

Modeling the graph of french law

State of the art

PERONNET Antonin

1. Introduction

1.1. The growing complexity of the law

Making the law easier to navigate and to edit is a major challenge, for experts as well as for the general public. Reports about the increasing complexity and volume of the law date back to [1], and French law is no exception. Most legal experts only know a tiny fraction of all the law, thus there is no global vision of the law when it is written. As a consequence, law becomes harder to read, to navigate and to change. There are multiple factors making the law more or less easy to work with. The french government itself has written extensively on the topic [2]. Currently, the main points the french government focus on are:

- How texts are sorted and organised (**codification**).
- Whether they are in sync with the amendments made to them from other documents (**consolidation**).
- what actors are involved in the process of making the law, and what are their motivations while writing new laws (**law-making**).
- how the laws are written and verified (**standardisation**).

The number of different (and potentially conflicting) articles in the law make it harder to enforce, as it demands more time from legal experts to read and learn it. To solve this issue, one potential solution is to reduce the number of articles, and deliberately making the law less precise [3]. But with computers, a new possibility emerges: translating the law to a standard format that can be efficiently treated by computers. This can allow a lot of tasks than are currently limited to human speed to be treated automatically: retrieving relevant documents, checking if a law is in sync with public information, checking for redundancy and obsolete norms.

1.2. Representing the law

The question of how the law should be represented for a computer is a profound and difficult one, because the choice of the representation will influence which features of the law are relevant and which are not. This question first was studied in [4], a foundational work in the field of Philosophy of Law.

All representations of the law are a trade-offs between 2 opposite visions of the law:

- a strict, **algorithmic** law. In this vision, the law is a process that decides if someone is guilty, and compute the judiciary decision. The law is then the source code of the algorithm.
- a flexible, **example-based** law. In this vision, the law is the collection of all past decisions, and they serve as jurisprudence to judge future cases.

The most strict vision of a law as a programming language has been used successfully in specific contexts. For example, the Catala language has been created specifically to translate law into code [5]. Legal programming languages are used especially for taxes, where checking if criteria are respected is easier and more valuable.

But as explained in [6], the approach of a rule-based law is not satisfying because “No area of law is covered exclusively by statute”. Law is full of guideline and vague concepts like “respect human dignity” that are used by judges in novel cases. These concepts can be represented using a formal language based on grammar. In that case, the representation of 2 texts with the same information content should be the same. Such representations exist for text in general and are called **meaning representations**. It is an active area of research, see [7].

A more flexible way to represent legal knowledge (or any kind of knowledge about the world) is called an **ontology**. An ontology is a set of **entities** linked by **relations**, which can be of different types. In this view, law is a graph constituted by many legal categories interacting with each other. To use such an ontology, one must define the set of entity and relation types, what is called a **schema**. There have been multiple proposals to create such a schema, the first one being [8]. With time, legal documents have become very specialized depending on the domain they are in, leading to different ontologies. Ontologies/schemas exist for multiple subfields of the legal domain [9], but unifying them is still an open problem [10].

At the end of the spectrum, there is pure text. With advances in Natural Language Processing (NLP), this medium has been preferred for a wide variety of tasks because of its flexibility. But note that this flexibility is also the reason why the way law is written is not standardized. Researchers in NLP often see the question of representing the law as a one-way problem, but it is really a two-way problem: at one end, lawmakers express what is allowed or not, and systems at the other end must use this knowledge in a consistent way.

2. Existing solutions to treat legal information

Let's take a step back and review the history of how AI systems have treated information in the legal domain.

2.1. Expert systems

The first algorithms that processed legal documents were very specific and rule-based [11]. One very important limitation of these systems was knowledge about the real world.

One of the first systems that integrated knowledge about the real world (not only for legal tasks) for question answering was the Watson project [12]. The algorithms for extracting information were manually crafted by humans, to construct ontologies. The algorithms evolved with NELL [13] and openIE [14], more statistical models that learned rules to extract information. But with the rise of deep learning and Large Language Models (LLM), people realized it was easier to answer questions directly by memorizing or retrieving documents than to construct a knowledge base.

2.2. Embedding knowledge in LLMs

Since GPT2 [15], an increasing number of tasks in natural language processing is solved by pre-training large language models (LLMs) to predict text data, and then training them specifically on specific tasks. We will call this approach **expert LLMs**. This has been done for the law by training models directly on law corpus. The most recent model of this type is SauLM [16], trained on a large corpus of English law and discussions about the law.

