LLM-assisted Construction of the United States Legislative Graph

Andrea Colombo
Dipartimento di Elettronica, Informazione e Bioingegneria,
Politecnico di Milano
Milan, Italy
andrea1.colombo@polimi.it

Francesco Cambria
Dipartimento di Elettronica, Informazione e Bioingegneria,
Politecnico di Milano
Milan, Italy
francescoluciano.cambria@polimi.it

ABSTRACT

Graphs are recently being proposed as a powerful alternative to modeling legislation and conducting legislative knowledge management. However, the adoption of such a novel solution depends on the possibility of mapping a fundamentally unstructured source of data within graph objects, namely, nodes, edges, and properties. In this work, we introduce a pipeline for constructing the most extensive Knowledge Graph designed to support the structured representation and analysis of U.S. public laws. The pipeline is based on a recently introduced unifying property graph model, with nodes representing acts and articles and edges capturing their reference links and containing the text of the laws, thus combining semantics, metadata, and content in a structured database. Our pipeline processes all available digitalized U.S. public laws, with older bills available only as image-based PDFs and more recent laws, which instead are published in more modern semi-structured formats such as HTML and XML. To achieve this, we develop an LLM-assisted strategy that also involves fine-tuning LLMs (from the LLama and Mistral families) to extract knowledge from these documents and infer nodes and edge information from low-quality, unstructured texts processed through the popular OCR engine Tesseract. The pipeline's design and the integration of LLMs aims to create the most complete and integrated representation of U.S. legislation, which is crucial for allowing historical and temporal analysis, for instance, from the economic community.

VLDB Workshop Reference Format:
Andrea Colombo and Francesco Cambria. LLM-assisted Construction of the United States Legislative Graph. VLDB 2025 Workshop: LLM+Graph.

1 INTRODUCTION

The complexity of legislative texts as unstructured data and the growing demand for the possibility of conducting analysis have highlighted the importance of creating accessible and structured resources for managing legislative knowledge. In the legislative context, one of the main challenges is navigating the intricate set of references and interdependencies that characterize laws, with recursive patterns linking documents through multiple types of citations, thus raising complexity in the navigation.

An emerging powerful technology for modeling complex and interconnected information is Knowledge Graphs (KGs), especially if the underlying data naturally takes the form of graphs that can be stored within graph databases. Recent works have started applying such technology to model legislative systems by representing laws and their content as nodes and citations as edges [4, 5]. Unifying graph database schemas that aim to model multiple different legislations by capturing the common foundations are also being proposed [15], also favored by the adoption of internationally adopted publication standards, such as the Legal Knowledge Interchange Format (LKIF) [20] or AkomaNtoso [36, 37], which defines common XML schemas to capture the internal structure of laws and acts.

Building graphs of legislation is a powerful way to unlock novel knowledge management applications that allow monitoring patterns and complexities [16]. In addition, it favors economic sciences that rely on network theory to detect trends in the evolution of legislation, which are crucial to understanding the social dynamics of a country [7, 40]. In particular, the Property Graph (PG) model, whose Graph Query Language (GQL) has recently become an ISO standard [22], has been recently proposed as a powerful alternative to modelling legislation, thanks to its ability to combine semantics and properties in a flexible and rich schema, particularly useful to network analysis applications, a unique feature of PGs that distinguishes the data model from the more popular RDF-based graph model.

By following this trend, in this paper, we build a pipeline that allows us to build the most comprehensive Knowledge Graph designed to support the structured representation and analysis of U.S. public laws, i.e., the bills introduced by the federal U.S. Congress. The creation of the graph involved addressing several challenges inherent to legislative data. First, U.S. laws are available in diverse digital formats according to the publication year, from semi-structured XML to unstructured, low-quality images in PDFs, necessitating the development of preprocessing and extraction techniques based on optical character recognition (OCR) engines. Furthermore, the XML schema adopted by the Library of Congress is a US-specific schema, which requires ad-hoc procedures for converting it into graph objects. Then, we tackle the problem of identifying references among acts, an issue further expanded by the presence of the United States Code, a periodically published and updated systematic collection of introduced laws whose instability and content harm the possibility of using it as a bridge to link introduced laws.

