Legal Knowledge and Information Systems
R. Markovich et al. (Eds.)

doi:10.3233/FAIA251575

Automated Creation of a Legal Knowledge Graph Addressing Cases of Violence Against Women

Resource, Methodology and Lessons Learned

Claudia D'AMATO [a] and Giuseppe RUBINI [a] and Francesco DIDIO [a] and Donato FRANCIOSO [a] and Fatima Zahra AMARA [a] , and Nicola FANIZZI [a]

[a] *Computer Science Department, University of Bari Aldo Moro, Italy*

ORCiD ID: Claudia d'Amato https://orcid.org/0000-0002-3385-987X, Fatima Zahra Amara https://orcid.org/0000-0001-8463-0330, Nicola Fanizzi https://orcid.org/0000-0001-5319-7933

Abstract. Legal decision-making requires comprehensive legislative knowledge and up-to-date case information. Legal Knowledge Graphs can enhance accessibility of such information, support semantic querying, and serve as knowledge-intensive components for predictive machine learning applications. To address the limited availability of legal KGs, this work develops a KG focused on cases of violence against women. Two complementary construction approaches are presented: a bottom-up, domain-customized methodology and a novel Large Language Model-based solution. Both integrate data extraction, ontology development, and semantic enrichment using sentences from the European Court of Justice as sources. The resulting KGs are validated through competency questions and show potential to improve access to legal information, enable complex queries, and provide a valuable knowledge backbone for supporting predictive tasks such as case outcome analysis.

Keywords. Knowledge Graph Generation, Large Language Models, Linked Data, Semantic Web, Violence Against Women, Prediction.

1. Introduction

In recent years, the legal domain has witnessed an increasing usage of solutions based on Artificial Intelligence (AI) to effectively manage the legal consultation and decision process [1,2]. Despite the advancement of legal technologies [3], there is a significant gap in the availability of structured and easily queryable resources tailored to the legal domain, such as Knowledge Graphs (KGs) [4]. Current KGs in the legal domain often do not address specific subtle challenges, particularly with regard to interoperability, semantic richness, and adaptability to predictive tasks [5,6].

A legal KG must be able to cope with the specific and intricate writing style of laws and sentences and keep track of their evolution, including exceptions and applicability.

We address the urgent global phenomenon of violence against women[2], a violation of human rights with serious physical, mental, economic and social consequences [7]. Effectively countering this problem requires coordinated actions from governments, civil society, and individuals. Our work contributes by developing advanced tools and knowledge resources, particularly knowledge graphs, to support the analysis of judicial reasoning, facilitate policy evaluation, and improve access to structured case-law data, on gender-based violence. To this end, we introduce a novel Legal KG, specifically designed to address judicial cases related to violence against women, along with the methodologies adopted for its development.

The Legal KG is automatically derived from the jurisprudence of sentences of the *European Court of Human Rights*[3] (ECHR), targeting cases of gender-based violence. It adheres to the FAIR (*Findable*, *Accessible*, *Interoperable*, *Reusable*) data principles. The resource is built upon two methodologies: a customization of the general KG development process to the legal domain [8] (with a bottom-up solution combining data extraction, ontology development, and semantic enrichment through alignment to domain-specific ontologies), and a new approach to automated KG generation, based on the exploitation of *Large Language Models* (LLMs), as recent advancements have opened up new possibilities for automating and improving various phases of ontology development[9], that could be customized to the legal domain. The two methodologies produced complementary results: the former provided more precise outcomes, but is more computationally expensive while the latter is more scalable, but ensured less accuracy. The obtained resource has also been validated via suitable CQs (*Competency Questions*) [10].

Besides being the first result addressing the ECHR case law to cases of violence against women, the Legal KG may serve as a proxy for broader applications. Judicial pronouncements prediction can be thus viewed a link prediction problem within the Legal KG. This approach can be generalized to other domains, provided that legal KGs are generated through the proposed methodologies.

The remainder of this paper is structured as follows: Sect. 2 reviews the state of the art on legal informatics, KGs and ontologies in law. Sect. 3 presents a customization to the legal domain of the bottom-up KG development process and the resulting Legal KG. Sect. 4 presents a new approach for automated KG generation, grounded on the exploitation of LLMs and customized to the legal domain, and the obtained Legal KG. Sect. 5 discusses the results and the effectiveness of the adopted methodologies. Sect. 6 concludes the paper by outlining future research directions.

