous input-output and the error produced, e.g. by the external solver, in order to give a more exhaustive prompt. We cannot mention the value provided by Retrieval Augmented Generation (RAG) techniques which aims to transfer a specific knowledge base in LLMs; in particular the "Blendend RAG" method proposed in [15], combines dense vector indexes and sparse encoder indexes with hybrid query strategies, enhancing the semantic search over keyword similarity-based searches.

7.2 Ethical issues

Technology innovation cannot be stopped, and AI is one in the lead. Its usage is at point of no return; it would be unimaginable to stop using it considering how much it has grounded in everyday life. Even though its benefits, this has a cost. In this SLR we concerned about how AI could assist legal practitioners, supporting and relieving some of their workload but we are aware that the adoption of this technology could cause job losses for some legal professionals, especially junior figures [7]. Regardless of major improvements in the field, the abilities such as handling nuanced legal issues, typical of domain expertise, cannot be replaced by AI systems. On another plane, the problem of global warming also involves the use of AI. As evidenced in [21], the energy consumption needed for training and regular utilization by users is remarkable, which contributes to the energy footprint (adopting pre-trained model is a way to limit this problem).

Bibliography

[1] Pearl Brereton, Barbara A. Kitchenham, David Budgen, Mark Turner, and Mohamed Khalil. "Lessons from applying the systematic literature review process within the software engineering domain". In: *Journal of Systems and Software* 80.4 (2007). Software Performance, pp. 571–583. ISSN: 0164-1212. DOI: https://doi.org/10.1016/j.jss.2006.07.009. URL: https://www.sciencedirect.com/science/article/pii/S016412120600197X.

[2] Tore Dybå and Torgeir Dingsøyr. "Strength of Evidence in Systematic Reviews in Software Engineering". In: Oct. 2008, pp. 178–187. DOI: 10.1145/1414004.1414034.

[3] Leilei Gan, Kun Kuang, Yi Yang, and Fei Wu. "Judgment Prediction via Injecting Legal Knowledge into Neural Networks". In: *Proceedings of the AAAI Conference on Artificial Intelligence* 35.14 (May 2021), pp. 12866–12874. DOI: 10.1609/aaai.v35i14.17522. URL: https://ojs.aaai.org/index.php/AAAI/article/view/17522.

[4] Neel Guha, Julian Nyarko, Daniel E. Ho, Christopher Ré, Adam Chilton, Aditya Narayana, Alex Chohlas-Wood, Austin Peters, Brandon Waldon, Daniel N. Rockmore, Diego Zambrano, Dmitry Talisman, Enam Hoque, Faiz Surani, Frank Fagan, Galit Sarfaty, Gregory M. Dickinson, Haggai Porat, Jason Hegland, Jessica Wu, Joe Nudell, Joel Niklaus, John Nay, Jonathan H. Choi, Kevin Tobia, Margaret Hagan, Megan Ma, Michael Livermore, Nikon Rasumov-Rahe, Nils Holzenberger, Noam Kolt, Peter Henderson, Sean Rehaag, Sharad Goel, Shang Gao, Spencer Williams, Sunny Gandhi, Tom Zur, Varun Iyer, and Zehua Li. *LegalBench: A Collaboratively Built Benchmark for Measuring Legal Reasoning in Large Language Models*. 2023. arXiv: 2308.11462 [cs.CL]. URL: https://arxiv.org/abs/2308.11462.

[5] Kailash A. Hambarde and Hugo Proença. "Information Retrieval: Recent Advances and Beyond". In: *IEEE Access* 11 (2023), pp. 76581–76604. DOI: 10.1109/ACCESS.2023.3295776.

[6] Nils Holzenberger and Benjamin Van Durme. "Connecting Symbolic Statutory Reasoning with Legal Information Extraction". In: All Open Access, Hybrid Gold Open Access. 2023, pp. 113–131. DOI: 10.18653/v1/2023.nllp-1.12. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85184996768&doi=10.18653%2fv1%2f2023.nllp-1.12&partnerID=40&md5=b4367e51dd40ccd00552c73686c928f9.

