Can LLMs help lawyers? Argument analysis in legal texts

Karla Salas-Jimenez

Université Toulouse Capitole, IRIT, Toulouse, France

Karla-Denia.Salas-Jimenez@irit.fr

Résumé

Le raisonnement juridique dépend fondamentalement de la capacité des professionnels du droit à identifier, formaliser et évaluer les arguments dans les textes juridiques. Cependant, la recherche actuelle repose sur un ensemble limité de ressources annotées et intègre rarement la logique formelle dans l'ensemble de la chaîne d'analyse. Ce travail propose un cadre modulaire qui combine l'extraction d'arguments, la traduction NL-logique et l'évaluation de la force des arguments, en utilisant des stratégies de prompting et des schémas argumentatifs. L'objectif est de faire progresser le raisonnement juridique computationnel en proposant des outils qui améliorent la transparence et la cohérence dans la prise de décision juridique, en créant de nouveaux corpus annotés variés pour soutenir des modèles plus robustes, et en contribuant à des systèmes d'IA qui aident les juristes à construire et à interpréter des arguments juridiques bien fondés grâce à une évaluation basée sur la logique.

Mots-clés

Extraction d'arguments, Raisonnement juridique, Logique déontique défaisable, Programmation par ensembles réponses.

Abstract

Legal reasoning fundamentally depends on the the ability of the legal professionals to identify, formalize, and evaluate arguments in legal texts. However, current research relies on a limited set of annotated resources and rarely integrates formal logic into the full analysis pipeline. This work proposes a modular framework that combines argument extraction, natural language to logic (NL-Logic) translation, and argument strength evaluation, using prompting strategies and argument schemes. The goal is to advance computational legal reasoning by offering tools that improve transparency and consistency in legal decision-making, creating new and diverse annotated corpora to support stronger models, and contributing to AI systems that help lawyers construct and interpret well-founded legal arguments through Logic-based evaluation.

Keywords

Argument Mining, Legal Reasoning, Defeasible Deontic Logic, Answer Set Programming

1 Introduction

Argumentation plays a crucial role in legal reasoning. Lawyers must construct coherent and logically structured arguments to support their clients' positions, while judges are responsible for evaluating their validity, identifying implicit assumptions, and addressing potential exceptions. Since these tasks are complex and cognitively demanding, automating parts of the argumentative analysis can help improve consistency, transparency, and efficiency in legal decision-making.

Over the past decades, research has explored the intersection between law and computational argumentation. Early work by Mochales and Moens [40] pioneered the automatic identification and classification of arguments in legal texts. Subsequent approaches have incorporated formal and explainable models of legal reasoning. For instance, Collenette et al. [18] proposed an explainable AI framework for legal reasoning applied to Article 6 of the European Court of Human Rights (ECHR), formalizing legal factors using the ADF for kNowledGe Encapsulation of Legal Information for Cases (ANGELIC) methodology within Abstract Dialectical Frameworks (ADFs) [15] and evaluating them in PROLOG to predict ECHR decisions. More recently, the emergence of Large Language Models (LLMs) has opened new possibilities. Trajano et al. [51] employed LLMs to translate natural language arguments into computational representations, while Abdullah et al. [3] analyzed the performance of LLMs in legal argument mining tasks, showing that these models can assist in identifying argumentative components.

However, most existing approaches focus on isolated subtasks and are typically evaluated on a single dataset or closely related corpora, often centered on ECHR decisions [44]. Moreover, many proposals either rely exclusively on symbolic methods under controlled settings or use LLMs without integrating them into a structured logical framework. As a result, there remains a gap between natural language argument extraction and formal, logic-based evaluation of argumentative structures.

To address these limitations this research proposes a system that extracts arguments from legal documents, translates them into a logic representation, and evaluates their internal structure and potential contradictions.

The present work describes a research plan, with preliminary results, that aims at achieving the following

contributions:
1) the creation of new corpora for argument mining and NL-Logic translation in the legal domain, and 2) a practical pipeline combining LLM-based extraction, structured prompting, and formal reasoning. These contributions will provide a methodology for future studies, enabling specialized datasets and models for fine-grained legal argument analysis, achieving consistency and fairness in judicial reasoning.
The paper is organized as follows. Section 2 reviews related work on legal argument mining, the formalization of arguments into logic, and the evaluation of argument strength. Section 3 presents the methodology underlying each module of the proposed pipeline. Section 4 reports and analyzes our preliminary results. Finally, the paper is concluded with a discussion of the findings and directions for future work.

2 Related Work

This section is divided into three parts: *Argument Mining*, *Translation of Arguments to Logic*, and *Argument Strength*. The first two are emphasized, as they constitute the main focus of this study. The last is for future work in order to give the completed idea of the work.

2.1 Argument Mining

Argument mining aims to automatically identify and extract argumentative components and their relations from natural language texts in order to generate structured, machine-processable representations for computational argumentation models [16].
Most works rely on argument schemes [39], which "represent stereotypical patterns of reasoning that capture the inferential relationships between premises and conclusions" [52]. In particular, in the legal domain, Atkinson et al. [7] conducted a study on the impact of Walton's conception of argumentation schemes on AI and Law research. Their work shows that incorporating argumentation schemes helps address normative aspects of legal reasoning by operating over a structured knowledge base, thereby providing richer and more accurate representations.
In the legal domain, research has mainly focused on cases from the European Court of Human Rights (ECHR). Poudyal et al. [44] analyzed 20 Decisions and 22 Judgments of the ECHR, noting that Decisions are more concise (≈3,500 words) than Judgments (≈10,000 words) and contain, on average, 18 arguments, while Mumford et al. [41] compiled legal cases under Article 6. Additional datasets from related domains have also been introduced, for example: EthiX [52], built from ethical debates in 22 ethical topics on the Kialo[1] platform, contains 686 arguments categorized into eight ethical classes based on argument schemes. NLAS-multi [49] is a large corpus of 3,810 arguments spanning 20 argumentation schemes and 50 topics, including several legal-related issues, all generated using GPT-4 and Araucaria [46], developed at the Arg-tech Centre, annotates arguments at the level of claims and premises. It includes texts from newspapers, parliamentary records, court reports, magazines, and online sources, covering multiple countries and topics such as human rights and climate change.
Approaches to legal argument extraction range from traditional NLP and machine learning methods [38] to a growing set of LLM-based techniques. Recent work has explored how different prompting strategies shape LLM performance, including studies on clause classification [55], argument identification [22], relation detection [30], and automated debate construction [25]. In parallel, symbolic systems such as ANGELIC [9], applied to ECHR outcome prediction [18], have demonstrated strong performance relative to purely data-driven approaches. More recently Kong et al. [37] proposed to use the LLMs as annotators, using a third LLM as a mediator.
This study also builds on the modular LLM-based annotation pipeline introduced by Berghegger et al. [12], that we describe in Section 3.1.