2. Related Work

This section reviews state-of-the-art work in legal informatics with a focus on KGs, ontologies, and thesaurus initiatives, covering methods for structuring legal data, automated reasoning, and extracting insights from texts.

The *Joint Knowledge Enhancement Model* refines LLMs for improved extraction, applied in building the *Chinese Legal KG* with 3480 triples but no domain ontology reuse [2]. Similarly, a judicial case KG leveraging joint event extraction and Dgraph en-

[2] https://www.who.int/news-room/fact-sheets/detail/violence-against-women

[3] https://www.echr.coe.int/

ables retrieval and case analysis but lacks interlinking [11]. The ManyLaws platform provides advanced legal data access in the EU through analytics and visualization, though without semantic querying [12]. KG-based approaches for automatic document generation improve efficiency and quality but do not enable semantic queries [13].

At the operational level, ontologies support compliance checking by modeling deontic norms and enabling reusable reasoning patterns [14]. An RDF-based graph for fine-grained document retrieval leverages ontologies but requires heavy manual construction [15].

Prominent KG efforts include the Lynx project on GDPR and contract compliance, structuring legal documents and linguistic data [16]. Inspired works process institutional publications for queryability with limited ontology reuse [17], while Vietnamese and other KGs [18,19] similarly show ontology underuse, though some incorporate ontology design patterns.

Finally, ontologies and thesauri have been developed to standardize legal terminology and foster interoperability across jurisdictions [20]:

EuroVoc[4] is a multidomain, multilingual thesaurus provided by the *Publications Office* of the European Union used to classify EU documents into categories to facilitate information searching. It is based on the *Simple Knowledge Organization System*[5] (SKOS) standard that is used to represent and organize concepts and vocabularies.

European Law Identifier[6] is a standard created to identify legislative documents from European states by providing an ontology and metadata. It provides simplified access, exchange and reuse of legislation for legal professionals or citizens, and forms the basis for a representation of the Official Journals of the member states in the Semantic Web[7].

European Case Law Identifier[8] **(ECLI)** defines a standard identifier for European jurisprudence, together with a minimal set of metadata. The identifier provided by ECLI is intended to have an identification code whose structure is common among all member states. This is divided into five parts separated by the colon as in the following example: `ECLI:CE:ECHR:2022:0210JUD007397516`.
ECLI identifies case-law and ELI legislative texts, serving complementary roles. To our knowledge, this is the first KG resource dedicated to legal cases of violence against women, providing SPARQL queryability, FAIR compliance, and integration with existing ontologies and legal thesauri, thereby addressing European legislative specifics and supporting broader legal reasoning and decision-making.

3. Legal Knowledge Graph Construction: Bottom-Up Approach

Advancements in KG engineering have significantly impacted the legal domain by supporting the organization of legislative texts, case law, and regulations, thereby improving legal reasoning, prediction, data integration, and complex querying. Following the

[5] https://www.w3.org/TR/skos-reference/
[7] Major progress towards transparency: a new European legislation identifier: https://ec.europa.eu/commission/presscorner/detail/en/IP_12_1040.

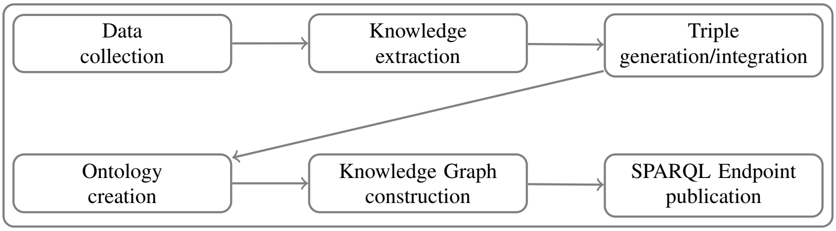

Figure 1. The adopted pipeline.

general methodology proposed by Tamašauskaitė and Groth [8], which distinguishes between top-down and bottom-up approaches, we adapt the bottom-up strategy to cases of violence against women. The process consists of six steps: data collection, knowledge extraction, processing and triple generation, ontology creation, KG construction, and maintenance, with the resulting KG accessible via a SPARQL endpoint. The full pipeline is openly available as a Python project with documentation on GitHub[9].

