

# Improving Legal Question Answering through Structured Knowledge Representation

Ankita Gupta<sup>1,2,\*</sup>, Frank Schilder<sup>1</sup>

<sup>1</sup>Thomson Reuters Foundational Research, Minnesota, USA

<sup>2</sup>University of Massachusetts Amherst, USA

## Abstract

Large language models (LLMs) exhibit exciting potential to assist legal practitioners and enhance legal services while reducing costs. However, legal texts present unique challenges for automated processing due to their complex sentence structures, specialized terminology, and fact-intensive nature. These processing difficulties often impact downstream task performance, particularly in question answering scenarios that require careful reasoning about laws and precedents. In this work, we examine whether structuring knowledge in legal texts can improve LLMs' ability to answer legal questions. Our approach prompts LLMs to first generate structured triples of the form *entity-relation-argument* from a given legal text. We then prompt the LLM to answer questions based on these triples, which serve as a structured knowledge representation of the text. Our results demonstrate that this approach improves the performance of small language models like Qwen-3-8B and Llama-3-8B in two settings: a) when the gold passage relevant to the query is given, and b) when passages relevant to the query must be retrieved from a corpus.<sup>1</sup>

## Keywords

Legal question answering, Argumentation, Knowledge graphs, Large language models

## 1. Introduction

Legal professionals worldwide are increasingly enthusiastic about integrating large language models (LLMs) into their workflows to enhance legal services while reducing costs. Such AI tools are already being deployed across various aspects of legal practice, including providing jurisdiction-specific legal information, spotting potential issues in cases, creating legal documents, and numerous other applications.<sup>1</sup>

However, legal texts present unique challenges for automated processing due to their complex sentence structures with multiple subclauses embedded within a single sentence. These texts are also fact-intensive, interleaved with specialized legal terminology, making their comprehension difficult. For instance, prior work has observed limitations in the ability of LLMs to distinguish between arguments made by different legal actors [1].

These comprehension difficulties can significantly impact downstream task performance. For instance, answering legal questions based on a relevant legal text requires the model to carefully understand and reason about laws and precedent cases mentioned in the text and apply them to answer the specific question. The challenge intensifies in real-world scenarios where relevant passages must first be retrieved from extensive legal corpora, as in a retrieval augmented generation (RAG) pipeline [2]. In such cases, imperfect retrieval mechanisms can introduce noisy or only partially relevant passages, further complicating the question answering process.

In this work, we examine whether structuring the knowledge present in the legal text can help improve the LLMs' ability to answer legal questions. In particular, before answering the question, we

<sup>1</sup>Work done during an internship at Thomson Reuters Foundational Research.

*Proceedings of the First Argument Mining and Empirical Legal Research Workshop (AMELR 2025), June 20, 2025, Chicago, United States*

\*Corresponding author.

✉ ankitagupta@umass.edu (A. Gupta); frank.schilder@thomsonreuters.com (F. Schilder)

id 0009-0002-5137-6457 (A. Gupta); 0000-0001-8227-5099 (F. Schilder)



© 2025 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

<sup>1</sup><https://legal.thomsonreuters.com/blog/legal-ai-tools-essential-for-attorneys/>

prompt an LLM to generate triples of the form `entity-relation-argument` from the given legal text(s), where `entity` refers to a legal actor making a statement in the text (e.g., litigants, judge, cited precedents or statutes), `argument` is the information that is expressed by the entity in the legal text(s) and `relation` is how the information is expressed, whether the entity supports or contradicts the information. After extracting these triples, we further prompt the LLM to answer the question based on these triples, which serve as a structured knowledge representation of the input legal text(s).

Our results show that the proposed approach helps improve the performance of small language models, such as Qwen-3-8B and Llama-3-8B, in both settings: a) when the query-relevant passage is given and b) when the passage relevant to the query is not known and has to be retrieved from a given legal corpus. Our work opens several interesting avenues for future work, such as using the proposed approach to reduce misattributions in LLM-generated answers and improving legal reasoning models.