2.2 Translation of Arguments to Logic

Translating parts of the human thought process, particularly arguments, into formal representations is a highly challenging task. Multiple logical formalisms are available, and a natural question is which of them is most suitable for a given application. One of the most widely used formalisms is First-Order Logic (FOL).
Early efforts in Natural Language-First Order Logic (NL-FOL) translation were rule-based and difficult to scale [1, 14], whereas modern approaches use LLMs, recent work has advanced NL-FOL translation [53, 45] through in-context learning and prompting strategies.
However, FOL presents important limitations in the legal domain [2]. In particular, it does not naturally capture normative notions such as obligations, permissions, and exceptions. Additionally, reasoning in FOL may raise computational complexity concerns in practical derivations, since it is undecidable in general. For these reasons, alternative approaches have emerged.
Trajano et al. [51] explore the use of LLMs to translate natural language arguments into structured computational representations based on argumentation schemes. Their approach combines scheme classification with retrieval-augmented generation to guide the translation process. Results show that LLMs perform well on simple schemes, while more complex argumentative structures require additional contextual guidance. The study demonstrates the feasibility of leveraging LLMs to connect natural language argumentation with formal reasoning frameworks.
Gupta et al. [34] analyze the formalization of human argumentative reasoning by mapping informal logic into Answer Set Programming (ASP), showing that ASP better supports default reasoning and the management of exceptions. Building on this direction, Governatori [31, 33]

[1] https://www.kialo.com/

[2] During the initial stage of this PhD research, the NL→FOL translation was examined, and this limitations were identified.

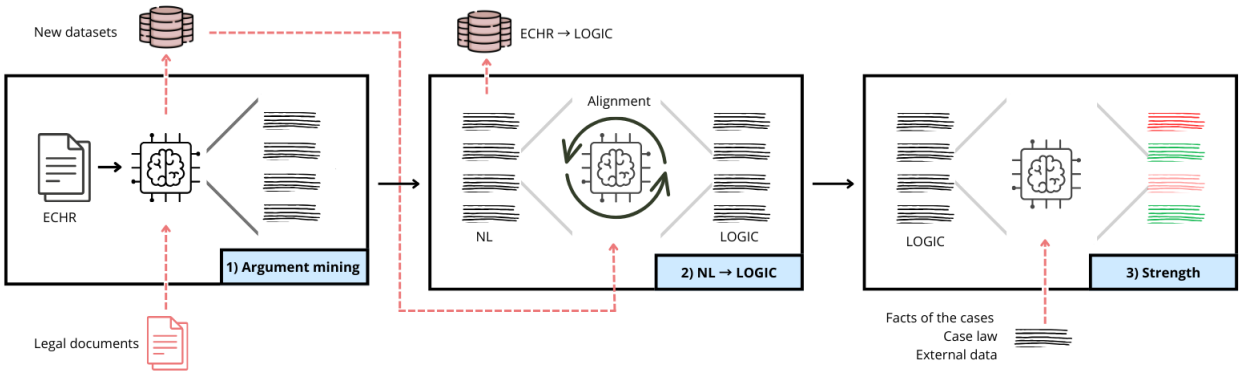

Figure 1: Pipeline from legal text to logical translation and evaluation of arguments strength: (1) argument mining, (2) NL-Logic translation and alignment, and (3) strength evaluation.

first proposed a logic designed to capture key features of the legal domain, the Defeasible Deontic Logic (DDL) which extends Defeasible Logic by incorporating deontic operators.
A DDL consists of three main components:

1. **Facts** - indisputable evidence or conditions of a case.
2. **Rules** - representing legal norms, which can be of three types:
   (a) Constitutive rules: define terms in legal documents.
   (b) Prescriptive rules: assert obligations or prohibitions, potentially organized in compensation chains where the violation of one obligation triggers the next.
   (c) Permissive rules: assert permissions.
3. **Superiority relation** - resolves conflicts between rules.

DDL employs three deontic operators: O (obligation), F (prohibition), and P (permission), which function as modal operators qualifying the truth of propositions. Obligations require a bearer to perform an act or achieve a state, with non-compliance resulting in a violation; prohibitions forbid an act or state, with violations similarly penalized. Permissions hold when no obligation or prohibition forbids an action, with weak permission indicating the absence of restrictions and strong permission representing an explicit exception.
More recently, an implementation of DDL using Answer Set Programming (ASP) as a meta-programming framework has been presented [32], enabling formal reasoning over complex normative structures, employed LLMs to translate arguments into ASP-DDL representations, demonstrating that LLMs can effectively support this task [36].
Given the variety of formalisms available for representing legal norms, Robaldo et al. [47] provide a comparative study of several reasoning frameworks, including ASP, SHACL, DLV, Arg2P, PROLEG, and SPINdle. Their results show that ASP-based reasoners achieve the best computational performance, while also offering a favorable balance between expressivity and ease of representation. Although ASP reasoning is exponential in the worst case [24], modern solvers [27] are efficient in practice. In particular, with bounded arities and restricted domains, we may reduce the problem to propositional ASP where reasoning reduces to NP-complete problem for non-disjunctive programs [20].