To this end, we developed a pipeline that automates the construction of the Knowledge Graph, starting from a document published as PDF images, HTML, or XML. A key element of our pipeline is the integration of open-source Large Language Models (LLMs), to (i) act as legislative experts to correct OCR-induced errors, (ii) extract and infer edges and references by learning from available data, (iii) classify edges into the distinct types of references, namely, *amends, abrogates, is legal basis of* and *cites* references that might link acts

Proceedings of the VLDB Endowment. ISSN 2150-8097.

and (iv) integrate properties of graph objects to integrate helpful information and metadata which are crucial for unlocking powerful graph-based traversal analysis. Finally, we discuss potential applications enabled and facilitated by the availability of a Knowledge Graph of the U.S. legislation.

Our contributions can be summarized as follows:

- We build a pipeline powered by Large Language Models to assist in constructing the U.S. Congress legislative graph. First, we adopt tailored methods to improve OCR performance for more accurate digitization of legislative text. Then, based on the fine-tuning of two specialized Mistral-7B models. One model is tailored for extracting references between legal provisions, while the other focuses on identifying high-level subjects and topics from the text of an act. By leveraging the adaptability and reasoning capabilities of LLMs, our approach enables accurate, scalable information extraction and supports the long-term sustainability and extensibility of the pipeline.
- We construct and share the most extensive Knowledge Graph of U.S. public laws (available at [14]), spanning over 60 years of legislative history, including laws previously existing only as non-machine-readable images, making them accessible and analyzable in a structured digital format.

2 RELATED WORK

Legislative knowledge has traditionally been modeled by utilizing XML schemas to represent the internal structure of documents, with many proposals of XML-based models to handle and unify the differences between legislative traditions. More recently, advancements in knowledge Graph technology have attracted the interest of the computer law community since graph representation can be crucial in capturing the tangled net of connections between laws. As highlighted in [33], legal documents are linked through various relationships, such as amendments and implementations, significantly influencing their legal value. Other works also suggest modeling the data as a graph to enhance the exploration of the different connections between laws [32] and propose graph matching for performing better information retrieval tasks within legal collections [34].

The modeling of such acts within a graph database, which maintains the richness of knowledge as the one proposed in [15], finds natural applications of network analysis techniques. For instance, in [8], authors propose a method for evaluating the similarity between legal documents, significantly improving accuracy by leveraging their citation network. In [10], network analysis is used to evaluate the structure of the French Environmental Code, while in [11], the entire French Legal Code is represented as a graph and analyzed. Concerning the U.S. legislation, [9] evaluates the citation network of the United States Code by examining its degree distribution; however, this work limits the analysis to a version of the United States Code, a collection of laws built by the U.S. bureaucracy and already organized in chapters and do not consider the flow of laws as introduced by the Congress, thus missing the importance of the temporal evolution of the U.S. legislation.

A first effort in modeling U.S. laws into graphs, and in particular, adopting the property graph model, was conducted with the Legis Graph project [25]; however, this effort was very limited on the amount of data that was considered (with laws only from the 114th Congress). Additionally, their focus was on connecting the actions of legislators, committees, and bills in Congress, thus not considering the textual content of the laws and, especially, not modeling the network of citations among acts.

In the rest of the sections, pushed by recent developments in LLMs efficacy in extracting information, we follow the line of efforts in organizing legislative knowledge within a graph model, such as the ones undergoing in Italy [4, 15], and we present a similar resource for the U.S. legislation, by using LLMs to support the KG construction.

3 KNOWLEDGE GRAPH CONSTRUCTION

In this section, we present our approach and pipeline deriving the Knowledge Graph of the U.S. public laws, published on a federal level. First, we outline the underlying database model, the data source, and the challenges related to the documents-to-graph transformation. Then, we present how we employed Large Language Models to derive a machine-readable and accurate version of all the bills and how we transformed such elements into graph objects. Finally, we discuss the technique we adopted to derive references among acts. An overview of the pipeline we implemented is presented in Figure 1.