[7] British Institute of International and Comparative Law. *Use of Artificial Intelligence in Legal Practice*. 2023. URL: https://www.biicl.org/documents/170_use_of_artificial_intelligence_in_legal_practice_final.pdf.

[8] Yaohui JIN and Hao HE. "An Artificial-Intelligence-Based Semantic Assist Framework for Judicial Trials". In: *Asian Journal of Law and Society* 7.3 (2020), pp. 531–540. DOI: 10.1017/als.2020.33.

[9] Barbara Kitchenham. "Procedures for Performing Systematic Reviews". In: *Keele, UK, Keele Univ.* 33 (Aug. 2004).

[10] Barbara Ann Kitchenham and Stuart Charters. *Guidelines for performing Systematic Literature Reviews in Software Engineering*. English. Tech. rep. EBSE 2007-001. Keele University and Durham University Joint Report, July 2007. URL: https://www.elsevier.com/__data/promis_misc/525444systematicreviewsguide.pdf.

[11] Ha-Thanh Nguyen, Wachara Fungwacharakorn, and Ken Satoh. "LogiLaw Dataset Towards Reinforcement Learning from Logical Feedback (RLLF)". In: *Frontiers in Artificial Intelligence and Applications* 379 (2023). Cited by: 0; All Open Access, Hybrid Gold Open Access, pp. 217–226. DOI: 10.3233/FAIA230967. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85181175677&doi=10.3233%2fFAIA230967&partnerID=40&md5=564319506835d9bbf85138982b6acfcd.

[12] Theo Olausson, Alex Gu, Ben Lipkin, Cedegao Zhang, Armando Solar-Lezama, Joshua Tenenbaum, and Roger Levy. "LINC: A Neurosymbolic Approach for Logical Reasoning by Combining Language Models with First-Order Logic Provers". In: *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. Ed. by Houda Bouamor, Juan Pino, and Kalika Bali. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 5153–5176. DOI: 10.18653/v1/2023.emnlp-main.313. URL: https://aclanthology.org/2023.emnlp-main.313.

[13] Liangming Pan, Alon Albalak, Xinyi Wang, and William Wang. "Logic-LM: Empowering Large Language Models with Symbolic Solvers for Faithful Logical Reasoning". In: *Findings of the Association for Computational Linguistics: EMNLP 2023*. Ed. by Houda Bouamor, Juan Pino, and Kalika Bali. Accessed:

2024/09/10. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 3806–3824. DOI: 10.18653/v1/2023.findings-emnlp.248. URL: https://aclanthology.org/2023.findings-emnlp.248.

[14] Carlo Sansone and Giancarlo Sperlí. "Legal Information Retrieval systems: State-of-the-art and open issues". In: *Information Systems* 106 (2022), p. 101967. ISSN: 0306-4379. DOI: https://doi.org/10.1016/j.is.2021.101967. URL: https://www.sciencedirect.com/science/article/pii/S0306437921001551.

[15] Kunal Sawarkar, Abhilasha Mangal, and Shivam Raj Solanki. "Blended RAG: Improving RAG (Retriever-Augmented Generation) Accuracy with Semantic Search and Hybrid Query-Based Retrievers". In: *2024 IEEE 7th International Conference on Multimedia Information Processing and Retrieval (MIPR)*. 2024, pp. 155–161. DOI: 10.1109/MIPR62202.2024.00031.

[16] ZhongXiang Sun, Kepu Zhang, Weijie Yu, Haoyu Wang, and Jun Xu. "Logic Rules as Explanations for Legal Case Retrieval". In: *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*. Ed. by Nicoletta Calzolari, Min-Yen Kan, Veronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue. Torino, Italia: ELRA and ICCL, May 2024, pp. 10747–10759. URL: https://aclanthology.org/2024.lrec-main.939.