2.3 Argument Strength

Argument strength has received much attention in the formal argumentation community [43, 35]. In the field of abstract argumentation [23], arguments strength is usually associated with the acceptability of arguments that can be computed via extension-based semantics (as in Dung's original approach for assessing collective acceptability [23]) or through ranking-based [4, 13] or gradual semantics [17] (when the focus is on individual acceptability). These approaches mainly focus on the "resulting" strength of arguments, *i.e.* how strong they are after facing some conflicts. Some works introduce also a notion of "intrinsic" strength of arguments, using weights to represent how strong are arguments *before* facing the conflicts. In these cases, semantics (*e.g.* extension-based in [48] and gradual in [5]) are adapted to take into account the initial arguments strength for determining their acceptability, *i.e.* their final strength.
In Value-based Argumentation [8], arguments are associated with a value (*i.e.* an abstract label representing a moral or social value), and a preference relation over values allows to assign different strengths to the corresponding argument. The link between Value-based Argumentation and normative reasoning has been emphasized in [11]. However, before using any such approaches for legal reasoning, two challenges must be solved: 1) determining, among the many different approaches for defining the semantics of formal argumentation frameworks, which ones are the best suited for modeling the reasoning schemes of legal practitioners, and 2) how to extract from natural language arguments the formal components used for reasoning (*e.g.* the weights, or the preference order over values). All these works assume that arguments and their relationships have already be obtained (from natural

language or from logical knowledge) and abstracted away. However, arguments strength has also received some attention in structured (logic-based) argumentation and natural language argumentation.

Within the dialectical dimension, Spaans [50] proposes a principle-based approach and concrete methods for computing the intrinsic strength of logic-based arguments from their internal structure. Macagno et al. [39] distinguish premise, conclusion, and inferential (undercutting) attacks, the latter being particularly relevant in legal reasoning.

Beirlaen et al. [10] formalize argument strength through three dimensions: *support* (*e.g.* premises and inference rules reliability), *dialectical* (interactions between competing arguments), and *evaluative* (cumulative support). Approaches for reasoning with abstract argumentation, mentioned previously, focus on the dialectical and evaluative aspects, but ignore the support dimension. For the support dimension, Lenz et al. [38] propose graph-based models labeling edges as support or attack relations.

From a procedural perspective, Gordon et al. [28] introduce the Carneades model, which evaluates arguments through proof standards and dynamically allocated burdens of proof. In this framework, the acceptability of a claim depends not only on its supporting and attacking structure but also on whether it satisfies the applicable proof standard under the current dialectical stage. By distinguishing ordinary premises, assumptions, and exceptions, and by allowing burdens of production and persuasion to shift between parties, Carneades captures an institutional dimension of argumentative strength particularly suited to legal contexts.

Finally, integrating structural and dynamic perspectives, Obermaier et al. [42] show that multiple weak arguments can paradoxically strengthen or weaken a stronger one, highlighting the non-linear dynamics of argumentative influence (in the same vein as the approach by [48]).

3 Methodology

This work proposes a three modules: argument extraction, NL-Logic translation, and strength evaluation. Figure 1 shows the complete pipeline.

3.1 Argument mining

Berghegger et al. [12] propose an LLM-based methodology composed of three pipelines: (1) extracting arguments, composed by the claim and premise(s), directly from the text, (2) extracting claims first and then the complete argument, and (3) extracting the claims, identifying the paragraphs related to those claims, and subsequently identifying the premises within those paragraphs. For evaluation they propose two approaches:

1. **Unstructured evaluation:** Arguments are compared as whole units using vector similarity in two directions (original-to-LLM and LLM-to-original). A match is defined when similarity exceeds a threshold (t = 0.75), and performance is measured using a matching ratio and mean similarity.

2. **Structured evaluation:** Arguments are decomposed into claims and premises. Similarity is computed separately for each component and combined using a weighted formula inspired by similarity measures in logical argumentation [6, 21] ($sim\_arg(A_1, A_2) = \alpha \times simP(P_1, P_2) + (1 - \alpha) \times simC(C_1, C_2)$), where a parameter α (set to 0.7) balances the relative importance of claim and premise similarity.

Their experiments use nine short texts and six long texts from the ECHR dataset tested in *meta-llama/Meta-Llama-3-8B-Instruct*[2] and *Equall/Saul-7B-Instruct-v1*[19]. They also compare their results with traditional ML approaches such as *ArgueMapper/ArgueBuf* [38].

Following this approach, we have run similar experiments on the whole ECHR corpus (Corpus details are in Section 2.1.). Initial results indicate that while LLMs can identify arguments, they often miss key elements, especially in claim detection, and tend to over-generate arguments. As a post-processing step, we introduce a mechanism to merge redundant arguments and refine their structure. To this end, we propose using several LLMs as annotators, and aggregate their results. The arguments extracted by Llama and Saul can be interpreted as alternative annotations of the same legal text. Our objective, in future experiments, is to establish an agreement mechanism between them in order to generate a consolidated final annotation.

Following the findings of Berghegger et al. [12], who report that Llama achieves higher mean similarity scores, we use Llama's arguments as the starting point for the agreement process. In Figure 2 we can observe the agreement process.