Regarding **Data collection**, ECHR judgments and decisions, accessible through the official website[10] have been collected and referenced in a compiled document[11]. Specifically, 73 judgments and decisions (in English[12]) have been selected by experts in international law. For each judgement/decision, we specifically accessed tabs `View` and `Case Details` where `View` tab provides the full pdf document, while `Case Details` tab contains additional semi-structured information (in HTML format), such as publication details, importance level, and ECLI identifiers, in agreement with the specific format: `ECLI:EC:ECHR:year`. For example, `ECLI:EC:ECHR:2022:0210JUD007397516` denotes a judgment from February 10, 2022, with appeal number `73975/16`.

Using Selenium 4 Library[13], we set up a functionality[14] for automatic download PDF and HTML files for each url sentence, and have them ready for the successive knowledge extraction step. Indeed, Selenium allows to control a web browser automatically and simulate human actions, such as clicking with the mouse on a specific item.

The **Knowledge Extraction** and **Triple Generation** phases process sentence files to extract triples, using HTML files as the main source. Each sentence instantiates an `ECHRDocument` class. The Beautiful Soup library[15] was employed to extract information, including converting defendant state names to Wikidata URIs and standardizing Importance Level values. From 73 selected judgments (65) and decisions (8), a total of 10,325 triples were extracted, involving 22 predicates and 5,185 entities. These triples were serialized into JSON or processed with RDFLib[16] to produce RDF/Turtle files. Redundancy checks confirmed no duplicates within sentences, while RDFLib automatically removed cross-file duplicates.

[9] https://github.com/PeppeRubini/EVA-KG
[10] https://hudoc.echr.coe.int/
[11] https://github.com/PeppeRubini/EVA-KG/blob/main/data/mapping_doc_link.xlsx
[12] https://hudoc.echr.coe.int/eng
[13] https://selenium.dev/
[14] https://github.com/PeppeRubini/EVA-KG/blob/main/src/echr_scraper.py
[15] https://beautiful-soup-4.readthedocs.io/en/latest/
[16] https://rdflib.readthedocs.io/en/stable/gettingstarted.html

As for the **Ontology Creation** and **KG construction**, in agreement with the ontology engineering best practices, we started with the formalization of CQs[17] that the ontology-based KG should address. The CQs adopted for this purpose are summarized in Tab. 1 (first column). For this phase, we also considered existing ontologies and vocabulary, such as ECLI, for possible reuse for representing ECHR pronouncements. However, not all CQs could have been answered using the existing resources (Tab. 1, second column). New concepts such as `DomesticLaw`, `InternationalLaw`, and `StrasbourgCaseLaw` were introduced, along with properties including `applicationNumbers`, `importanceLevel`, `respondentStates`, `involvedArticles`, and `unanimousDecisionIndicators`. Individuals were created for each ruling under `StrasbourgCaseLaw`, using URI sentences. Overall, metadata from Case Details enriched the ECLI data and addressed most CQs.

The final KG has been constructed by combining the triples from various judgments and linked to external resources including Wikidata for geographic information representation. We also developed three independent visualization solutions by adopting: PyVis Library[18], RDF Grapher[19], and Neo4j[20], respectively.

In order to query the obtained KG, we proceeded with the **SPARQL Endpoint publication**. Specifically, the SPARQL endpoint has been implemented as a Flask[21] web application, leveraging the flexibility and simplicity of the Flask framework for web services. Flask, serves as the web framework for handling HTTP requests, routing, and rendering responses. SPARQL 1.1 engine has been used for executing the SPARQL queries. RDF data is loaded into the RDFLib graph at application startup. The graph supports multiple serialization formats, such as Turtle, RDF/XML, and N-Triples. The application includes two main routes: the home page (`/`) with a form for submitting queries and a query endpoint (`/query`) that processes and displays the results. The interface uses HTML templates: `INDEX.html` for query submission and `RESULT.html` for displaying results. This setup provides a user-friendly way to interact with and query RDF data. The SPARQL endpoint, along with the full implementation of the web application, is available on the GitHub repository[22].