[17] Francesco Testa. *Cheshire Cat AI: framework per la creazione di chatbot specializzati*. 2024. URL: https://amslaurea.unibo.it/id/eprint/33476/.

[18] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. *Attention Is All You Need*. 2023. arXiv: 1706.03762 [cs.CL]. URL: https://arxiv.org/abs/1706.03762.

[19] Siyuan Wang, Zhongkun Liu, Wanjun Zhong, Ming Zhou, Zhongyu Wei, Zhumin Chen, and Nan Duan. "From LSAT: The Progress and Challenges of Complex Reasoning". In: *IEEE/ACM Transactions on Audio Speech and Language Processing* 30 (2022). Cited by: 11; All Open Access, Green Open Access, pp. 2201–2216. DOI: 10.1109/TASLP.2022.3164218. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85127497922&doi=10.1109%2fTASLP.2022.3164218&partnerID=40&md5=8b61cb08011d6f3dbae371100d65c96b.

[20] Siyuan Wang, Wanjun Zhong, Duyu Tang, Zhongyu Wei, Zhihao Fan, Daxin Jiang, Ming Zhou, and Nan Duan. "Logic-Driven Context Extension and Data Augmentation for Logical Reasoning of Text". In: 2022, pp. 1619–1629. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85132164788&partnerID=40&md5=95b4d9e6cef0b9a07b310a88fbd2dd37.

[21] Yang Yu, Jiahui Wang, Yu Liu, Pingfeng Yu, Dongsheng Wang, Ping Zheng, and Meng Zhang. "Revisit the environmental impact of artificial intelligence: the overlooked carbon emission source?" In: *Frontiers of Environmental Science & Engineering* 18.12 (Oct. 2024), p. 158. ISSN: 2095-221X. DOI: 10.1007/s11783-024-1918-y. URL: https://doi.org/10.1007/s11783-024-1918-y.

[22] Xin Zhou, Yuqin Jin, He Zhang, Shanshan Li, and Xin Huang. "A Map of Threats to Validity of Systematic Literature Reviews in Software Engineering". In: *2016 23rd Asia-Pacific Software Engineering Conference (APSEC)*. 2016, pp. 153–160. DOI: 10.1109/APSEC.2016.031.

[23] Tomasz Zurek. "Modeling conflicts between legal rules". In: *2016 Federated Conference on Computer Science and Information Systems (FedCSIS)* (2016), pp. 393–402. URL: https://api.semanticscholar.org/CorpusID:119344.

Appendix A

Summary of the consulted papers

A.1 An Artificial-Intelligence-Based Semantic Assist Framework for Judicial Trials

JIN and HE [8] proposes a novel system that leverages AI to enhance the judicial decision-making process. The framework incorporates natural language processing (NLP) and semantic analysis to process legal documents and extract pertinent information. Because the electronic files and judicial documents have different structures and data distribution, it is introduced the legal fact as the basic unit of information in legal texts.
The semantic analysis component identifies relationships between legal facts, enabling the system to understand context and relevance. This framework aims to reduce workload on legal practitioners, thereby improving the quality and speed of legal research. In particular, it can accurately extract and identify the facts needed and the operation mode of the framework conforms to the logic process of judicial judgment, ensuring the traceability of intermediate results.

A.2 Connecting Symbolic Statutory Reasoning with Legal Information Extraction

Holzenberger and Van Durme [6] tackle the issue of statutory reasoning, specifically determining whether a given law – a part of a statute – applies to a given legal case. The aim of the research is to investigate a form of legal information extraction upon the StAtutory Reasoning Assessment (SARA) dataset, which is a benchmark for US federal tax law. The study integrates symbolic reasoning with Information Extraction (IE), automating the process of translating case descriptions into Prolog Knowledge Bases (KBs) for legal reasoning.