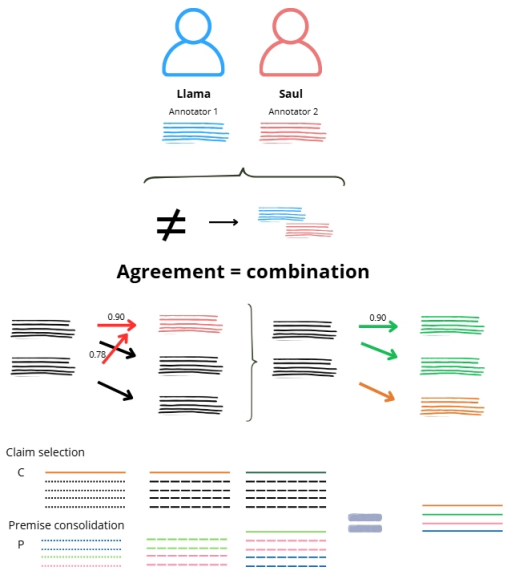

Figure 2: The automatic agreement process between two LLM annotators.

The agreement process consists of four stages:

1. **Matching.** We first matched arguments produced by Saul and Llama using a unstructured similarity threshold of 0.8[3]. This step follows the same procedure employed for matching ECHR arguments with LLM-generated outputs, computing cosine similarity between argument embeddings.

2. **Keep unmatched arguments.** The agreement process is asymmetric, prioritizing LLM1 to preserve its matching ratio and similarity. Its outputs are retained to maintain coverage, while agreement is applied only to matched arguments to reduce redundancy and improve structure.

3. **Make groups.** During matching, it is possible for one Llama argument to match multiple Saul arguments, and vice versa. This may artificially inflate the matching ratio due to highly similar or redundant arguments. To mitigate this effect when multiple matches occurred, we retained only the pair with the highest similarity score. After filtering, we created groups where one Llama argument could correspond to multiple Saul arguments.

4. **Argument combination.** For each group, we constructed a consolidated argument by first determining the claim and then merging the premises.

   (a) Claim selection: all claims within the group were collected and their frequencies computed. If frequencies were equal, the Llama claim was retained (based on prior evidence of higher similarity performance); otherwise, the most frequent claim was selected. Non-selected claims were moved to the premises set.

   (b) Premise consolidation: Similar premises were clustered using the Python string-matching method[4], and one representative premise per cluster was selected, the longest premise containing an article, or otherwise a random one. We intentionally preserve as much information as possible rather than applying aggressive filtering, as providing more context is preferable to omitting potentially relevant content, allowing annotators to determine what is relevant for argument construction.

To enhance the robustness of this module, we evaluated the agreement mechanism under different similarity thresholds. We also incorporated additional LLMs as annotators, such as *Qwen/Qwen2.5-7B-Instruct*, and introduced lexical features to strengthen the structured evaluation process.
Future work includes extending the annotation framework to newly created corpora across different legal domains. Two validation strategies are planned: (1) expert-based evaluation of the extracted arguments to assess their practical relevance, and (2) collaborative refinement aimed at building larger and more diverse annotated corpora.

[3] in 4 the results show that this threshold work better that the others
[4] `difflib.get_close_matches(..., cutoff=0.9)`

Argument	Translation	Attribution
P0 - The applicant considers...	P0 - ...	Applicant
P1 - The Government consider...	P1 - ...	Government
P2 - The Commission finds...	P2 - ...	Commission
P3 - He refers ...	P3 - ...	Applicant
C - There is no ...	C - ...	Undefined

Table 1: Scheme for the manual translation of the ECHR corpus to ASP-DDL.

3.2 Arguments to Logic

The translation methodology presented here is preliminary and part of ongoing work.
To automate the translation we began by refining the annotation guidelines. During this process, we observed that claims and premises frequently begin with attribution markers such as *The applicant considers that*, *The Government submits that*, or *The Commission finds that*. These recurrent formulations introduce additional complexity in the translation step, as they mix argumentative content with speaker attribution.
To address this issue, we separated attribution from the argumentative content by introducing a dedicated *Attribution* attribute and omitting these expressions from the logical translation. The *Attribution* field can take the following values: *Applicant*, *Government*, *Commission*, or *Undefined*.
We then proposed a structured annotation scheme to clarify both the attribution labeling and the subsequent translation process, as shown in Table 1.
For the translation phase, we adopted the ($\approx$ 30) predicates proposed by Governatori [32]. Although annotators were allowed to introduce additional predicates when necessary to capture specific semantic, the use of ASP-DDL predefined predicates was encouraged whenever possible to ensure consistency and formal alignment.
The manual translation process involved two human annotators. Translator 1 holds degrees in Computer Science and Law, while Translator 2 has a degree in Mathematics. This could be interesting for analysing differences between their background in their translations.
To construct the gold standard for each document in the corpus, the annotators conducted video meetings to compare their translations, preserve agreed-upon arguments, and resolve discrepancies in the remaining cases through discussion. An example of the argument translation is provided in Figure 4.
For the automated translation, we adopted a prompt-based strategy testing a zero-shot and few-shot approaches, with step-by-step task instructions to guide the model's reasoning process. Figure 3 shows the prompt template.
We evaluate the prompt by incorporating an explanation accompanied by a brief example of the predicates introduced in ASP to model DDL (<*Explanation of DDL predicates*>), as illustrated in the following example:

permissiveRule(r,x): a legal provision grants a

Template prompt for Argument → ASP-DDL

Your task is to translate the given argument into Defeasible Deontic Logic (DDL) in Answer Set Programming (ASP) syntax, following the exact specifications below.
1. Structure of input:
P0 - <premise 0>
...
PN - <premise N>
C - <claim>
Premises (P0...PN) are factual, legal, or argumentative statements providing support.
The claim (C) is the conclusion supported by the premises.

<Explanation of DDL predicates>
<Argument schemes in ASP-DDL>
<Example(s)>

Now translate the following argument: {}
Output:

Figure 3: Prompt template for translating NL arguments into ASP-DDL.

permission or exception.
Domestic law allows proceedings without an oral hearing in minor cases.
permissiveRule(domestic_law, non(oral_hearing)).

