Dallma: Semi-Structured Legal Reasoning and Drafting with Large Language Models

1. Introduction

This paper introduces Dallma, a framework for hybrid reasoning that integrates logical rules, user input, and text generated by Large Language Models (LLMs) to facilitate the drafting of semi-structured legal documents and arguments.

Large language models (LLMs) have proven to be exceedingly powerful at unstructured drafting and reasoning. Based on prompts, they are able to accomplish a variety of diverse tasks (OpenAI, 2023). They excel especially in free-form reasoning tasks, and are able to synthesize diverse information into a coherent textual response. However, many types of reasoning are semi-structured, as they include specific, logically connected reasoning or argumentation steps. Logical reasoning can be difficult for LLMs (Pan et al., 2023; Nezhurina et al., 2024).

Many legal documents are semi-structured. Legal forms have various fields that need to be filled in, each with their own specific instructions and requirements, frequently causing trouble for laypeople (Macfarlane, 2013). Likewise, the law itself acts like a structure for the argumentation of lawyers and judges (Ashley, 2017; Waterman & Peterson, 1981; Westermann & Benyekhlef, 2023; Westermann, 2023). Judges apply legal criteria in a logical sequence (often stemming from statutory or case law) to decide on a legal case. Lawyers draft arguments, arguing that criteria are fulfilled or not. While the logical reasoning sequence is predefined, there is often considerable freedom in how to argue that a criterion, or open-textured legal concept, is fulfilled or not (Hart, 1957). Several studies have highlighted issues of e.g. hallucination when using LLMs to perform legal tasks (Tan et al., 2023; Dahl et al., 2024; Magesh et al., 2024).

2. Proposed Framework

We describe a framework called Dallma (Document Automation, Large Language Model Assisted) to create tools that combine logical rules, user input and values provided by LLMs. Dallma comprises: **(1)** A template format that allows the encoding of complex legal writing or reasoning tasks as logically connected steps, and the specification of the requirements and source (user or LLM) for each step. **(2)** A logical reasoning system, that is able to traverse the template and decide which step should be carried out next to complete the document. **(3)** A prompting engine, which assembles custom prompts for each step, prompts an LLM and validates the answer. **(4)** An interface for user-interaction with Dallma, allowing the user to provide information, provide additional context upon request of the LLM, validate the outputs of the LLM, and print the final document.

Thus, various legal reasoning tasks could be encoded and executed with Dallma. combining advantages of expert systems and LLMs. Dallma could offer benefits in terms of explainability, by relying on deterministic logical reasoning, and potentially alleviate issues with hallucinations, by splitting reasoning into small, specific and verifiable steps.

We believe Dallma has significant potential to automate legal reasoning tasks, including improving access to justice by providing legal information and assisting laypersons in filling out forms, helping lawyers draft documents, and automating compliance checks and decision writing.

3. Example

Here, we will show an early example of how legal reasoning can be performed with Dallma, based on Article 1971 of the Civil Code of Quebec, which defines when a tenant can be evicted for late rent payments. The article is as follows:

The lessor may obtain the resiliation of the lease if the lessee is over three weeks late in paying the rent or, if he suffers serious injury as a result, where the lessee is frequently late in paying it.

This article thus features two alternative logical "paths" to terminate a lease. The article provides the logical structure of these criteria, but does not specify what is meant by e.g. "frequently late", leaving room for open-textured reasoning.

Figure 1 shows how this reasoning schema can be implemented in Dallma. Values in curly brackets are variables that are determined at runtime. The blue variables are provided by the user, while the yellow variables are provided by an LLM, in this case GPT-4o.[1]. After the user has described the facts, the LLM applies the legal criteria. If it does not have enough information, it asks follow-up questions. The block preceded by the #if clause is only rendered if the LLM