3.1 Property Graphs and Legislative Knowledge

The property graph data model is a versatile and widely utilized framework for structuring and managing knowledge graphs. It extends the traditional graph representation by embedding attributes into nodes and edges, enabling the representation of intricate relationships and detailed properties within a domain [19]. In this model, nodes represent entities or concepts and are often categorized with labels that classify them into distinct types. Edges capture the relationships between nodes and are similarly typed to reflect their specific nature. Both nodes and edges are enriched with properties, which take the form of key-value pairs that store additional contextual information.

The property graph model is particularly suited for knowledge graphs because it handles heterogeneous data effectively and supports complex queries. Unlike relational databases, where relationships are inferred through joins, the property graph treats relationships as first-class entities, explicitly representing connections within the data. This explicit structure facilitates efficient traversals and querying in interconnected datasets, making it ideal for applications requiring semantic depth and contextual richness with a high degree of flexibility [18]. The PG-Schema [6] that we adopted follows the line of [15], where a unifying graph schema for legislation was proposed, with additions that encode US-specific features as node properties (i.e., the presence of a short title and subjects for each law). Thus, the graph schema becomes:

```
CREATE GRAPH TYPE lawsGraphType STRICT{
  (lawType: Law {id STRING, title STRING,
   chamber STRING, congress INT, publicationDate
   DATE, lawnumber INT, presentedDate DATE,
   subjects LIST, shortTitle STRING, lawText
   STRING}),
```

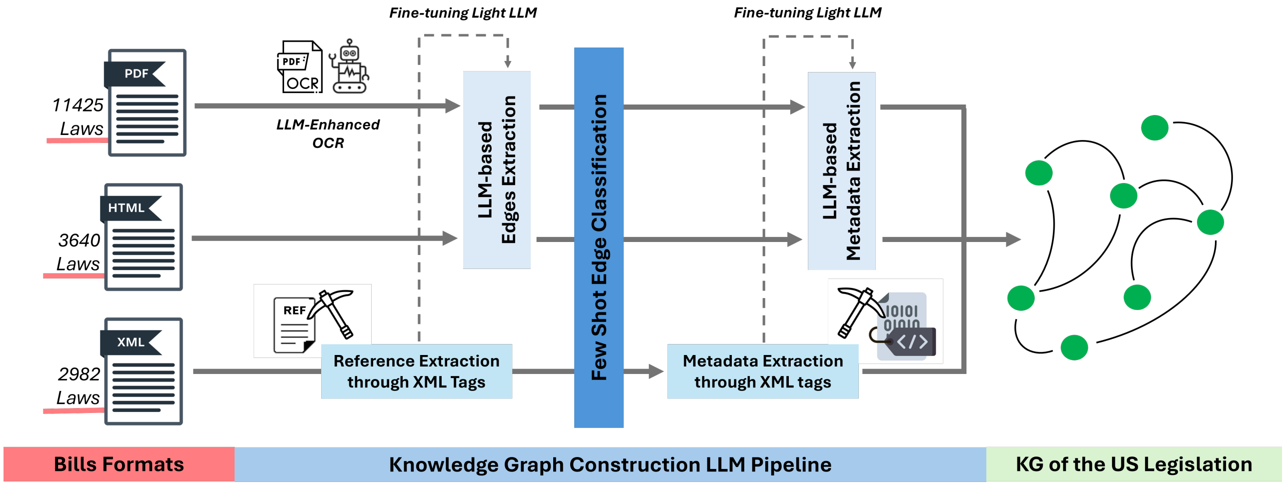
Fine-tuning Light LLM
Fine-tuning Light LLM
PDF
11425 Laws
HTML
3640 Laws
XML
2982 Laws
PDF
OCR
LLM-Enhanced OCR
REF
LLM-based Edges Extraction
Reference Extraction through XML Tags
Few Shot Edge Classification
LLM-based Metadata Extraction
Metadata Extraction through XML tags
Bills Formats
Knowledge Graph Construction LLM Pipeline
KG of the US Legislation