As this approach does work to answer easy questions, this approach has a number of problems: the model always suffers from hallucinations, and facts cannot be edited (when a law changes for example). In addition, a lot of training data is required both to train the base model (to have a general understanding of the law) and for the specific problem being solved. In addition, the answers of the model are not easily explainable.

2.3. Retrieval augmented generation

A more recent approach is Retrieval Augmented Generation (RAG). It is typically used to answer questions from a user using a specific corpus. The user makes a query in natural language, and a distance between this query and law articles is calculated using a given metric. Then, the k nearest articles are given as inputs to a LLM, that is asked to generate an answer to the user.

Since the knowledge is not only contained in the LLM, it is easier to update it. But this approach is limited by the size of the context window (the number of words a LLM can read) and by the quality of the metric. This metric is often very sensitive to the style of the text for example. This does not allow for granularity (one part of an article might be very relevant but not all of it).

This approach and the previous one require a LLM trained on the language of the law being considered. Unfortunately, french LLMs are a lot less developed, with a few exceptions. [17] and [18] use the BERT architecture on french documents, and [19] is a decoder-only model trained on English and french.

There are currently no LLM for french legal texts specifically.

2.4. Seeing the law as a graph

Beyond the LLM-based approaches, a lot of advancements have been made by viewing the law as a graph.

Some work use the citations between the articles to understand the relationship between the codes, as in [20] and [21]. Unfortunately the analysis is restricted to the french codes, and the conclusions are not very surprising. In a similar way, [22] use text similarity techniques to identify citations between cases in EU proceedings, and use the resulting graphs to compare types of cases.

Graph approaches have also been used to improve the performance of some tasks. [23] shows the performance improvement of using the citations between law documents for legal classification, and [24] uses graphs in the task of **legal judgement prediction** on a Chinese dataset.

In these approaches, using graphs helps to capture relations between law articles that would not be possible by using only text. Being able to manipulate directly the relations between law concepts could enable:

- creating a distance between articles that has good properties, in order to retrieve all laws that involve some particular concept.
- making explicit the complex dependencies and relations between law articles, in order to assist lawmakers to review it and make it evolve.
- automatically tagging a new law while it is written, which may help a lawmaker see what the law is interacting with.
- potentially detecting inconsistencies between laws.

More broadly, it would allow more approaches that combine the general knowledge of LLMs and the specific rules of expert systems. In particular, we focus on this goal: as a new law is written, being able to identify automatically what the relations with all the other existing laws are.

Unfortunately, the current approaches mainly use explicit or implicit citations between law articles as the edges of the graph. Indeed, such links are easily accessible, for example the official website for the french law Legifrance [25] allow to explore these links. But in practice, citation network is very limited compared to a complete ontology.

This motivates the question: is it possible to generate this graph representation from the text corpus directly ? How to **model the graph of french law** ?

More precisely, the task can be stated in this way (we will precise it below): how to extract an ontology from a legal corpus in an unsupervised or weakly supervised way ?

3. Problem statement

3.1. Definitions

3.1.1. Graph

A (homogeneous) graph is a set of vertices V with a set of edges $E \subset V \times V$.

In a **heterogeneous graph**, there can be multiple types of edges $E_0, E_1 \dots E_k \in V \times V$

3.1.2. Ontology induction

An ontology is a kind of heterogeneous graph. It consists in a set of entities (the vertices) and a set of labeled relations between these entities (the edges). In addition, the ontology often has a **schema**, a kind of typing system which adds constraints on the relations and which can be more or less complicated. As there are a lot of different approaches to express the constraints, we will not describe them here.

Ontology induction is the process of generating an ontology from unstructured data like text. It consists in 2 main tasks:

- **named entity recognition** (NER), finding the relevant entities in the text and identifying when they are the same.
- **relation extraction** (RE), finding the relations between these entities.

See [26] for a survey.

3.1.3. Supervised, unsupervised, transfer learning

Depending on the way data is collected and used, machine learning methods can be of different types:

In **Supervised learning**, models are trained on labeled data $D = \{(x_i, y_i) \in X \times Y\}$, meaning the input data is paired with the correct output. The algorithm learns an approximation function $f : X \rightarrow Y$ to map inputs to the desired outputs by minimizing a loss function.