We also include argumentation schemes [29] (<*Argument schemes in ASP-DDL*>) such as *Position to Know* (A claim is presumptively accepted if it is asserted by a source who is in a position to know, unless the source is unreliable.), *Verbal Classification* (If something has a property that entails membership in a category, it can be classified under that category.), *Established Rule* (If the conditions of a valid rule are met and no exceptions or higher-priority rules override it, its conclusion follows.), and *Precedent Case* (If two cases are sufficiently similar and a proposition holds in one, it provides a reason to accept it in the other.). For instance, in the case of verbal classification:

Individual Premise. a has property f.
Classification Premise. For all x, if x has property f, then x can be classified as having property g.
Conclusion. a has property g.

---

constitutiveRule(r_vc, property(a,f)).
applicable(r_vc, property(a,g)) :- property(a,f), property(x,f), property(x,g).

Additionally, we provide examples of translated arguments (<*Example(s)*>). Experiments are conducted under three conditions: without examples, without explanations, and with both explanations and examples. Preliminary results indicate that including examples does not improve performance, LLMs tend to focus excessively on them.
We propose evaluating the translation from two complementary perspectives: first, the semantic similarity between the manual and automated translations, and second, the preservation of the logical structure. A more comprehensive investigation and improved metrics are needed to fully address this goal, but for the present study, we adopt BERTScore [54] to assess semantic similarity.

		Sim.	Gran.	Match.	Mean
Baselines [12]					
Llama		83.42	67.69	72.75	74.62
Saul		75.28	63.48	53.43	64.06
Agreement-based approach					
Llama	0.6	<u>84.24</u>	69.66	82.16	78.69
Llama	0.7	84.15	<u>69.87</u>	81.99	78.67
Llama	0.8	84.00	69.46	<u>84.21</u>	**79.23**
Approach [12] for texts: 33, 35, 38, 40, 41, 42					
Llama		<u>84.29</u>	61.21	56.09	67.20
Saul		82.28	<u>70.43</u>	53.59	68.77
Agreement	0.8	83.24	65.50	<u>75.99</u>	**74.91**

Table 2: Evaluation results for Similarity (Sim.), Granular Similarity (Gran.), Matching Ratio (Match.), and their average (Mean). Baselines from Berghegger et al. [12] (including the missing Saul long-text results) are compared with our agreement-based approach (thresholds 0.6–0.8) with the generalization of this method to six additional ECHR texts. Bold indicates the best Mean score, underlined values denote the best score per metric.

To evaluate logical fidelity, we use a modified version of *LogicSim* [26] adapted to the ASP-DDL syntax. This metric compares a translation x with its reference y across multiple logical dimensions:

$$LogicSim(x, y) = pd + ap + tp + ld + IoU$$

where pd, ap, tp, and ld denote differences in premises, predicates, and logical operators, and IoU measures predicate overlap.

3.3 Arguments strength

Since this research is still in its early stages, we propose a preliminary notion of argument strength based on consistency and absence of contradictions, which can be assessed within an ASP-DDL framework using tools such as Clingo [27]. However, this notion is limited, as argument strength also depends on rule prioritization and contextual adequacy. Therefore, additional legal knowledge (case facts, statutes, and precedents) is required, motivating the use of a RAG-like framework to incorporate external context.
Challenges mentioned in Section 2.3, regarding the methods for extracting relative arguments strength in different legal contexts, as well as the choice of a reasoning approach for evaluating the final strength of arguments, remain to be investigated.

4 Results and analysis

In this section, we present the results achieved in the first steps of this PhD research project, in particular concerning the first two modules.

4.1 Argument mining

The results of the experiments on the short and long texts that we mentioned in Section 3.1, including the results

for the long texts with Saul that were missing in [12], are presented in the first two rows of Table 2. Overall, LLama outperforms Saul across all evaluation metrics. This difference may be partially attributed to model capacity, as LLama (8B parameters) is larger than Saul (7B).

Introducing agreement-based filtering consistently improves performance across all metrics. In particular, all agreement thresholds (0.6, 0.7, and 0.8) yield higher scores than the work of Berghegger et al. [12] which shows an improvement in the methodology. The effect of the threshold varies depending on the metric: a threshold of 0.6 slightly improves overall similarity (column *Sim*), 0.7 yields the highest granular similarity (column *Gran*), and 0.8 achieves the highest matching ratio (column *Match*). When averaging across metrics, the 0.8 threshold obtains the best overall mean score (column *Mean*).

We also evaluated the agreement-based approach on six additional ECHR texts, all of which are relatively short (*i.e.*, containing fewer than ten arguments). These new experiments showing an increase in the matching ratio for these texts, while overall similarity decreases slightly. This suggests that agreement may favor the recovery arguments maintaining the similarity. Interestingly, for these shorter documents Saul achieves better results in granular similarity than LLama, suggesting that future work could explore the agreement mechanism using Saul as the base model.

Importantly, we prioritize improvements in matching ratio over similarity. While similarity measures capture general semantic closeness, the matching ratio more directly reflects the recovery of relevant argumentative components. Therefore, the threshold of 0.8 appears to provide the most desirable balance between structural precision and content coverage.

4.2 Arguments to Logic

For our preliminary experiments, we selected Documents 30 and 41 from the ECHR corpus, as both contain only five arguments. This controlled setting allowed us to compare annotation strategies and identify an appropriate methodology before scaling to the full corpus.

As shown in Figure 4, differences between annotators can be partly explained by their backgrounds: Translator 1 (T1: computer science and law) incorporates more implicit structure and domain knowledge, whereas Translator 2 (T2: mathematics) adopts a more minimal formalization.

As illustrated in Figure 4, the annotators produce different translations of the same argument, which can be partly explained by their respective backgrounds (Section 3.2). Translator 1 (T1), with training in law, makes implicit normative reasoning explicit. In this example, T1 interprets the text as expressing a general requirement of independence and impartiality, together with a special attention in contexts involving specific vulnerabilities such as PTSD. This leads to the introduction of two rules (a general one and a context-sensitive one) and their prioritization via a superiority relation, as well as the representation of lack of independence and impartiality.