A KG on violence against women was built from 73 European Court of Human Rights judgments, containing 5,185 triples with 22 nodes, 1,747 properties, 5,108 subjects, and 10,325 objects this describes the cleaned KG supporting CQs on document details (type, date, parties, laws) and links to Wikidata for geographic information. The resulting KG, publicly available[23], provides a semantically rich interconnected structure that enables advanced querying and analysis of ECHR decisions pertaining to gender-based violence. It leverages established semantic web standards and vocabularies, enabling interoperability with external resources and datasets. The dataset has been published as part of the *Linked Data* cloud[24], adhering to the FAIR principles (Sect. 5).

---

[17] https://protege.stanford.edu/publications/ontology_development/ontology101-noy-mcguinness.html
[18] https://pypi.org/project/pyvis/
[19] https://www.ldf.fi/service/rdf-grapher
[20] https://neo4j.com/
[21] https://flask.palletsprojects.com/en/3.0.x/
[22] https://github.com/khaoulafatima/PJ4W
[23] https://github.com/PeppeRubini/EVA-KG/blob/main/KG.ttl
[24] The dataset can be found at: https://lod-cloud.net/dataset/PREJUST4WOMAN_PROJECT

Table 1. Competency Questions Adopted for the Bottom-Up Approach.

Question	`Corresponding Term`
What type of document are we dealing with?	`dcterms:type`
When is the document dated?	`dcterms:date`
Can I retrieve the information of a certain case given its identifier?	`dcterms:isVersionOf`
Who represents the applicant?	`dcterms:contributor`
To which European state does the applicant belong?	`respondentState`
What was the ruling?	`dcterms:abstract`
Was the ruling unanimous?	`unanimousDecision`
Which articles of the Convention were considered?	`involveConventionArticle`
Which laws were considered to reach this conclusion?	`dcterms:references`
What is the importance of the ruling concerning future cases?	`importanceLevel`
Is the document publicly accessible?	`dcterms:accessRights`
In which language is the document written?	`dcterms:language`
Where can I consult the document?	`dcterms:identifier`

4. Legal Knowledge Graph Construction: LLM and NLP Pipelines

Recent LLM advances have transformed *Natural Language Processing* (NLP) by enabling ontology development, knowledge graph construction, and query answering. These models enhance accuracy and efficiency in information extraction, addressing limitations of traditional knowledge engineering methods [21]. These models can augment traditional approaches by leveraging extensive training data to infer relationships, identify patterns, and construct meaningful representations of unstructured textual data, such as court rulings and legal documents.

Legal document analysis technology based on LLMs, with the semantic understanding capacity and contextual modeling ability, may effectively tackle these challenges and provide powerful auxiliary tools for [22]. This synthesis of LLM and KG enables better understanding and generation of structured information, addressing long-standing challenges in the field. Developing legal KGs is challenging due to the domain complexity. This approach combines LLMs for ontology generation and data extraction with traditional NLP for structured, precise text processing. Integrating both methods into a unified pipeline enables the creation of accurate, comprehensive KGs aligned with domain requirements and validated using CQs.

Effective *prompt engineering* [23] is essential for leveraging LLMs to produce desired outputs. This approach utilized two key techniques: *zero-shot prompting*, suitable for simple tasks with clearly expressed requirements, and *few-shot prompting*, which includes examples to guide the model in handling more complex tasks and reducing errors like hallucination.

4.1. KG Development

The proposed methodology for generating KGs from legal rulings integrates advanced LLMs and traditional NLP techniques into a cohesive workflow. It involves two pipelines, one of which is an NLP-based pipeline[25]. The LLM pipeline includes document preparation, RAG (*Retrieval Augmented Generation*) development, ontology and KG creation, and competency question validation, while the NLP pipeline focuses on preprocessing, POS tagging, and triple extraction for constructing KGs.