Figure 1: Pipeline for constructing a Knowledge Graph (KG) of U.S. legislation using Large Language Models (LLMs). The process involves extracting and classifying edges from various legislative document formats (PDF, HTML, XML) and combining LLM fine-tuning with few-shot learning techniques for edge and node properties retrieval and classification.

```
(:lawType)-[referenceType: is_legal_basis_of]
 ->(:lawType),
(:lawType)-[referenceType:amends|abrogates|cites]
->(:lawType)}
```

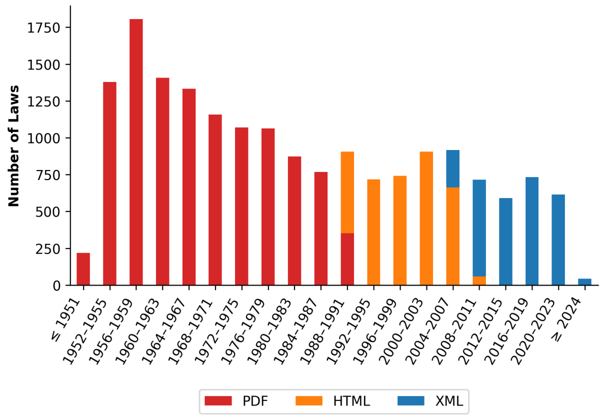

Figure 2: Distribution of the number of laws each year in the different available document formats.

3.2 Data Source

The Congress.gov Application Programming Interface (API) [28], developed by the U.S. Library of Congress, offers a structured way for Congress and the public to access and retrieve machine-readable data, i.e., the federal congressional bills from the collections hosted by the Library of Congress. The available bills on the portal cover the years from 1951 to 2024. While most recent bills are available in an XML or HTML format, older bills are only available as digitalized PDFs, i.e., images that have been scanned and added to the collection of bills. If multiple formats of the same law were available, we preferred the XML version over HTML because its structured tagging is more consistent, facilitating the conversion to graph objects [41]. In Figure 2, we report the number of laws available each year in different formats. There are 11,425 PDF laws from 1951 to 1992, 3,640 HTML laws from 1989 to 2014, and 2,982 XML laws from 2006 to 2024. Thus, we implement additional steps to our pipeline to expand the graph to all U.S. legislation available in digital version, allowing us to derive, to the best of our knowledge, the widest structured database of congressional legislation. In Figure 1, we depict the number of bills available for each publication format.

Congress Bills and Legislation. Congressional bills represent formal legislative proposals introduced within the United States Congress, which comprises two chambers: the House of Representatives and the Senate. A legislative proposal may take the form of either a bill or a joint resolution, and it can originate in either chamber. It is worth noting that multiple versions of a single bill often emerge during the legislative process. These variations may result from successful amendments made, for instance, when a bill transitions from one chamber to the other after passage in the originating chamber. Nevertheless, while the API gives access to all of these intermediate versions, we are interested in retrieving the final version of the law, as introduced within the legislation. We can distinguish between public and private bills. While public bills address general public matters and pertain to individuals only as members of broader categories, private bills aim to confer benefits limited to one or more specific individuals, including corporations or institutions, usually in cases where no alternative legal remedy exists. Resolutions, though also legislative acts, differ from bills in their scope and purpose. Simple resolutions pertain to the operations or collective opinions of a single chamber, while concurrent resolutions address issues affecting both chambers or express their shared stance on public policy [27].

In the proposed Knowledge Graph, we represent only public bills (or either law) as nodes in the graph, as they are the legislative basic units that represent the federal legislative system and are relevant for deriving the graph of in-force federal legislation. In the following paragraphs and sections, we will refer to bills or laws (interchangeably) to indicate Congress-introduced public bills.