The authors focus on improving the SARA dataset's annotations for enhanced consistency and scalability, introducing a structured ontology to map entities and events, and implementing a span-based parser for automatic KB extraction. The parser utilizes pre-trained language models like LEGALBERT and RoBERTa for encoding textual data into structured representations.
Experiments show that IE quality directly correlates with statutory reasoning performance. The enhanced Prolog KBs improve reasoning tasks, achieving better interpretability and auditability in the model's output, since it becomes clear what facts are used to perform statutory reasoning. The authors highlight limitations in the ontology's ability to represent complex temporal relationships and suggest adopting more flexible frameworks like event calculus for future improvements.
Finally, the paper shows the limitations of the study, admitting that the conclusions drawn about legal-domain IE and about statutory reasoning should take care to understand that the SARA dataset is not representative of the full scope of legal data since real-world legal cases are generally much longer, the language is denser, and the phrasing is more diverse.

A.3 Logic-Driven Context Extension and Data Augmentation for Logical Reasoning of Text

Wang et al. [20] proposed a logic-driven context extension framework and a logic-driven data augmentation algorithm. In particular, the first one called LReasoner, follows three step reasoning paradigm:

1. Logic identification: extracts logical expressions as elementary reasoning units. In order to ensure the general applicability of the framework, they designed a fairly simple logical identification approach using an off-the-shelf constituency parser and several common keywords of logical semantics. The logical symbols in each sentence are combined by logical connectives to constitute logical expressions, handling negative and conditional relationship between symbols.

2. Logic extension: performs logical inference over the identified logical expressions employing equivalence laws, specifically contraposition and transitivity, obtaining implicit logical expressions.

3. Logic verbalization: converts the extended logical expression set into natural language to serve as extended context. The final representation is feed into a classification layer to get each option's score and chose the one with the highest value as the predicted answer.

The framework also introduces a logic-driven data augmentation algorithm to improve logical understanding. By creating contrastive samples with logically altered

expressions, the system trains to distinguish between logically valid and invalid contexts, enhancing its ability to capture negation and conditional relationships. The score function calculates the score that the correct answer can achieve in a given context: $s'(c^+, q, o_a) \gg s'(c^-, q, o_a)$ where s' is the score function, and (c^+, q, o_a) is the positive sample with the positive context c^+, negative the other. Therefore, the contrastive loss can be formulated as a classification loss for predicting the most plausible context that supports the answer.
Evaluated on two challenging logical reasoning datasets, ReClor and LogiQA, LReasoner achieves state-of-the-art results, surpassing human performance on ReClor. Ablation studies confirm that both the logic-driven context extension framework and data augmentation significantly improve performance. In conclusion, the paper highlights the generalisability of the method, demonstrating its effectiveness in other tasks like SQuAD.

A.4 From LSAT: The Progress and Challenges of Complex Reasoning

Wang et al. [19] center on studying three challenging and domain-general task of the Law School Admission Test (LSAT), including analytical reasoning (AR), logical reasoning (LR) and reading comprehension (RC). For AR it introduces three models: Analytical Reasoning Machine (ARM), Constraint based Analytical Reasoning (CGAR) and Neural-Symbolic Model (NSAR). For the LR task it presents the LReasoner model presented in [20], which integrates logic-driven context extension with pre-trained LLMs, aiming to improve the extraction and utilization of symbolic knowledge. For the RC task it proposes a neural model which has a dual multi-head CoAttention module based on pre-trained language models with transfer learning from RACE.
Empirical evaluation focuses on three LSAT datasets: LR-LSAT (Logical Reasoning), AR-LSAT (Analytical Reasoning), and RC-LSAT (Reading Comprehension), demonstrating the strengths and weaknesses of LLMs in handling real-world reasoning complexities. The study finds that while pre-trained LLMs excel in semantic matching, their lack of symbolic reasoning capabilities limits their performance on tasks which require formal logic.
The paper identifies significant challenges, including the inability of current models to generate and manipulate symbolic logical forms effectively. Proposed improvements consist of integrating symbolic reasoning frameworks with neural models to enhance interpretability and robustness in reasoning tasks. The authors emphasize the role of neural-symbolic integration as a promising approach for bridging the gap between logical inference and semantic understanding.