[25] https://github.com/Fra3005/PreJust4Womans

4.1.1. LLM's Pipeline

Document Preparation: The case study focused on analyzing and constructing KG from legal judgments, considering two distinct types of input. The first type utilized the entire judgment (full-text), while the second involved a specific sub-part of the judgment, selected by domain experts. The aim was to assess the effectiveness of both input types. For preparation, automated functions were used to extract text from PDFs for the full-text input. However, due to the varying structure and complexity of each judgment, manual extraction was employed for the sub-part input. Once the text was extracted, it was formatted into a document, ready for the subsequent steps.
RAG's Creation: RAG models have been employed to limit the output of LLMs to the context of each specific document. Although LLMs are trained on vast amounts of data, sometimes they struggle with providing accurate responses to specialized tasks. To address this issue, RAGs enhance traditional LLMs by integrating well-defined corpora [24], enabling LLMs to access two types of data when responding to user queries: parametric data, which refers to the model's training data, and nonparametric data, which consists of new, external data stored in a vector container. The LLM inspects this non-parametric data to generate precise answers. For the creation of RAGs, the TogetherEmbedding method[26] from the LangChain library was used. BERT-M2 served as the embedding model for semantic search within the vector dataset, and FAISS[27], accessed through the langchain_community library, was used to manage the vector dataset. RAGs were created for all five documents involved, which were then used as the contextual basis for the LLM to generate responses to various prompts.
Base's Ontology Creations: At this stage, the ontology, serving as a T-box for constructing the final KGs, was developed. Initially, GPT-4.o was employed to generate a foundational ontology, as Mixtral 8x22b proved incapable of creating a general schema for this topic. Using GPT-4.o resulted in a basic ontology containing core elements such as classes (e.g., *Abuse and LegalCase*) and *ObjectProperty*. Subsequently, the ontology was enriched with additional elements using Mixtral, which, despite explicit instructions in the prompts, tended to add extra components like *DataProperty* and *ObjectProperty* during the KG generation process. To further refine the ontology, a zero-shot prompting technique was applied to domain-specific documents. Prompts instructed the model to expand the ontology by incorporating fundamental concepts from these documents. This process was repeated with several randomly selected documents and, after multiple iterations, it was observed that Mixtral consistently produced similar elements regardless of the input document. This indicated structural similarities among the documents, allowing the newly generated elements to be reused across them, effectively identifying common patterns. Once the ontology was finalized, a thorough review was conducted to eliminate superfluous elements in the schema. This step addressed inconsistencies and redundancies in *DataProperty* elements, ensuring the alignment with the project requirements.
KG Creation: Building on the created ontology and the RAG for each document, a prompt engineering phase was undertaken to generate KGs for each document. This phase utilized a few-shot prompting technique to design effective prompts. Once the individual KGs were generated, they were merged into a unified graph. The merging pro-

[26] `https://python.langchain.com/v0.2/docs/integrations/text_embedding/together/`
[27] `https://python.langchain.com/v0.1/docs/integrations/vectorstores/faiss/`

Table 2. Score CQs answering of Large Language Models Approach.

Results	Full-text	Sub-part
Which legal case is associated with Violation?	3	4
What is the legal outcome of Case?	4	4
What is the reason stated for the judgment?	4	2
What abuse is related to Judgment?	5	4
What is the severity level of Abuse?	2	2
What is the duration and frequency of Abuse?	3	2
Which legal articles are violated?	4	4
What is the context of Abuse?	5	4
Which court judged the Case?	2	1
What are the damages related to Judgment?	1	3
What are the consequences of Abuse?	5	4
How much in legal damages and costs were awarded?	0	4
What is the legal status?	5	3
Total	**40/65**	**37/65**

cess preserved the distinct entities from each document and ensured consistency across the combined KG.
CQ's Creation and Answering: CQs were generated with Mixtral based on the ontology, with a subset selected for use. Mixtral then produced zero-shot answers for these CQs, which were manually verified against the KG and source text to avoid hallucinations. The validated results were stored in a CSV file for further analysis.

4.1.2. NLP's Pipeline

Preprocessing: This step, as described above, was conducted to evaluate and compare new LLM-based approaches. The process was applied to only one of the five previously analyzed documents. Initially, after the document was read and its text extracted, a preprocessing phase was carried out using the NLTK library [28]. This phase involved several steps, including punctuation removal, tokenization, stopword removal, and lemmatization, to prepare the text for subsequent analysis.
Part of Speech: After tokenization, the SpaCy library was used to perform POS tagging for each token. POS tagging is a NLP technique that categorizes the components of a sentence, making it particularly valuable for complex tasks like Sentiment Analysis or Machine Translation. In this study, POS tagging was utilized to extract triples from the document. Tokens were categorized into "subj" (subject), "verb," and "obj" (object) based on their grammatical roles, enabling the creation of structured triples.
Triples Creation: Finally, triples were constructed by associating each identified subject with its corresponding verb and object. This process resulted in a structured representation of the text, capturing meaningful relations between entities within the document.