The United States Code. The United States Code represents the codified organization of the general and permanent laws of the United States, arranged systematically by subject matter. It comprises 53 titles, each addressing a broad area of law, and is published by the Office of the Law Revision Counsel of the U.S. House of Representatives. Initially released in 1926, the U.S. Code's second main edition followed in 1934, with subsequent primary editions issued at six-year intervals since that time [45].

The unstable nature of the U.S. Code, with new sections and topics being introduced at any publication, poses a significant challenge in its full integration into a structured Knowledge Graph, which should capture the temporal evolution of the legislative system and aims at modeling the interdependencies among acts. However, since its contents directly derive from the bills introduced by Congress and can be traced by selecting the corresponding law, we decided to exclude it from a direct representation in the graph.

Document Metadata as Node Properties. For each introduced bill, the Library of Congress API provides useful bill-specific metadata, such as the bill type, the bill number, the Congress that enacted the bill, the originating chamber, and the date of introduction and/or publication. Such information is available for all acts and can be directly assigned as properties of each law node. However, only some of these attributes are available only for a portion of laws, thus harming the possibility of conducting a comprehensive analysis and navigation of the graph by leveraging such information.

Quality of Digitalized Bills. While the adoption of international standards and machine-readable formats like XML-based ones naturally improves the quality and accessibility of legislative documents, such a conversion exercise has not been extended to old U.S. legislation, which is vastly available as figures within PDFs (see Figure 1). Finally, as we target to obtain a structured representation of bills and their references, both PDF and HTML-based bills do not tag references that appear within their text, raising the issue of (uniquely) identifying the reference within an unstructured text.

3.3 LLM-Enanched OCR

The advent of powerful Large Language Models (LLMs) has stimulated their use also as correction agents for Optical Character Recognition (OCR) tasks and, in particular, to fix OCR-induced errors derived from bad quality or other noise and image misalignment that results in wrong extraction of texts. For instance, in [30], they use LLMs to enrich the outputs of OCR applied to documents containing financial invoices, and they demonstrate that, by fusing OCR engines, such as Tesseract, with the two LLM models, Llama3 and Mistral, they improve the accuracy and reliability of information extraction operations. In [44], they also started using LLMs to enhance the OCR quality of scanned historical newspapers by using a prompt-based approach that instructs the model to detect and correct OCR errors and achieving a significant reduction in character-reading error rate. For this task, we prefer a two-step approach, instead of the more recently introduced vision models since (i) laws are mostly text and starting from a more controlled OCR-based output provides a more reliable output and (ii) the LLM fixing task allows us to use more flexibility in the choice of the model, with a reduction of costs and computational power required. Similarly, in [26, 29], they demonstrate the power of LLMs in extracting structured information from text, evaluating the ability of models like GPT-3 to accomplish the extraction task.

Following these lines of research, we combined a widely popular OCR engine, i.e., Tesseract [43], with an adequately instructed LLM to achieve a consistent and accurate extraction of PDF-scanned bills. In detail, we employed an open-source LLM, the LLama-3 70B model, to act as a correction agent for OCR digitalized documents and carefully instruct it to handle and read legislative bills. To this aim, we adopted the following steps:

(1) Each OCR digitalized document is split into multiple chunks if the length is higher than the length of the output of the LLama-3 model, which is restricted to 2048 max token output.

(2) The LLM is provided with the following system instructions, which combines popular prompt instructions used in LLM text correction [46], designed for maintaining both the content and keeping all information, minimizing the potential hallucinations:

```
Correct OCR-induced errors in the text, ensuring it
flows coherently with the previous context. Follow
these guidelines:
1. Fix OCR-induced typos and errors:
- Correct words split across line breaks
- Fix common OCR errors (e.g., 'rn' misread as 'm')
- Use context and common sense to correct errors
- Only fix clear errors, don't alter the content
  unnecessarily
- Do not add extra periods or any unnecessary
  punctuation
2. Maintain original structure:
- Keep all headings and subheadings intact
3. Preserve original content:
- Keep all important information from the original
  text
- Do not add any new information not present in
  the original text
- Remove unnecessary line breaks within sentences
  or paragraphs
- Maintain paragraph breaks
4. Maintain coherence:
- Ensure the content connects smoothly with the
  previous context
- Handle text that starts or ends mid-sentence
  appropriately
IMPORTANT: Respond ONLY with the corrected text.
Preserve all original formatting, including line
breaks. Do not include any introduction, explanation,
or metadata.
```