## 2. Related Work

The term knowledge representation refers to various formats that encode structured meaning representations in the form of pre-defined schemata representing entities and properties of these entities. The knowledge is often encoded via `entity-relation-entity` triples (e.g., `capital(France, Paris)`) extracted from plain texts or manually curated, which are used to construct knowledge graphs or databases with special data formats or schemata [3, 4]. Our approach draws inspiration from these methods, although our approach focuses on triples of the form `entity-relation-argument`, extracting arguments supported by the entities as mentioned within legal texts, thus extending the prior approaches beyond entity-level relations. In addition, we instruct the model to create triplets at the time of prompting and hence do not rely on a pre-curated knowledge graph (KG).

### 2.1. General knowledge graphs

KGs are graph-structured knowledge bases that integrate entities and relations from diverse data sources into a unified schema [5]. They typically employ semantic web standards (RDF, OWL) and ontologies to model concepts and relationships, enabling rich semantic queries (e.g., SPARQL) and both deductive and inductive reasoning. However, these KGs require pre-defined schemata, and it is impossible to anticipate every type of relation and property encountered in a lawsuit. KGs are also difficult to maintain and may be too rigid for legal reasoning use cases.

### 2.2. Legal knowledge graphs

Work in the domain of legal KG is closer to our approach because it often aims to structure case law, statutes, regulations, and related documents into a semantic network. Nodes represent legal entities (courts, cases, laws, legal concepts) and edges capture relations such as citations, amendments, or topic hierarchies. Legal KGs often build on domain-specific ontologies (e.g., LKIF Core,<sup>2</sup> LegalRuleML<sup>3</sup>) to model norms, actions, and agents. Similarly to general-purpose KG, they tend to be cumbersome to maintain, although recent work has shown the utility of combining KG modeling Chinese criminal statutes and historical cases, achieving much higher law-article recommendation accuracy [6]. Similarly, Li et al. [7] propose the automated construction of a Chinese legal KG by fine-tuning a large language model with legal prior knowledge, yielding a KG of thousands of legal triples.

### 2.3. Structured representations for prompting LLMs

Structured data (graphs, tables, schemata) are increasingly used to guide large language models, especially in domains like law that demand precise reasoning. Motivated by Chain-of-Thought (CoT) approaches [8], recent work has used more structured data in the prompts, showing improvements

---

<sup>2</sup><https://github.com/RinkeHoekstra/lkif-core>

<sup>3</sup><https://docs.oasis-open.org/legalruleml/legalruleml-core-spec/v1.0/os/legalruleml-core-spec-v1.0-os.html>

for complex problems that require multi-step reasoning. For example, frameworks such as StructGPT [3] leverage structured inputs by interleaving *reading* and *reasoning* phases: the system first queries a knowledge graph or table to collect evidence, then prompts the LLM to reason over that evidence. Hannah et al. [9] design a prompt-reformulation system tied to a legal KG: the KG is queried to generate precise legal citations for issues raised by the LLM’s output, enriching and correcting the response but not modeling the attribution of different statements to the respective legal actors.

Overall, these works suggest that the explicit semantic structure created and retrieved from a KG can improve the responses and even the reasoning capabilities of LLMs. In contrast to other work, we prompt the LLM directly to produce the respective triplet structure of legal entities and arguments, not requiring a well-defined, pre-determined KG of legal concepts.

### 3. Method

We next discuss our method to generate the triples from a given legal text. We extract these triples from the legal text(s) that have been either provided in the input or have been retrieved from a legal corpus to answer the question.

#### 3.1. Triples generation

We prompt an LLM to generate triples of the form *entity-relation-argument* from the given legal text(s). An *entity* refers to a legal actor making a statement in the text (e.g., litigants, judges, precedents, statutes). An *argument* is the information that is expressed by the entity in the legal text(s). An argument can be a clause within a sentence, complete sentences, or even multiple sentences, based on the amount of information provided by the legal actor in the text. Finally, the *relation* describes how the argument is expressed by the entity, i.e., whether the entity supports or contradicts the argument. An example legal text along with extracted triples is shown in Figure 1.

#### 3.2. Answer generation

After extracting the triples from the given legal text(s), we further prompt the LLM to answer the question based on these triples. Our motivation is that allowing the model to represent the legal text in a structured knowledge representation format can help its ability to comprehend and reason over it.

## 4. Experiments and Results

### 4.1. Experimental Setup

#### 4.1.1. Large language models.

We consider small open-source models including Llama-3.1-8B-Instruct and Qwen-3-8B. These models are particularly valuable as they can be downloaded locally and help alleviate privacy concerns. However, the off-the-shelf performance of such models is often not on par with their larger counterparts, necessitating the development of methods to improve their performance.