4.2. Empirical Evaluation and Results

The LLM-based KG generation was qualitatively evaluated using CQs. Across 5 documents with 13 CQs each, full text inputs produced 61.5% consistent answers, while

[28] https://www.nltk.org/

sub-part inputs yielded 56.9%. This shows a trade-off: full-text offers broader coverage, whereas sub-parts provide more focused and relevant results, particularly for factual or procedural queries, offering insight into the KG structure, coverage, and correctness.

This approach assesses LLMs in knowledge engineering tasks such as ontology generation, instance creation, and comparison with traditional NLP methods. LLMs struggled to create complete domain-specific ontologies using only pretrained knowledge, requiring additional documents and expert oversight. While the generated ontologies generally performed well, instances were sometimes incomplete or inaccurate, highlighting the need for human validation. Two specific use cases were explored to evaluate the effectiveness of LLMs in document processing:
1) *Full-text approach:* This use case involved processing the entire document to construct KGs and answer CQs. While this approach provided the highest quantity of responses due to its comprehensive coverage, the relevance and accuracy of the responses varied. Full-text processing captured more nuanced information but required significant manual verification to filter out irrelevant or incorrect outputs. The technical challenges included longer processing times and token size limitations inherent in LLMs.
2) *Sub-part approach*: The approach used expert-selected text segments to focus on the most relevant content. While it generated fewer responses than using the full text, it was faster, avoided token limits, and better aligned with expert objectives, though it relied on careful expert input to avoid missing important information.

The evaluation of the CQs (Tab. 2) emphasized responses aligned with the text, excluding incomplete entities generated solely by the LLM. The resulting KG, available in the GitHub repository[29], structures knowledge from legal and policy documents on women's rights and gender-based issues. It defines 12 classes, 9 object properties, and 17 data properties, outlining the domain's conceptual structure without instances, and serves as a basis for instantiating real-world scenarios and answering CQs.

The CQs were validated for accuracy against source documents. Comparing the two use cases showed that the full-text approach yields more information, while the sub-part approach is better for targeted, domain-specific tasks. This highlights the need to balance data coverage with precision, guided by expert input, to optimize LLM results.

5. Comparative Overview: Bottom-Up vs LLM-Based Approaches

We have proposed two distinct methodologies for constructing KGs based on sentences on cases of violence against women: a bottom-up approach and an LLM-based approach. This section discusses the two approaches, comparing their methodologies, outputs, and suitability for our purpose. A summary of the discussion is provided in Tab. 3.

The bottom-up approach yields precise, domain-specific KGs aligned with resources like ECLI and Wikidata, supporting formal reasoning and SPARQL but demanding significant effort and offering limited adaptability. The LLM-based approach, using zero-/few-shot prompting and retrieval-augmented generation, enables scalable, flexible, and rapid KG creation, though validation is essential to mitigate inaccuracies. Together, they are complementary: bottom-up ensures semantic accuracy, while LLMs support fast prototyping and large-scale processing.

[29]https://github.com/Fra3005/PreJust4Womans/blob/main/Commons/final_onto.ttl

Table 3. Comparative Analysis of Bottom-Up and LLM-Based Approaches.