(3) Each chunk is parsed within the following template, creating a set of LLM requests:

```
Previous context:{previousChunk.lastParagraph}
Current chunk to process:{currentChunk}
Corrected text:
```

(4) Each LLM request is performed by providing the system content and a small set of manually crafted examples that instruct the LLM on the OCR correction task, thus performing the so-called few-shot learning [38]

(5) Chunks are joined to output a single document

Each bill is thus modeled as a node in the graph, with the high-quality text resulting from the LLM being a property of each node. As a unique identifier for each node, we use the public law alphanumeric format, replicating what is used by the Library of Congress:

<chamber-acronym><number of congress>-<law number> (1)

3.4 Edge Derivation and Classification

Each act of the Congress is cited through a *short title*, which is indicated in a specific section of the act itself. For instance, at the beginning of its content, law `HR86-705` states that: *This act may be cited as the "Mineral Leasing Act Revision of 1960".* Thus, all other acts that refer to law `HR86-705` must use such a short title.

The XML structure adopted by the Library of Congress for publishing the introduced bills includes the use of specific tags to indicate the presence of references throughout the text, as, for instance:

... Section 21 of the Small Business Act <external-xref legal-doc="usc" parsable-cite="usc/15/648">15 U.S.C. 648</external-xref> is amended by adding at the end the following: ...

Therefore, for bills whose XML version is available, we can extract such citations by leveraging the *xref* tags. However, when parsing such elements, two scenarios might occur:

(1) The tagged XML element explicitly indicates the referenced Public Law through its identifier, as in (1), allowing us to directly retrieve the destination node and, thus, the edge.

(2) The tagged XML element indicates the part of the U.S. Code that is being referenced. As described in Section 3.2, rules deriving from introduced acts are arranged within the U.S. Code according to the subject or topic. However, the unstable nature of the U.S. Code, with sections being updated, moved, or added, hinders the possibility of utilizing the U.S. Code as a bridge towards the underlying real introduced bill being cited. Nevertheless, while the tag contains the Section of the U.S. Code being referenced, we noticed that the textual citation always refers to the introduced act (see previous example). Thus, in such a scenario, we use the *xref* tag as a placeholder, which indicates the position where there is a textual citation, and we then parse the text to derive the *short title* of the referenced act (in the previous example, "Small Business Act"). To derive the reference, we implement heuristics that leverage the linguistic feature of how legislative acts are cited in U.S. acts (e.g., in most cases, the citation is preceded by a *the* article). We use the node metadata (i.e., the short title) to retrieve the public law identifier through an exact string matching.

Edge Inference. While for the XML-published bills, we can leverage tags and heuristics to detect the references to other acts, the approach does not apply to HTML and, especially, PDF-derived bills. In both cases, no tag is available in the text, which can be used to retrieve the cited act. A potential tool to solve this issue is Eyecite [17], an open-source tool to extract references (through Hyperscan [47]) from raw text. It is based on a regular expressions database built from multiple sources and specifically designed for the United States' laws and citations. However, when we tried to run it, the results were unsatisfactory since it only detected the compact U.S. Code reference without the short titles of acts. For instance, in the previous example, Eyecite would extract the reference *"15 U.S.C. 648"* instead of the short title one, which is the only way to infer an edge between two public laws since no other identifier is available and used when citing another legal document. In fact, the U.S. code references are inconsistent over time since the U.S. code evolves and can't be used to map bills directly.