> *The applicant submits that, inter alia, the above factors demonstrate a lack, or at least a perceived lack, of independence and impartiality particularly when, as in his case, an important policy issue in respect of Post Traumatic Stress Disorder (PTSD) arose for consideration.*

T1

```
prescriptiveRule(x, court-martial(independence)).
prescriptiveRule(x, court-martial(imparciality)).
prescriptiveRule(y, court-martial(reinforce_imparciality)).
prescriptiveRule(y, court-martial(reinforce_independence)).
superior(y, x) :- important_policy_issue(ptsd).
non(court-martial(independence)).
non(court-martial(imparciality)).
```

T2

```
defeasible(non(independence)).
defeasible(non(impartiality)).
```

Gold

```
prescriptiveRule(x, imparciality).
prescriptiveRule(x, independence).
prescriptiveRule(y, reinforce_imparciality).
prescriptiveRule(y, reinforce_independence).

fact(important_policy_issue(ptsd)).
superior(y, x) :- important_policy_issue(ptsd).
defeasible(non(independence)).
defeasible(non(impartiality)).
```

LLama

```
fact(applicant_submits_factors_demonstrate
     _lack_of_independence),
fact(applicant_submits_factors_demonstrate
     _lack_of_impartiality).
```

GPT

```
fact(important_policy_issue(post_traumatic_
     stress_disorder)).
defeasible(non(independence_impartiality(court_martial))).
```

Figure 4: Example of NL → ASP-DDL translations generated by T1 and T2, compared against the Gold standard, along with the automatic translations produced by GPT and Llama.

By contrast, Translator 2 (T2), with a mathematical background, focuses strictly on explicitly stated content. As the text does not directly express rules or prioritization, T2 encodes only the observable propositions, treating independence and impartiality as defeasible facts and omitting contextual elements such as PTSD, which are not directly operational in the argument structure. The Gold standard aims to balance these approaches by preserving as much relevant information as possible while avoiding unsupported assumptions. In particular, elements such as `court-martial` are excluded, as they cannot be inferred from the argument alone, whereas implicit reasoning structures are retained when sufficiently grounded in the text.

The second part of Figure 4 shows the translations produced by LLama and GPT. Both models struggle to infer implicit implications, tending instead to extract explicitly stated information, similarly to Translator 2. This behavior is more pronounced in LLama, which only

	Model	Document 30		Document 41	
		LogicSim	BERTScore	LogicSim	BERTScore
T1	GPT	0.682	<u>0.858</u>	0.747	<u>0.879</u>
	Llama	**0.828**	0.831	0.736	0.850
T2	GPT	0.591	0.825	<u>0.781</u>	0.866
	Llama	0.719	0.813	**0.791**	0.832
Gold	GPT	0.708	**0.865**	0.714	**0.881**
	Llama	<u>0.809</u>	0.833	0.721	0.843

Table 3: NL→ASP-DDL translation results for Documents 30 and 41, evaluated using LogicSim and BERTScore. *T1* and *T2* denote the two translators, and *Gold* the gold standard. Bold indicates the highest score per metric among T1, T2, and Gold (across GPT and Llama), while underlined values denote the second-best score.

captures the lack of independence and impartiality as factual statements. GPT, in contrast, recovers additional contextual information by referencing PTSD and encodes uncertainty through a defeasible statement. Nevertheless, neither model captures the underlying rule structure, as implications between factors are systematically collapsed into facts.

These observations highlight that the main discrepancies lie at the structural level, particularly in the failure to represent implications. This is further reflected in the evaluation results (Table 3): while BERTScore and LogicSim indicate that the models recover key semantic content, they do not adequately penalize structural or syntactic deviations in ASP-DDL. This claim is based on the structural characteristics of the outputs for Texts 30 and 41, which exhibit patterns similar to those in Figure 4.

This limitation points to two necessary improvements. First, LogicSim should be extended to explicitly account for predefined DDL predicates, rather than evaluating only predicate overlap and logical components. Second, evaluation should include reasoning evaluation, verifying whether the generated ASP-DDL programs produce derivations comparable to those of the Gold standard via syntactic tree construction and subsequent graph similarity analysis and comparing the answer sets adapting metrics like Jaccard. Such an approach would allow us to assess not only textual and structural similarity, but also functional equivalence at the reasoning level.

5 Conclusions

This work introduces a novel modular pipeline for argument analysis in legal texts, offering significant advancements over existing approaches in several key ways. First, it addresses the challenge of limited resources by providing a new method for (semi-)automatic annotation, paving the way to the creation of new, diverse annotated datasets for argument mining. It also provides new annotated data for NL-Logic translation in the legal domain. Furthermore this work explores the connection between LLM-based extraction and formal reasoning through structured prompts, alignment strategies, and argument schemes to test how LLMs can generate coherent and logical representations. This aspect of the study directly addresses the question of how far LLMs can be effectively be applied in law, showing that, even with limitations, they can still provide meaningful support to legal professionals in improving the clarity and transparency of legal arguments.

Finally, the datasets, guidelines, and prompts produced in this study lay the foundation for training future domain-specific models that will be better equipped to perform accurate, explainable, and interpretable legal reasoning.

At this stage, we have presented only the preliminary results of the complete pipeline. As future work, we plan to complete the experimental evaluation following the methodology of [12] and to extend the agreement analysis to additional large language models, such as Qwen (Qwen/Qwen2.5-7B-Instruct). We also intend to incorporate lexical and semantic features to further improve performance in granular similarity metric. Regarding the NL → ASP-DDL translation, we aim to complete the translation of the datasets and conduct a systematic analysis of the correspondence between the natural language texts and their formal semantic representations, in order to establish a gold standard. Based on these results, we will refine the prompting strategies for automatic translation. If performance gains remain limited, we plan to explore the integration of reinforcement learning techniques to further enhance the models' reasoning and translation capabilities.