Criteria	Bottom-Up Approach	LLM-Based Approach
Methodology	Structured pipeline	Combine LLM capabilities with NLP and RAG techniques
Strengths	High precision and semantic alignment; Reliable for domain-specific; Structured tasks	Fast and scalable KG creation
Challenges	Labor-intensive and time-consuming; less adaptable to novel or unexpected patterns	Accuracy issues (hallucination and token limits
Performance on CQs	Excels in precise, structured queries using semantic standards	Generate broader responses but needs post-processing to ensure relevance
Use Case Suitability	Best for formal legal reasoning and SPARQL-based tasks	Ideal for exploratory tasks, large scale processing, and rapid prototyping
Scalability	Limited scalability due to manual data processing	Highly scalable for diverse and large datasets
Output Consistency	Outputs are precise and domain-aligned	Outputs may vary in consistency and require refinement
Ontology Adaptability	Limited to predefined ontologies	Highly adaptable to diverse and nuanced data patterns
Ontology Concepts	Higher initially, with many specific concepts emerging from data	More limited, well-organized in a clear hierarchy
Ontology Relations	More dynamic, based on frequently occurring relations in data	Accurate and based on logical modeling
Ontology Size	Large in size (583.7 KB), including a more extensive set of classes, properties, and relationships	Smaller in size (6.4 KB), suggesting a more concise ontology with potentially fewer classes and relationships

Ontology Development: The project produced two ontologies: one generated with an LLM, which was efficient but generic, and another developed bottom-up, which was semantically richer and aligned with legal reasoning. Their comparison shows the complementary strengths of automation and expert-driven design, emphasizing the need for refinement to ensure legal fidelity. This hybrid approach demonstrates the feasibility of combining structured extraction with LLM techniques for scalable KG generation, supporting broader access to legal data and enhancing interpretability of predictions.
Ensuring FAIRness in Legal Knowledge Engineering: The LKG complies with the FAIR *(Findable, Accessible, Interoperable, Reusable)* principles. It is *findable* via a persistent DOI[30] and its registration in the Linked Data Cloud[31]. It is *accessible* through Zenodo, GitHub, and a public SPARQL endpoint. The KG ensures *interoperability* by using the RDF/Turtle format and standard vocabularies, and is *reusable* under a CC BY 4.0 license with full documentation and source code on GitHub.

6. Conclusion

With this paper, we introduced a novel Legal Knowledge Graph focused on sentences about cases of violence against women. Designed in accordance with FAIR principles, the Legal KG ensures interoperability and possibly reusability across legal and AI systems. To construct the KG, two automated methodologies have been adopted/tailored: a bottom-up approach and an LLM-based approach. Each methodology offers distinct advantages: precision and domain-specific accuracy for the case of the bottom-up solution; scalability and flexibility for the case of the LLM-based approach.

Future work will focus on merging the two KGs into a unified resource, to be possibly expanded, and on extending the KG applicability to the automated detection of legal patterns. Additionally, we are planning to experiment the applicability of the proposed methodologies to other legal domains in order to showcase the generality of the proposed solutions and ultimately contributing to a unified, FAIR-aligned representation of European laws that may enhance legal automation and judicial efficiency.

[30] https://doi.org/10.5281/zenodo.15270173
[31] https://lod-cloud.net/dataset/PREJUST4WOMAN_PROJECT

Acknowledgments

This work was partially supported by project *FAIR - Future AI Research* (PE00000013), spoke 6 - Symbiotic AI (https://future-ai-research.it/) under the PNRR MUR program funded by the European Union - NextGenerationEU, and by PRIN project *HypeKG - Hybrid Prediction and Explanation with Knowledge Graphs* (Prot. 2022Y34XNM, CUP H53D23003700006) under the PNRR MUR program funded by the European Union - NextGenerationEU.

References

[1] Porto FD, Fantozzi P, Naldi M, Rangone N. Mining EU consultations through AI. Artificial Intelligence and Law Article. 2024.

[2] Li J, Qian L, Liu P, Liu T. Construction of Legal Knowledge Graph Based on Knowledge-Enhanced Large Language Models. Information. 2024;15(11):666.

[3] Westermann H, Savelka J, Walker VR, Ashley KD, Benyekhlef K. Computer-Assisted Creation of Boolean Search Rules for Text Classification in the Legal Domain. In: Proceedings of JURIX 2019; 2019. .

[4] Hogan A, Gutierrez C, Cochez M, de Melo G, Kirrane S, Polleres A, et al. Knowledge Graphs. Synthesis lectures on data, semantics, and knowledge. Morgan & Claypool Publishers; 2022.

[5] Wang C, Su W, Hu Y, Ai Q, Wu Y, Luo C, et al. LeKUBE: A Legal Knowledge Update BEnchmark. arXivorg. 2024;abs/2407.14192.