[1] https://openai.com/index/hello-gpt-4o/

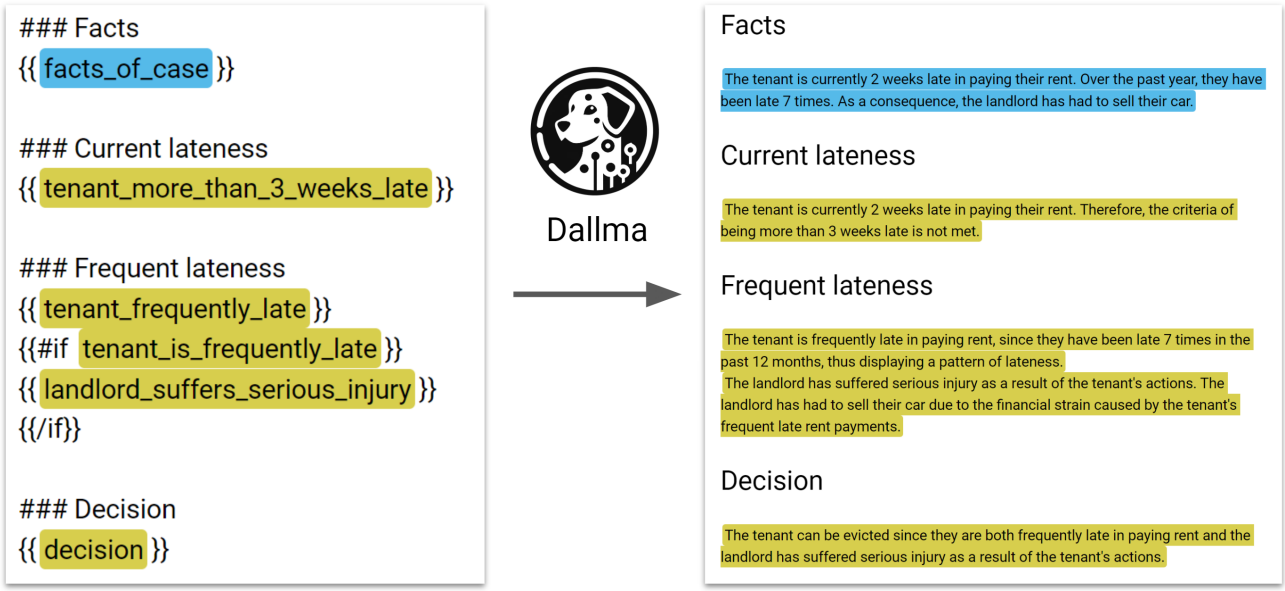

Figure 1. An example of a Dallma and completed document (here with GPT-4o). The template is on the left, the resulting document of a specific run on the right. Blue variables are provided by the user, yellow variables are provided by the LLM.

finds that the tenant is frequently late, thus mirroring the logical structure of the legal article.

4. Related Work

Rule-based reasoning systems, where legal rules are encoded into a computer to conduct legal reasoning, have been developed for many different legal areas (Sergot et al., 1986; Paquin et al., 1991; Branting, 2001; Walker, 2006; Thompson, 2015; Westermann & Benyekhlef, 2023). Likewise, such systems have been used to provide legal information to laypeople and fill out forms, using systems such as Docassemble[2] and A2J Author[3].

LLMs, such as GPT-4 (OpenAI, 2023) have been explored in the legal domain to e.g. answer bar exam questions (Katz et al., 2023), to provide legal information (Tan et al., 2023), to mediate disputes (Westermann et al., 2023), to annotate legal documents (Savelka et al., 2023; Savelka, 2023) and to perform statutory reasoning (Blair-Stanek et al., 2023) and teaching legal concepts (Jiang et al., 2024).

More recently, approaches that combine logical reasoning with large language models have emerged. In the legal field, (Janatian et al., 2023) use GPT-4 to extract a structured representation from a legislative article, which can serve as the basis for legal information tools. (Nguyen et al., 2023) seek to enhance LLM-based reasoning with feedback from logical reasoners. (Steenhuis et al., 2023) uses GPT-4 to automate part of the creation of DocAssemble interviews for legal forms. (Pan et al., 2023) use an LLM to turn a task into a logical representation, and then using a logical system to determine the answer.

Dallma combines logical reasoning systems and LLMs. The logical reasoning system acts as a slot-filler, deciding how to obtain the necessary information or reasoning from a user or LLM. The reasoning method is inspired by that used in docassemble, but adds LLMs as a versatile information processor, able to analyze information, apply reasoning steps or reformulate user answers.

Integrating structured reasoning in LLMs has also been an important subject outside the field of law. In (Wei et al., 2022), the authors describe chain-of-thought prompting, where the model is asked to first create a reasoning plan and then carry it out, leading to improved performance in various tasks. Other approaches rely on directly constraining the outputs of LLMs to specific text, thereby enabling the injection of certain structure into the generated output (Beurer-Kellner et al., 2023)[4]. Compared to these approaches, the system presented here focuses on legal reasoning and drafting, providing a simple template format to encode legal reasoning steps. Further, Dallma provides an interface to allow the end-user to interact with the system by providing information or verifying the LLM-generated answers.

[2] https://docassemble.org/
[3] https://a2jauthor.org/
[4] See also https://github.com/guidance-ai/guidance

References

Ashley, K. D. *Artificial intelligence and legal analytics: new tools for law practice in the digital age*. Cambridge University Press, 2017.