In **unsupervised learning**, the models use unlabeled data $\{x_i \in X\}$. The algorithm learns a representation of data (see Section 3.1.4), that can be used in tasks where labeled data is not easily available.

In **transfer learning**, a model is trained on labeled data on some distribution and is asked to generalize on cases $D' = \{x'_i \in X\}$ that are very different from the original distribution. This is the same task, but in a different context. In the context of very general models like LLMs, the term “zero-shot learning” is also used when the model is tested on a completely novel task, without specific data [27].

Once a model has been trained to solve a task on a specific dataset D_0 , people may want to adapt it to solve a novel task by using it on a new dataset D_1 . This adaptation can be:

- **supervised**, by re-training the model on labeled data $D' = \{(x', y')\}$ (we will use the expression “weakly supervised” if the labeled data is a very small dataset)
- **unsupervised**, by retraining the model on unlabeled data $D' = \{x'\}$
- **transferred**, with no additional data.

3.1.4. Representations and embeddings

A **representation** of a source text is a computer representation of the source text that:

- 1) is a compression of the text
- 2) allows to measure some kind of distance (or similarity) with another representation.

Representations are often used for words. There are two main types of representations:

- **discrete** or **categorical** representations, where a word w_i is represented by a set of properties S_i
- **continuous** representations where a word w_i is represented by a vector $v_i \in \mathbb{R}^n$. They are often called **embeddings**

See [28] for more detail.

3.2. Formalisation

Formally, our task is the following:

- input:
 - our legal corpus: a set of texts t_i , each containing a sequence of words $w_{i,j}$
 - a set of seed entities E_0
 - a set of relations R
 - a set of seed relations $\{(e_1, r, e_2)\}$ with $e_1, e_2 \in E$ and $r \in R$
 - external legal knowledge: a set of texts d_i containing knowledge **about the law**, like a dictionary of the law concepts.
- output:
 - a set of entities E
 - for each word $w_{k,j}$, an entity $e_{k,j} \in E$ the words refers to
 - a set of relations between entities $\{(e_1, r, e_2)\}$

3.3. Specificity of the task

Our problem is novel because of the following properties:

- the information extraction part is unsupervised
- it relies on external legal knowledge as input. [29] is one of the only approach using external knowledge, see below.
- it is very granular, as each word is given a legal entity. This is different from article classification.
- the language used is french

Because of the novelty of the task, one important part of our work will be to **evaluate** our methods. This can be done by creating benchmarks or scores with good properties. Creating good evaluation is also useful to serve as baselines for subsequent works. See Section 4.3 for more details.

4. Related work

4.1. Ontology induction

Constructing an ontology of the law from text documents is an unsolved problem. One major difficulty is that understanding the law requires a lot of prerequisite knowledge, that is not written in the law: it is implicit knowledge.

Ontology induction in law is quite different from ontology induction in other domains, because law is both **descriptive** and **normative**. While most domains like medicine only give knowledge about the world (descriptive), law can additionally create definitions, apply them and express duties (normative). Most approaches in the literature focus on extracting descriptive information.

Another major challenge which we already mentioned is the choice of the ontology schema ([9]). Many types of relations have been proposed and may be useful depending on the task, but choosing the best legal ontology schema for a given task is far from solved.

4.1.1. In general

As we mentioned, the default paradigm to process textual information today is RAG, despite its limitations. Thus, work in the domain of ontology induction is a lot less investigated. Still, there are some successful approaches that use state of the art LLMs:

- [30] uses a pre-trained language model to do NER and RE in a pure text format. It is trained on a large inter-disciplinary corpus of data.
- [31] uses constrained text-generation for RE in a similar way.

Unfortunately, these approaches are all supervised. They use a lot of training data from the specific domains of the texts, with ontologies manually constructed by humans as training data. Similar datasets have not been made for legal law articles, in particular in French. In addition, knowing how to extract information for one domain (for example medical) will not transfer to knowing what is relevant in a legal text: transfer learning often fails.

- [32] uses a lot less training data, but the performances vary a lot with the domain (medical, geography ...)

4.1.2. In the legal domain

[33] is the most advanced attempt to construct a legal ontology from a legal corpus. It uses fine-tuning techniques to improve the legal knowledge of a LLM in Chinese's criminal law, and uses it to extract entities and relations from text. The paper manually annotated 460 criminal offenses in the Criminal Code according to a fixed ontology schema, and showed strong results on the test evaluation.