#### 4.1.2. Datasets.

We consider the Bar Exam QA datasets introduced by Zheng et al. [10] for evaluation. The Bar Exam QA dataset is a dataset of questions from multistate bar exam (MBE), which certifies law students to practice law in the U.S. Each example in this dataset contains a legal scenario, a question about a specific legal issue implicated in the scenario, a gold passage mentioning laws that can help answer the question and four answer choices. The task is to select the correct answer choice. We use the test split of this dataset for our evaluations.

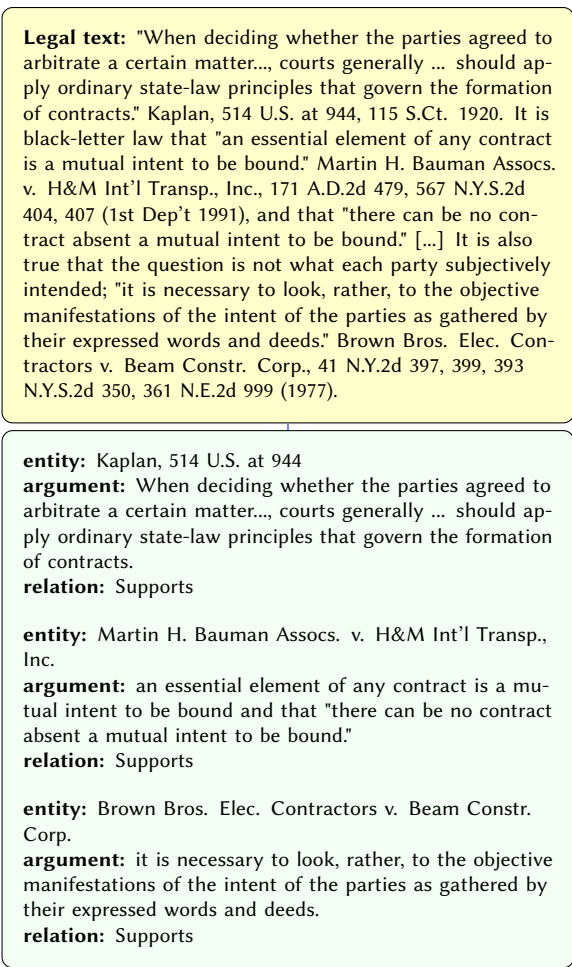


Figure 1: An example of a legal text and corresponding triples.

4.1.3. Evaluation metric.

We report weighted-average F1 scores, calculated over option choices.

4.2. Experiments

4.2.1. Gold passage relevant to the query is given.

We first examine the performance of our proposed approach in a simpler setting, where the legal text that can help answer the query is already known. Thus, in the input, we provide LLM with the relevant legal passage and the query.

4.2.2. Gold passage relevant to the query is not given.

In this case, we simulate a more realistic setting, when the passages useful for answering the legal query are not already known and must be retrieved from a large legal corpus, similar to a RAG setting. To simulate this setting, we consider a simulated retriever, providing the LLM with input containing the gold passage, distractor passages (intended to simulate imperfect retrieval) and the legal query. We randomly order the gold and distractor passages.

Dataset	Models	w/ Triples	w/o Triples
Bar Exam QA	Llama-3.1-8B-Instruct	0.64	0.59
	Qwen-3-8B (no think)	0.54	0.56
	Qwen-3-8B (think)	0.57	0.54

**Table 1**

Weighted-F1 scores for different models with and without triple generation when the gold passage for answering the question is known.

### 4.3. Results

#### 4.3.1. Triple generation helps improve performance.

Table 1 shows the performance of different models when prompted to answer the question with and without an intermediate triple generation step. Both Llama-3.1-8B-Instruct and Qwen-3-8B models benefit from the intermediate triple generation step.

We also observe an interesting negative result for the Qwen-3-8B model in the non-reasoning (*no think*) setting, which does not benefit from intermediate triple generation. This is possible since this mode is specialized for answering simple questions, and the model is not able to utilize the generated triples in its reasoning. Furthermore, anecdotally, we also observe that this model often directly answers the question, ignoring the instruction to generate triples, and thus obtains lower performance compared to the w/o triples setting. As such, this model is not able to incorporate new triple information into their reasoning. Instead, they follow their reasoning and possibly overfit to the type of problem they have been trained on (e.g., solving math or logic puzzles). We also observed similar trends with state-of-the-art reasoning models, such as deep-seek-32B, which always ignores the triple generation instruction and generates its reasoning to answer the question, consistent with concurrent work [11]. Future work can explore methods to improve the instruction-following abilities of such models.