[6] Wang X, Zhang X, Hoo V, Shao Z, Zhang X. LegalReasoner: A Multi-Stage Framework for Legal Judgment Prediction via Large Language Models and Knowledge Integration. IEEE Access. 2024:1-1.

[7] Dawa I, Genene M. Violence against Women. In: Kurtz LR, editor. Encyclopedia of Violence, Peace, & Conflict. 3rd ed. Elsevier; 2022. .

[8] Tamašauskaite G, Groth P. Defining a Knowledge Graph Development Process Through a Systematic Review. ACM Transactions on Software Engineering and Methodology. 2022;32:1 40.

[9] Li J, Garijo D, Poveda-Villalón M. Large Language Models for Ontology Engineering: A Systematic Literature Review. SWJ. 2025. Under review.

[10] Monfardini GKQ, Salamon JS, Barcellos MP. Use of Competency Questions in Ontology Engineering: A Survey. In: Almeida JPA, Borbinha J, Guizzardi G, Link S, Zdravkovic J, editors. Conceptual Modeling. Cham: Springer Nature Switzerland; 2023. p. 45-64.

[11] Zhao B, Zhao Y, Mao Y. A Method for Judicial Case Knowledge Graph Construction Based on Event Extraction. In: Proceedings of the 2024 9th International Conference on Intelligent Information Technology; 2024. p. 62-9.

[12] Stavropoulou S, Romas I, Tsekeridou S, Loutsaris MA, Lampoltshammer T, Thurnay L, et al. Architecting an innovative big open legal data analytics, search and retrieval platform. In: Proceedings of the 13th international conference on theory and practice of electronic governance; 2020. p. 723-30.

[13] Wei H. Intelligent Legal Document Generation System and Method Based on Knowledge Graph. In: Proceedings of the 2024 International Conference on Machine Intelligence and Digital Applications; 2024. p. 350-4.

[14] Francesconi E, Governatori G. Patterns for legal compliance checking in a decidable framework of linked open data. Artificial Intelligence and Law. 2023;31(3):445-64.

[15] Oliveira Fd, Oliveira JMPd. A RDF-based graph to representing and searching parts of legal documents. Artificial Intelligence and Law. 2024;32(3):667-95.

[16] Schneider JM, Rehm G, Montiel-Ponsoda E, Rodríguez-Doncel V, Martín-Chozas P, Navas-Loro M, et al. Lynx: A knowledge-based AI service platform for content processing, enrichment and analysis for the legal domain. Information Systems. 2022;106:101966.

[17] Anelli VW, Brienza E, Recupero M, Greco F, De Maria A, Noia D, et al. Navigating the legal landscape: Developing Italy's official legal knowledge graph for enhanced legislative and public services. In: Falchi F, et al., editors. Proc. of Ital-IA 2023. vol. 3486. CEUR; 2023. p. 223-8.

[18] Vuong THY, Hoang MQ, Nguyen TM, Nguyen HT, Nguyen HT. Constructing a Knowledge Graph for Vietnamese Legal Cases with Heterogeneous Graphs. In: Proc. of IEEE-KSE 2023. IEEE; 2023. .

[19] Sovrano F, Palmirani M, Vitali F. Legal Knowledge Extraction for Knowledge Graph Based Question-Answering. In: Proc. of JURIX 2020. Frontiers in Artificial Intelligence and Applications. IOS; 2020. p. 143-53.

[20] Filtz E, Kirrane S, Polleres A. The linked legal data landscape: linking legal data across different countries. Artificial Intelligence and Law. 2021.

[21] Chen L, Xu J, Wu T, Liu J. Information Extraction of Aviation Accident Causation Knowledge Graph: An LLM-Based Approach. Electronics. 2024;13(19):3936.

[22] Ammar A, Koubaa A, Benjdira B, Nacar O, Sibaee S. Prediction of Arabic Legal Rulings Using Large Language Models. Electronics. 2024;13(4):764.

[23] Ye Q, Axmed M, Pryzant R, Khani F. Prompt Engineering a Prompt Engineer; 2024. Available from: https://arxiv.org/abs/2311.05661.

[24] Lewis P, Perez E, Piktus A, Petroni F, Karpukhin V, Goyal N, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. Advances in Neural Information Processing Systems. 2020;33:9459-74.