[29] used a different approach to extract entities from a legal corpus. It uses an existing LLM and an external dictionary of word definitions. The ontology is very simple, since it contains only hierarchical entities and no relations. It is more similar to a text annotation task than an ontology induction task.

4.2. Topic modelling and text annotation

A weaker form of ontology induction is **topic modelling**, or **text annotation**. Given a set of documents, the goal is to induce categories (or topics) the documents are related to. The categories can mix entities and relation aspects, which make it harder to build a knowledge graph from it.

[34] tries to create a hierarchy of law types by making experts interacting with an algorithm.

[35] introduces a methodology for legal topic modelling using a model trained on legal data, so that it can be used as baseline.

[36] tries to predict aspects of the US federal law dynamics using classical topic modelling techniques.

[37] uses Word2Vec, a classical word embedding model, to compute a distance between law articles. It is specific to tax law and the task is very precise: renumbering the sections of the document.

A very important question in topic modelling is whether to use **bag of words** or **text embeddings** to represent categories.

- in **bag of words**, the categories exactly correspond to a set of words appearing in the text. It is a **discrete** representation. It is easily interpretable, but it is very limited as it cannot capture relationships between words or words used in different meanings
- in **text embeddings**, parts of the text are directly transformed into vectors, and the categories exist in this vector space. It is a **continuous** representation. This approach can lead to more overfitting and less explainability. For example, it might be sensitive to the style of the articles, which is not relevant.

There are hybrid methods that use discrete and continuous representations at different stages.

One of the most recent techniques to extract concepts from a language model is sparse auto-encoders. It was used in [38] to identify concepts in a text model in an unsupervised way.

4.3. Evaluation

There are 2 main ways to evaluate the quality of the ontology extraction: **intrinsic** and **extrinsic** evaluations. An important part of the research project will be to come up with such evaluation, to measure how successful our approach is.

4.3.1. Intrinsic

An intrinsic evaluation measures the quality of the ontology in itself. This requires to know which properties the ontology should have. For example, an evaluation could measure how much our algorithm detects 2 opposite concepts in the same paragraph, which indicates a form of inconsistency.

This type of evaluation is better because it measures the quality directly, but it can be very hard to implement. In the case of legal ontologies, a “good classification” depends a lot on human preferences and standards, which is costly to collect.

For example, researchers in [39] generated decision trees from legal sentences and evaluated the result using ratings by human evaluators.

4.3.2. Extrinsic

An extrinsic evaluation measures how useful the ontology is for different downstream tasks. For example, how well it can help to predict the charges that apply in various contexts.

It is easier to come up with an extrinsic evaluation, but they are heavier to setup and may be misleading. An ontology may be very useful for certain tasks, but useless for others.

Such evaluations could include legal question-answering [40] and detecting similarities between articles [41].