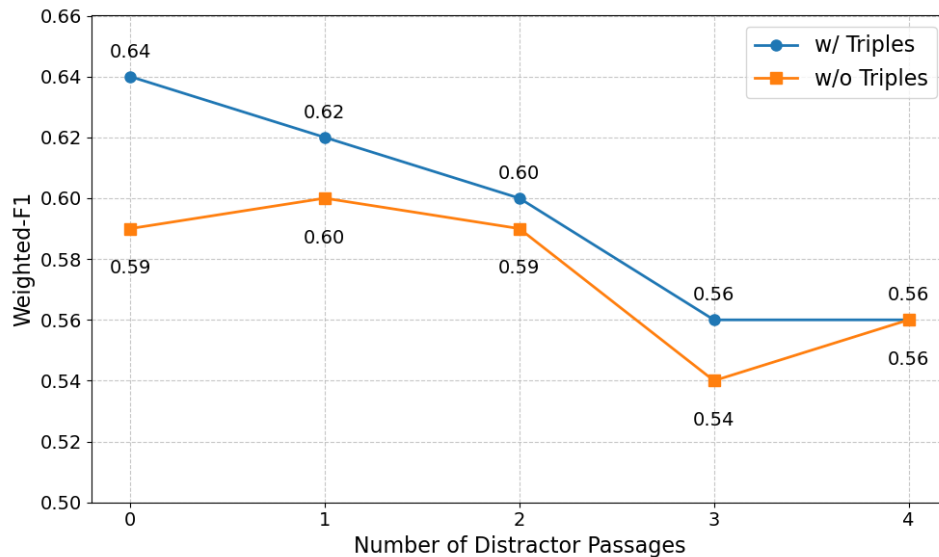
#### 4.3.2. Performance with varying number of distractor passages in the RAG setting.

We further examine the performance of the model when the gold passage is not given in advance and must be retrieved from a corpus. To simulate this setting, in addition to the gold passage, we add a varying number of distractor passages and provide all of them as input to the model. We again test the Llama-3.1-8B-Instruct model’s performance with and without the triple generation step.

As shown in Figure 2, the intermediate generation of triples consistently helps to achieve better performance compared to the setting without triples. As expected, when the number of distractor passages increases, the performance generally declines in both settings, as more noise in the input affects the model’s performance. One potential reason for decreased performance gains is that LLMs struggle to effectively process longer input contexts [12] when more passages are included in the prompt. Nevertheless, the triple generation method maintains its advantage across different distractor levels.

## 5. Conclusion

Our work demonstrates that structuring legal texts as entity-relation-argument triples significantly enhances LLMs’ performance on legal question answering tasks. By decomposing complex legal texts into structured triple representations before answering questions, we enable models like Qwen-3-8B and Llama-3.1-8B to better navigate the inherent challenges of legal language. The effectiveness of our approach across both controlled settings (with gold passage given) and more realistic retrieval settings highlights its robustness and practical utility. This structured knowledge extraction serves as an effective intermediary step that helps models systematically process the complex relationships between legal actors, their arguments, and the underlying legal principles. Future work can explore



**Figure 2:** Performance with varying number of distractor passages in the RAG setting. Performance increases when asking the model to generate triples.

the correlation between the complexity of the legal text (e.g., using measures such as the number of subclauses) and the effectiveness of the triple generation approach for answering questions based on this text to determine which types of complex texts benefit most from the proposed approach. The triple generation approach can also be used for reflection over the generated answer, similar to the self-reflection style prompting approaches [13].

More broadly, our findings open several promising avenues for improving legal AI systems. Future work could explore extending this approach to other legal tasks such as contract analysis, compliance verification, and legal document drafting, as well as investigating how these structured representations might improve explainability and reduce hallucinations in legal AI systems.