In conclusion, this work not only offers valuable new resources but also contributes a comprehensive and extensible framework for computational legal analysis, advancing the development of more transparent, reliable, and semantically grounded AI systems for the legal domain.

6 Acknowledgements

This work was funded by the French National Research Agency (grant AIDAL, ANR-22-CPJ1-0061-01).

Experiments presented in this paper were carried out using the OCCIDATA platform that is administered by IRIT and supported by CNRS (https://occidata.irit.fr).

References

[1] Lasha Abzianidze. LangPro: Natural language theorem prover. In *Proc. of EMNLP*, pages 115–120, 2017.

[2] AI@Meta. Llama 3 model card, 2024.

[3] Abdullah Al Zubaer, Michael Granitzer, and Jelena Mitrović. Performance analysis of large language models in the domain of legal argument mining. *Frontiers in Artificial Intelligence*, Volume 6 - 2023, 2023.

[4] Leila Amgoud and Jonathan Ben-Naim. Ranking-based semantics for argumentation frameworks. In *Proc. of SUM*, pages 134–147, 2013.

[5] Leila Amgoud, Jonathan Ben-Naim, Dragan Doder, and Srdjan Vesic. Acceptability semantics for weighted argumentation frameworks. In *Proc. of IJCAI*, pages 56–62, 2017.

[6] Leila Amgoud and Victor David. Measuring similarity between logical arguments. In *Proc. of KR*, pages 98–107, 2018.

[7] Katie Atkinson and Trevor J. M. Bench-Capon. Argumentation schemes in AI and law. *Argument Comput.*, 12(3):417–434, 2021.

[8] Katie Atkinson and Trevor J. M. Bench-Capon. Value-based argumentation. *FLAP*, 8(6):1543–1588, 2021.

[9] Katie Atkinson and Trevor J. M. Bench-Capon. ANGELIC II: an improved methodology for representing legal domain knowledge. In *Proc. of ICAIL*, pages 12–21, 2023.

[10] Mathieu Beirlaen, Jesse Heyninck, Pere Pardo, and Christian Straßer. Argument strength in formal argumentation. *FLAP*, 5(3):629–676, 2018.

[11] Trevor J. M. Bench-Capon and Sanjay Modgil. Norms and value based reasoning: justifying compliance and violation. *Artif. Intell. Law*, 25(1):29–64, 2017.

[12] Christina Berghegger, César Philippe, Karla Salas-Jimenez, Jean-Guy Mailly, Leila Moudjari, and Laurent Perrussel. Discovering the Potential of LLMs in Annotating Legal Texts for Argument Mining. In *Proc. of Arg&App*, 2025.

[13] Elise Bonzon, Jérôme Delobelle, Sébastien Konieczny, and Nicolas Maudet. A parametrized ranking-based semantics compatible with persuasion principles. *Argument Comput.*, 12(1):49–85, 2021.

[14] Johan Bos and Katja Markert. Recognising textual entailment with logical inference. In *Proc. of HLT/EMNLP*, pages 628–635, 2005.

[15] Gerhard Brewka, Stefan Ellmauthaler, Hannes Strass, Johannes P. Wallner, and Stefan Woltran. Abstract dialectical frameworks. an overview. *FLAP*, 4(8), 2017.

[16] Elena Cabrio and Serena Villata. Five years of argument mining: a data-driven analysis. In *Proc. of IJCAI*, pages 5427–5433, 2018.

[17] Claudette Cayrol and Marie-Christine Lagasquie-Schiex. Graduality in argumentation. *J. Artif. Intell. Res.*, 23:245–297, 2005.

[18] Joe Collenette, Katie Atkinson, and Trevor J. M. Bench-Capon. Explainable AI tools for legal reasoning about cases: A study on the european court of human rights. *Artif. Intell.*, 317:103861, 2023.

[19] Pierre Colombo, Telmo Pessoa Pires, Malik Boudiaf, Dominic Culver, Rui Melo, Caio Corro, Andre F. T. Martins, Fabrizio Esposito, Vera Lúcia Raposo, Sofia Morgado, and Michael Desa. Saullm-7b: A pioneering large language model for law, 2024.

[20] Evgeny Dantsin, Thomas Eiter, Georg Gottlob, and Andrei Voronkov. Complexity and expressive power of logic programming. *ACM Comput. Surv.*, 33(3):374–425, 2001.

[21] Victor David, Jérôme Delobelle, and Jean-Guy Mailly. Similarity measures for first-order logical arguments. In *Proc. of NMR*, pages 46–59, 2025.

[22] Adrian de Wynter and Tangming Yuan. "I'd like to have an argument, please": Argumentative reasoning in large language models. In *Proc. of COMMA*, pages 73–84, 2024.

[23] Phan Minh Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artif. Intell.*, 77(2):321–358, 1995.

[24] Thomas Eiter, Wolfgang Faber, Michael Fink, and Stefan Woltran. Complexity results for answer set programming with bounded predicate arities and implications. *Ann. Math. Artif. Intell.*, 51(2-4):123–165, 2007.

[25] Elliot Faugier, Frédéric Armetta, Angela Bonifati, and Bruno Yun. Assisted debate builder with large language models. In *Proc. of ECAI*, pages 4447–4450, 2024.

[26] Francisco Fernando Lopez-Ponce and Gemma Bel-Enguix. Into the limits of logic: Alignment methods for formal logical reasoning. In *Proc. of MathNLP*, pages 112–123, 2025.

[27] Martin Gebser, Roland Kaminski, Benjamin Kaufmann, and Torsten Schaub. Multi-shot ASP solving with clingo. *Theory Pract. Log. Program.*, 19(1):27–82, 2019.