Bibliography

- [1] D. M. Katz, C. Coupette, J. Beckedorf, and D. Hartung, “Complex Societies and the Growth of the Law,” *Scientific Reports*, vol. 10, no. 1, p. 18737, Oct. 2020, doi: 10.1038/s41598-020-73623-x.
- [2] R. Cassin, J.-M. Sauvé, B. Stirn, H. Toutée, and P. Gérard, “Simplification et qualité du droit.”
- [3] G. Tullock, “On the desirable degree of detail in the law,” *European Journal of Law and Economics*, vol. 2, no. 3, pp. 199–209, Sep. 1995, doi: 10.1007/BF01540806.
- [4] J. Noonan John T., “THE CONCEPT OF LAW. By H. L. A. Hart. Oxford: Oxford University Press, 1961. Pp. viii, 263. 21s.,” *The American Journal of Jurisprudence*, vol. 7, no. 1, pp. 169–177, 1962, doi: 10.1093/ajj/7.1.169.
- [5] D. Merigoux, N. Chataing, and J. Protzenko, “Catala: a programming language for the law,” *Proceedings of the ACM on Programming Languages*, vol. 5, pp. 1–29, 2021, [Online]. Available: <https://api.semanticscholar.org/CorpusID:232110399>
- [6] J. Popple, “Legal Expert Systems: The Inadequacy of a Rule-Based Approach,” *LSN: Legal Information Scholarship (Topic)*, 1990, [Online]. Available: <https://api.semanticscholar.org/CorpusID:14767786>
- [7] Z. Sadeddine, J. Opitz, and F. Suchanek, “A Survey of Meaning Representations – From Theory to Practical Utility,” in *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, K. Duh, H. Gomez, and S. Bethard, Eds., Mexico City, Mexico: Association for Computational Linguistics, Jun. 2024, pp. 2877–2892. doi: 10.18653/v1/2024.naacl-long.159.
- [8] A. Valente and J. Breuker, “A functional ontology of law,” 1994. Accessed: Dec. 09, 2024. [Online]. Available: <https://www.semanticscholar.org/paper/A-functional-ontology-of-law-Valente-Breuker/f9102f2c3749cf5a08367b6073133f1e4981b264>
- [9] V. Leone, L. Di Caro, and S. Villata, “Taking stock of legal ontologies: a feature-based comparative analysis,” *Artificial Intelligence and Law*, vol. 28, no. 2, pp. 207–235, Jun. 2020, doi: 10.1007/s10506-019-09252-1.
- [10] C. Cardellino, M. Teruel, L. A. Alemany, and S. Villata, “Ontology Population and Alignment for the Legal Domain: YAGO, Wikipedia and LKIF,” in *International Workshop on the Semantic Web*, 2017. [Online]. Available: <https://api.semanticscholar.org/CorpusID:4543463>
- [11] A. A. Martino, “Expert systems in law,” 1992. [Online]. Available: <https://api.semanticscholar.org/CorpusID:60429180>
- [12] D. Ferrucci *et al.*, “Building Watson: An Overview of the DeepQA Project,” *AI Magazine*, vol. 31, no. 3, pp. 59–79, Jul. 2010, doi: 10.1609/aimag.v31i3.2303.
- [13] T. Mitchell *et al.*, “Never-Ending Learning,” in *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI-15)*, 2015.
- [14] B. Yu *et al.*, “Towards Generalized Open Information Extraction.” [Online]. Available: <https://arxiv.org/abs/2211.15987>
- [15] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, “Language Models are Unsupervised Multitask Learners,” 2019. [Online]. Available: <https://api.semanticscholar.org/CorpusID:160025533>
- [16] P. Colombo *et al.*, “SaulLM-7B: A pioneering Large Language Model for Law.” [Online]. Available: <https://arxiv.org/abs/2403.03883>

- [17] H. Le *et al.*, “FlauBERT: Unsupervised Language Model Pre-training for French,” in *Proceedings of The 12th Language Resources and Evaluation Conference*, Marseille, France: European Language Resources Association, May 2020, pp. 2479–2490. [Online]. Available: <https://www.aclweb.org/anthology/2020.lrec-1.302>
- [18] W. Antoun, F. Kulumba, R. Touchent, É. de la Clergerie, B. Sagot, and D. Seddah, “CamemBERT 2.0: A Smarter French Language Model Aged to Perfection.” [Online]. Available: <https://arxiv.org/abs/2411.08868>
- [19] M. Faysse *et al.*, “CroissantLLM: A Truly Bilingual French-English Language Model.” [Online]. Available: <https://arxiv.org/abs/2402.00786>
- [20] R. Boulet, “A Network Approach to the French System of Legal codes - Part I: Analysis of a Dense Network.” Accessed: Oct. 01, 2024. [Online]. Available: <http://arxiv.org/abs/1201.1262>
- [21] R. Boulet, P. Mazzega, and D. Bourcier, “Network approach to the French system of legal codes part II: the role of the weights in a network,” *Artificial Intelligence and Law*, vol. 26, no. 1, pp. 23–47, Mar. 2018, doi: 10.1007/s10506-017-9204-y.
- [22] Y. Panagis, U. Šadl, and F. Tarissan, “Giving Every Case Its (Legal) Due - The Contribution of Citation Networks and Text Similarity Techniques to Legal Studies of European Union Law,” *Legal Knowledge and Information Systems*. IOS Press, pp. 59–68, 2017. doi: 10.3233/978-1-61499-838-9-59.
- [23] R. Russo *et al.*, “Graph-Based Approach for European Law Classification,” in *2023 IEEE International Conference on Big Data (BigData)*, Dec. 2023, pp. 1–9. doi: 10.1109/BigData59044.2023.10386684.
- [24] Q. Zhao, T. Gao, S. Zhou, D. Li, and Y. Wen, “Legal Judgment Prediction via Heterogeneous Graphs and Knowledge of Law Articles,” *Applied Sciences*, vol. 12, no. 5, p. 2531, Jan. 2022, doi: 10.3390/app12052531.
- [25] “Legifrance Website.”
- [26] W. Wong, W. Liu, and M. Bennamoun, “Ontology Learning from Text: A Look Back and into the Future,” *ACM Computing Surveys - CSUR*, vol. 44, pp. 1–36, 2011, doi: 10.1145/2333112.2333115.
- [27] T. B. Brown *et al.*, “Language Models are Few-Shot Learners.” [Online]. Available: <https://arxiv.org/abs/2005.14165>
- [28] U. Naseem, I. Razzak, S. K. Khan, and M. Prasad, “A Comprehensive Survey on Word Representation Models: From Classical to State-of-the-Art Word Representation Language Models,” *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, vol. 20, no. 5, Jun. 2021, doi: 10.1145/3434237.
- [29] S. Castano *et al.*, “Enforcing legal information extraction through context-aware techniques: The ASKE approach,” *Computer Law & Security Review*, vol. 52, p. 105903, Apr. 2024, doi: 10.1016/j.clsr.2023.105903.
- [30] X. Wang *et al.*, “InstructUIE: Multi-task Instruction Tuning for Unified Information Extraction,” 2023, doi: 10.48550/ARXIV.2304.08085.
- [31] M. Josifoski, N. D. Cao, M. Peyrard, F. Petroni, and R. West, “GenIE: Generative Information Extraction.” Accessed: Dec. 18, 2024. [Online]. Available: <http://arxiv.org/abs/2112.08340>
- [32] H. B. Giglou, J. D’Souza, and S. Auer, “LLMs4OL: Large Language Models for Ontology Learning,” 2023, doi: 10.48550/ARXIV.2307.16648.