## Declaration on Generative AI

During the preparation of this work, the authors used Perplexity/Ollama Llama-3.1-8b in order to: The authors wrote the literature review with the help of AI starting from a set of relevant papers and improved clarity and conciseness of the text using suggestions generated via an LLM. After using these tools/services, the authors reviewed and edited the content as needed and take full responsibility for the publication’s content.

## References

- [1] V. Magesh, F. Surani, M. Dahl, M. Suzgun, C. D. Manning, D. E. Ho, Hallucination-free? assessing the reliability of leading ai legal research tools, *Journal of Empirical Legal Studies* (2024).
- [2] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, S. Riedel, D. Kiela, Retrieval-augmented generation for knowledge-intensive nlp tasks, in: *Proceedings of the 34th International Conference on Neural Information Processing Systems, NIPS ’20*, Curran Associates Inc., Red Hook, NY, USA, 2020.
- [3] J. Jiang, K. Zhou, Z. Dong, K. Ye, X. Zhao, J.-R. Wen, StructGPT: A general framework for large language model to reason over structured data, in: H. Bouamor, J. Pino, K. Bali (Eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Singapore, 2023, pp. 9237–9251. URL: <https://aclanthology.org/2023.emnlp-main.574/>. doi:10.18653/v1/2023.emnlp-main.574.



- [4] H. Li, J. Zhang, C. Li, H. Chen, Resdsql: decoupling schema linking and skeleton parsing for text-to-sql, in: Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence, AAAI'23/IAAI'23/EAAI'23, AAAI Press, 2023. URL: <https://doi.org/10.1609/aaai.v37i11.26535>. doi:10.1609/aaai.v37i11.26535.
- [5] L. Ehrlinger, W. Wöß, Towards a definition of knowledge graphs., in: SEMANTiCS (Posters, Demos, SuCCESS), 2016.
- [6] Y. Chen, M. Chen, Y. Zhu, J. Pei, S. Chen, Y. Zhou, Y. Wang, Y. Zhou, H. Li, S. Zhang, Leverage knowledge graph and large language model for law article recommendation: A case study of chinese criminal law, 2025. URL: <https://arxiv.org/abs/2410.04949>. arXiv:2410.04949.
- [7] J. Li, L. Qian, P. Liu, T. Liu, Construction of legal knowledge graph based on knowledge-enhanced large language models, Information 15 (2024). URL: <https://www.mdpi.com/2078-2489/15/11/666>. doi:10.3390/info15110666.
- [8] J. Wei, X. Wang, D. Schuurmans, M. Bosma, E. H. Chi, Q. Le, D. Zhou, Chain of thought prompting elicits reasoning in large language models, CoRR abs/2201.11903 (2022). URL: <https://arxiv.org/abs/2201.11903>. arXiv:2201.11903.
- [9] G. Hannah, R. T. Sousa, I. Dasoulas, C. d'Amato, A prompt engineering approach and a knowledge graph based framework for tackling legal implications of large language model answers, 2024. URL: <https://arxiv.org/abs/2410.15064>. arXiv:2410.15064.
- [10] L. Zheng, N. Guha, J. Arifov, S. Zhang, M. Skreta, C. D. Manning, P. Henderson, D. E. Ho, A reasoning-focused legal retrieval benchmark, in: Proceedings of the 2025 Symposium on Computer Science and Law, 2025, pp. 169–193.
- [11] X. Li, Z. Yu, Z. Zhang, X. Chen, Z. Zhang, Y. Zhuang, N. Sadagopan, A. Beniwal, When thinking fails: The pitfalls of reasoning for instruction-following in llms, ArXiv abs/2505.11423 (2025). URL: <https://api.semanticscholar.org/CorpusID:278715317>.
- [12] M. Karpinska, K. Thai, K. Lo, T. Goyal, M. Iyyer, One thousand and one pairs: A “novel” challenge for long-context language models, in: Y. Al-Onaizan, M. Bansal, Y.-N. Chen (Eds.), Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Miami, Florida, USA, 2024, pp. 17048–17085. URL: <https://aclanthology.org/2024.emnlp-main.948/>. doi:10.18653/v1/2024.emnlp-main.948.
- [13] N. Shinn, F. Cassano, A. Gopinath, K. Narasimhan, S. Yao, Reflexion: language agents with verbal reinforcement learning, in: Proceedings of the 37th International Conference on Neural Information Processing Systems, NIPS '23, Curran Associates Inc., Red Hook, NY, USA, 2023.