Beurer-Kellner, L., Fischer, M., and Vechev, M. Prompting is programming: A query language for large language models. *Proceedings of the ACM on Programming Languages*, 7(PLDI):1946–1969, 2023.

Blair-Stanek, A., Holzenberger, N., and Van Durme, B. Can GPT-3 perform statutory reasoning? *Proceedings of the Nineteenth International Conference on Artificial Intelligence and Law*, pp. 22–31, 2023.

Branting, L. K. Advisory systems for pro se litigants. In *Proceedings of the 8th international conference on Artificial intelligence and law*, pp. 139–146, 2001.

Dahl, M., Magesh, V., Suzgun, M., and Ho, D. E. Large legal fictions: Profiling legal hallucinations in large language models, 2024.

Hart, H. L. A. Positivism and the separation of law and morals. *Harvard Law Review*, 71:593–629, 1957.

Janatian, S., Westermann, H., Tan, J., Savelka, J., and Benyekhlef, K. From text to structure: Using large language models to support the development of legal expert systems. In *Legal Knowledge and Information Systems*, pp. 167–176. IOS Press, 2023.

Jiang, H., Zhang, X., Mahari, R., Kessler, D., Ma, E., August, T., Li, I., Pentland, A. S., Kim, Y., Kabbara, J., and Roy, D. Leveraging large language models for learning complex legal concepts through storytelling, 2024.

Katz, D. M. et al. GPT-4 passes the bar exam. *SSRN 4389233*, 2023.

Macfarlane, J. The national self-represented litigants project: Identifying and meeting the needs of self-represented litigants final report. 2013.

Magesh, V., Surani, F., Dahl, M., Suzgun, M., Manning, C. D., and Ho, D. E. Hallucination-free? assessing the reliability of leading ai legal research tools. *arXiv preprint arXiv:2405.20362*, 2024.

Nezhurina, M., Cipolina-Kun, L., Cherti, M., and Jitsev, J. Alice in wonderland: Simple tasks showing complete reasoning breakdown in state-of-the-art large language models, 2024.

Nguyen, H.-T., Toni, F., Stathis, K., and Satoh, K. Beyond logic programming for legal reasoning. *Workshop on Logic Programming and Legal Reasoning*, 2023.

OpenAI. GPT-4 technical report, 2023.

Pan, L., Albalak, A., Wang, X., and Wang, W. Y. Logic-lm: Empowering large language models with symbolic solvers for faithful logical reasoning, 2023.

Paquin, L.-C., Blanchard, F., and Thomasset, C. Loge–expert: from a legal expert system to an information system for non-lawyers. In *ICAIL 1991*, pp. 254–259, 1991.

Savelka, J. Unlocking practical applications in legal domain: Evaluation of GPT for zero-shot semantic annotation of legal texts. *arXiv preprint arXiv:2305.04417*, 2023.

Savelka, J., Ashley, K. D., Gray, M. A., Westermann, H., and Xu, H. Can GPT-4 support analysis of textual data in tasks requiring highly specialized domain expertise? *Proceedings of ASAIL'23*, 2023.

Sergot, M. J., Sadri, F., Kowalski, R. A., Kriwaczek, F., Hammond, P., and Cory, H. T. The british nationality act as a logic program. *Communications of the ACM*, 29(5): 370–386, 1986.

Steenhuis, Q., Colarusso, D., and Willey, B. Weaving pathways for justice with gpt: Llm-driven automated drafting of interactive legal applications, 2023.

Tan, J., Westermann, H., and Benyekhlef, K. ChatGPT as an artificial lawyer? *Artificial Intelligence for Access to Justice (AI4AJ 2023)*, 2023.

Thompson, D. Creating new pathways to justice using simple artificial intelligence and online dispute resolution. *IJODR*, 2:4, 2015.

Walker, V. R. A default-logic paradigm for legal fact-finding. *Jurimetrics*, 47:193, 2006.

Waterman, D. and Peterson, M. Models of legal decision-making, r-2717-icj, 1981.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., Le, Q. V., Zhou, D., et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

Westermann, H. Using artificial intelligence to increase access to justice. 2023.

Westermann, H. and Benyekhlef, K. Justicebot: A methodology for building augmented intelligence tools for laypeople to increase access to justice. *ICAIL 2023*, pp. 351–360, 2023.

Westermann, H., Savelka, J., and Benyekhlef, K. LLMediator: Gpt-4 assisted online dispute resolution. *Artificial Intelligence for Access to Justice (AI4AJ 2023)*, 2023.