The conclusion highlights the outstanding performance of the model and identifies future directions, such as automatically extracting the logical elementary units and identifying the logical relationships between units in an unsupervised manner. These contributions provide a comprehensive overview of progress and remaining gaps in the domain of complex reasoning for LLMs.

A.5 Judgment Prediction via Injecting Legal Knowledge into Neural Networks

Gan et al. [3] addresses the task of Legal Judgment Prediction (LJP) by proposing a novel approach that incorporates explicit legal knowledge into neural networks. Traditional models treat LJP as a text classification task, relying heavily on data-driven methods, which are not interpretable and struggle to incorporate domain-specific legal reasoning. To overcome these limitations, the authors represent declarative legal knowledge using first-order logic (FOL) rules and integrate these rules into a co-attention neural network architecture which facilitates interaction between fact descriptions and claims, improving contextual representations. These outputs are then adjusted using a symbolic legal knowledge module based on probabilistic logic, which ensures predictions comply with legal principles. The approach was evaluated on a large dataset of private loan cases, including over 60,000 instances. Results demonstrate that injecting prior knowledge provides neural networks with inductive bias, which not only improves performance but also reduces data thirsty. Ablation studies showed that incorporating multiple FOL rules improves performance, enhancing accuracy. Comparisons with baseline models such as BERT, RoBERTa, and AutoJudge reveal the advantages of combining symbolic reasoning with neural networks.

A.6 LogiLaw Dataset Towards Reinforcement Learning from Logical Feedback (RLLF

Nguyen, Fungwacharakorn, and Satoh [11] proposes a refined evaluation method, introduce the LogiLaw dataset and a novel approach termed Reinforcement Learning from Logical Feedback (RLLF) to address limitations in the logical reasoning capabilities of LLMs within the legal domain. In particular, it proposes a refined evaluation method that requires LLMs to generate Prolog code to answer legal questions, followed by a Prolog independent engine which verify the correctness of the generated code. In this way, it ensures the models rely on accurate reasoning pathways and eliminates the possibility of models obtaining correct answers by chance or exploiting

superficial patterns, denoting an advantage over binary classification evaluation.
The LogiLaw dataset is derived from the COLIEE dataset and includes legal questions, related articles, generated Prolog code, and verification results from an independent Prolog engine. The aim of the dataset is to be used for training LLMs in order to improve their logical reasoning performance through reinforcement learning. Precisely, unlike traditional methods like Reinforcement Learning from Human Feedback (RLHF) which is witness of biases and subjectiveness, RLLF emphasizes logical feedback, leveraging Prolog engine outputs to enhance accuracy.
To demonstrate the value of the LogiLaw dataset, they conducted experiments using GPT-4 on the COLIEE dataset using Prolog code. The results show significant room for improvement in logical reasoning, with high error rates in Prolog code verification. The authors highlight challenges such as insufficient logical consistency and limited background knowledge in state-of-the-art models, underscoring the importance of datasets like LogiLaw and the opportunity RLLF provides to develop more effective training methods that prioritize logical insights over human feedback, reducing subjectivity and promoting accuracy. On a final note, the authors suggest as promising direction the integration of knowledge graphs and other external sources of structured information into the legal reasoning process.