- [33] J. Li, L. Qian, P. Liu, and T. Liu, "Construction of Legal Knowledge Graph Based on Knowledge-Enhanced Large Language Models," *Information*, vol. 15, no. 11, p. 666, Nov. 2024, doi: 10.3390/info15110666.
- [34] B. Waltl, J. Muhr, I. Glaser, G. Bonczek, E. Scepankova, and F. Matthes, "Classifying Legal Norms with Active Machine Learning," *Legal Knowledge and Information Systems*. IOS Press, pp. 11–20, 2017. doi: 10.3233/978-1-61499-838-9-11.
- [35] R. Silveira, C. G. Fernandes, J. Araujo Monteiro Neto, V. Furtado, and J. E. Pimentel Filho, "Topic Modelling of Legal Documents via LEGAL-BERT." Accessed: Oct. 20, 2024. [Online]. Available: <https://papers.ssrn.com/abstract=4539091>
- [36] K. Adusumilli *et al.*, "The Structure and Dynamics of Modern United States Federal Case Law," *Frontiers in Physics*, vol. 9, Jan. 2022, doi: 10.3389/fphy.2021.695219.
- [37] A. Blair-Stanek and B. Van Durme, "AI for Tax Analogies and Code Renumbering." Accessed: Sep. 13, 2024. [Online]. Available: <https://papers.ssrn.com/abstract=3850796>
- [38] A. Templeton *et al.*, "Scaling Monosemanticity: Extracting Interpretable Features from Claude 3 Sonnet," *Transformer Circuits Thread*, 2024, [Online]. Available: <https://transformer-circuits.pub/2024/scaling-monosemanticity/index.html>
- [39] S. Janatian, H. Westermann, J. Tan, J. \vS avelka, and K. Benyekhlef, "From Text to Structure: Using Large Language Models to Support the Development of Legal Expert Systems," *ArXiv*, 2023, [Online]. Available: <https://api.semanticscholar.org/CorpusID:265067125>
- [40] A. Louis, G. van Dijck, and G. Spanakis, "Interpretable Long-Form Legal Question Answering with Retrieval-Augmented Large Language Models." [Online]. Available: <https://arxiv.org/abs/2309.17050>
- [41] T. Charmet *et al.*, "Complex Labelling and Similarity Prediction in Legal Texts: Automatic Analysis of France`s Court of Cassation Rulings," in *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, N. Calzolari, F. Béchet, P. Blache, K. Choukri, C. Cieri, T. Declerck, S. Goggi, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, J. Odijk, and S. Piperidis, Eds., Marseille, France: European Language Resources Association, Jun. 2022, pp. 4754–4766. [Online]. Available: <https://aclanthology.org/2022.lrec-1.509/>