[28] Thomas F. Gordon, Henry Prakken, and Douglas Walton. The carneades model of argument and burden of proof. *Artificial Intelligence*, 171(10):875–896, 2007. Argumentation in Artificial Intelligence.

[29] Thomas F Gordon and Douglas Walton. Legal reasoning with argumentation schemes. In *Proc. of ICAIL*, pages 137–146, 2009.

[30] Deniz Gorur, Antonio Rago, and Francesca Toni. Can large language models perform relation-based argument mining? In *Proc. of COLING*, pages 8518–8534, 2025.

[31] Guido Governatori. Practical normative reasoning with defeasible deontic logic. In *RW Summer School*, pages 1–25. Springer, 2018.

[32] Guido Governatori. An asp implementation of defeasible deontic logic. *KI-Künstliche Intelligenz*, 38(1):79–88, 2024.

[33] Guido Governatori, Francesco Olivieri, Antonino Rotolo, and Simone Scannapieco. Computing strong and weak permissions in defeasible logic. *J. Philos. Log.*, 42(6):799–829, 2013.

[34] Gopal Gupta, Sarat Varnasi, Kinjal Basu, Zhuo Chen, Elmer Salazar, Farhad Shakerin, Serdar Erbatur, Fang Li, Huaduo Wang, Joaquín Arias, et al. Formalizing informal logic and natural language deductivism. In *ICLP Workshops*, 2021.

[35] Jesse Heyninck, Kenneth Skiba, and Matthias Thimm. Preface for the special issue on argument strength. *Argument Comput.*, 14(3):245–246, 2023.

[36] Elias Horner, Cristinel Mateis, Guido Governatori, and Agata Ciabattoni. From legal texts to defeasible deontic logic via llms: A study in automated semantic analysis. In *Proc. of ASAIL@ICAIL*, pages 83–100.

[37] Yuntao Kong, Ye Xiong, Shuyuan Zheng, and Ken Satoh. Reinforcement learning with argument-structured reward for court decision abstractive summarization. In *Proc. of JURIX*, pages 312–317, 2025.

[38] Mirko Lenz and Ralph Bergmann. User-centric argument mining with arguemapper and arguebuf. In *Proc. of COMMA*, pages 367–368, 2022.

[39] Fabrizio Macagno, Douglas Walton, and Chris Reed. *Argumentation Schemes*. Cambridge University Press, 2018.

[40] Raquel Mochales and Marie-Francine Moens. Study on the structure of argumentation in case law. In *Proc. of JURIX*, pages 11–20, 2008.

[41] Jack Mumford, Katie Atkinson, and Trevor J. M. Bench-Capon. Annotated insights into legal reasoning: A dataset of article 6 ECHR cases. *Argument Comput.*, 15(2):113–119, 2024.

[42] Magdalena Obermaier and Thomas Koch. The paradox of argument strength: how weak arguments undermine the persuasive effects of strong arguments. *Scientific Reports*, 14(1):22244, 2024.

[43] Gabriella Pigozzi and Srdjan Vesic. Preface for the special issue on argument strength. *Argument Comput.*, 12(1):1–2, 2021.

[44] Prakash Poudyal, Jaromir Savelka, Aagje Ieven, Marie Francine Moens, Teresa Goncalves, and Paulo Quaresma. ECHR: Legal corpus for argument mining. In *Proc. of ArgMining*, pages 67–75, 2020.

[45] Amin Rabinia and Sepideh Ghanavati. The FOL-based legal-grl (FLG) framework: Towards an automated goal modeling approach for regulations. In *Proc. of MoDRE@RE*, pages 58–67, 2018.

[46] Chris Reed, Raquel Mochales Palau, Glenn Rowe, and Marie-Francine Moens. Language resources for studying argument. In *Proc. of LREC*, pages 2613–2618, 2008.

[47] Livio Robaldo, Sotiris Batsakis, Roberta Calegari, Francesco Calimeri, Megumi Fujita, Guido Governatori, Maria Concetta Morelli, Francesco Pacenza, Giuseppe Pisano, Ken Satoh, et al. Compliance checking on first-order knowledge with conflicting and compensatory norms: a comparison among currently available technologies. *Artificial Intelligence and Law*, 32(2):505–555, 2024.

[48] Julien Rossit, Jean-Guy Mailly, Yannis Dimopoulos, and Pavlos Moraitis. United we stand: Accruals in strength-based argumentation. *Argument Comput.*, 12(1):87–113, 2021.

[49] Ramon Ruiz-Dolz, Joaquín Taverner, John Lawrence, and Chris Reed. NLAS-multi: A multilingual corpus of automatically generated natural language argumentation schemes. *CoRR*, abs/2402.14458, 2024.

[50] Jeroen Paul Spaans. Intrinsic argument strength in structured argumentation: A principled approach. In *Proc. of CLAR*, pages 377–396, 2021.

[51] Guilherme Trajano, Débora C Engelmann, Rafel H Bordini, Stefan Sarkadi, Jack Mumford, and Alison R Panisson. Translating natural language arguments to computational arguments using LLMs. In *Proc. of COMMA*, pages 289–300, 2024.

[52] Elfia Bezou Vrakatseli, Oana Cocarascu, and Sanjay Modgil. EthiX: A dataset for argument scheme classification in ethical debates. In *Proc. of ECAI*, 2024.

[53] Yuan Yang, Siheng Xiong, Ali Payani, Ehsan Shareghi, and Faramarz Fekri. Harnessing the power of large language models for natural language to first-order logic translation. In *Proc. of ACL*, pages 6942–6959, 2024.

[54] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. BERTScore: Evaluating text generation with BERT. In *Proc. of ICLR*, 2020.

[55] Abdullah Al Zubaer, Michael Granitzer, and Jelena Mitrovic. Performance analysis of large language models in the domain of legal argument mining. *Frontiers Artif. Intell.*, 6, 2023.