A.7 Modeling conflicts between legal rules

Zurek [23] focuses on formalizing mechanisms for resolving conflicts between statutory legal rules to integrate them into legal advisory systems. The study is build on the basis of the $ASPIC^+$ argument modeling framework, which is very powerful and useful tool for structured argumentation representation. The author distinguishes between legal and commonsense rules, emphasizing the need for precise modeling to reflect statutory norms accurately, including their imperfections.
The paper identifies four primary methods for resolving legal conflicts: *Lex superior derogat legi inferiori* (higher legal priority prevails), *Lex posterior derogat legi priori* (later laws override earlier ones), *Lex specialis derogat legi generali* (specific laws take precedence over general ones), and arguments based on social importance. Among these, the first three are formally modeled and integrated into the $ASPIC^+$ framework, while the last one is not treated.
The study highlights that conflicts arise not only from direct contradictions but also through indirect attacks in argumentation, such as undercutting and rebutting attacks, defined in section C. *Model of conflict of legal rules.*
A key contribution is the formal model of three main methods of conflict solving. The prioritization of rules in argumentation systems ensures consistent application. The research concludes by discussing the challenges of applying these models to real-world cases, particularly modeling the strength of an argument and the balance

between two conflicting commonsense argument.

A.8 Logic Rules as Explanations for Legal Case Retrieval

Sun et al. [16] introduces Neural-Symbolic Legal Case Retrieval (NS-LCR), a model-agnostic framework (meaning that does not impose constraints on the choice of the neural retriever, thus it can be integrated with different retrieval models) for explainable legal case retrieval. NS-LCR addresses the need for interpretable reasoning in legal case retrieval by explicitly incorporating law-level and case-level logic rules into the retrieval process. These rules provide faithful and logically consistent explanations for the relevance of retrieved cases.
The framework combines two neuro-symbolic modules:

- Law-Level Logic Rules Module: Extracts predicates from law articles, formalizing them into first-order logic (FOL) rules connected by logical operators. The law-level explanation e_L defined by the equation 8 in section 4.2.2, indicates the similarity between query and all predicates (facts or circumstance) in the law article applicable to candidate case. To further induce the law-level relevance score the module evaluates relevance through T-norm fuzzy logic.

- Case-Level Logic Rules Module: Uses a pre-trained Sentence-BERT to extract the embeddings of query and case sentences. The case-level logic rule e_C defined by the equation 10 in section 4.3.1, detect relevant facts and circumstances from the query and the case. In this instance, the relevance score is calculated applying the geometric mean to aggregate all sentence pair predictions.

NS-LCR use the fusion module to combine the outputs of all modules r_N, r_L, r_C in order to compute the final ranking score, utilizing the Weighted Reciprocal Rank Fusion (WRRF) introduced in section 4.1 *Fusion module*. Tested on two datasets (LeCaRD and ELAM), NS-LCR significantly improves retrieval accuracy and explainability over baseline models, including Criminal-BERT and Lawformer. Additionally, it shows robustness in low-resource scenarios, making it suitable for legal contexts with limited labeled data.

A.9 LegalBench: A Collaboratively Built Benchmark for Measuring Legal Reasoning in Large Language Models

Guha et al. [4] they proposed a collaboratively constructed legal reasoning benchmark consisting of 162 tasks covering six different types of legal reasoning, specifically: rule recall, issue spotting, rule application, rule conclusion, interpretation and rhetorical analysis. It is built through an interdisciplinary process, in which they collected tasks designed and hand-crafted by legal professionals, ensuring that these problems are both a good measure for legal reasoning capabilities and are practically useful for legal practitioners.
They evaluated and compared 20 LLMs on the average performance over the different LEGALBENCH tasks. The empirical results revealed, as we could expect, the best performance for the family of large commercial models. In particular, the tasks where they achieved outstanding outcomes are: issue spotting and rule conclusion, with scores above 80%. The task with the least grade of GPT-4 was rule recalling with just under 60%. The authors of the paper also underlined the significant variation in performance across tasks, implying that this benchmark captures a various spectrum of challenge.
In section *Limitations and social impact*, they identified limitations in LEGALBENCH. One major problem is that it does not include tasks over long document, which are essential for legal practice and resemble the real life cases. Other issues are the lack of evaluation for multilingual, or non-English, legal tasks and that certain legal domains were treated, representing only a subset of the entire field.