LLM Edge Extractor. In its *instruct* chat-based version, Mistral 7B is a large language model that balances accuracy and computational efficiency in performing specific tasks [23]. It is significantly smaller than larger models like GPT-4 or Llama3-70B, outperforming comparable large language models. It is released under the Apache 2.0 license, allowing users to fine-tune for specific tasks. In our scenario, we fine-tuned the model to extract references to other acts based on the short title[1]. To this aim, we leveraged the XML-derived citations to create a training dataset of true pairs of texts and citations. To facilitate the task, we split the text of laws into paragraphs by wrapping the text (this is also done for the inference task). Through this approach, we created a dataset of 18k pairs[2], split into 80-20 training and validation sets for fine-tuning. The model has been trained for 3200 steps with a batch size of 4, 4-bit quantization using bits and bytes, and a LoRA rank of 64. We use the paged Adam optimizer, a learning rate of 5e-05, and a cosine learning rate scheduler with a 0.03 warm-up fraction. We used an A100 GPU with 40 GB of memory, and the best model reported a validation loss of 1.001. In our Appendix, we illustrate the details of the parameters used for fine-tuning the model (Table A.1) and, in Figure A.1, we show the training and validation loss of the training process. The model is publicly available at [2].

For performing inference, we combined the fine-tuned model with few-shot learning, which enhances the accuracy of the model by instructing it with a small set of manually crafted examples that are provided as contextual input to the model [38, 48].

Results. The number of detected citations is depicted in Table 2, where we compare our approach with benchmarks, namely a deterministic approach (which can only be used on XML files) and Eyecite, as described in the previous paragraphs. The LLM-based approach outperforms both benchmarks and can find many more citations that can be used to generate an interconnected legislative graph. For PDFs, the approaches are comparable, which is most likely due to citations to much older acts that the LLM has rarely seen during training. Focusing only on edges that can be represented within the graph (due to the presence of both the source and destination node), we get 31.180 unique citations. This figure does not include multiple citations with the same source and destination.

[1] During our tests, we experienced better results by fine-tuning a Mistral model rather than a Llama3-7B. The analysis of such differences is out-of-scope of this work.

[2] The dataset is available at [1]

Paragraph of the law	Reference	Type
(c) Reporting Amendment.–The Sudan Peace Act (50 U.S.C. 1701 note) is amended by striking section 8 and inserting the following:…	The Sudan Peace Act	`AMENDS`
2. Since the enactment of the Trafficking Victims Protection Act of 2000 (division A of Public Law 106-386), the United States Government has made significant progress in investigating and prosecuting acts of trafficking and in responding to the needs of victims of trafficking in the United States and abroad …	Trafficking Victims Protection Act of 2000	`CITES`

Table 1: Few-shot learning example provided to the LLama-3 70B model for performing edge labeling, i.e., classifying the reference according to its type and coherently with the graph schema

Approach	PDFs	HTMLs	XMLs
Deterministic (XML Tags)	-	-	18.801
Eyecite [17]	26.266	23.112	10.551
LLM Edge Extractor	27.556	36.567	25.852

Table 2: Citations to other acts extracted by different approaches, based on the publication format available. The deterministic approach refers to using XML tags, combined with heuristics, to detect references. Also, note that Eyecite does not detect short titles but only U.S. Code references, which do not allow to derive the edge, as discussed in Section 3.2. Note that some of the citations found by our approach do not become edges in our KG since they refer to laws published before 1951.

Edge Labelling. Citations can have different meanings, and users might be interested in having a richer set of citation labels to use when conducting their analysis. This issue has been studied in [39], where authors built a manually annotated dataset that contains labeled U.S. law references according to their types, very close to the ones of our graph schema. However, since the dataset is not publicly available, we can't leverage such a resource in our context. To tackle this, we adopted a few-shot learning approach of a large LLM model, instructing it to perform text classification. We adopt the following instructions as system content:

```
You are an assistant that, based on a reference, you have
to classify it into the following categories:
-'AMENDS', if the text is modifying something about the
    reference law
-'ABROGATES', if the text is scraping something about the
    reference law
-'IS_LEGAL_BASIS_OF', if the reference law is used as the
    foundation for stating something
-'CITES' in the rest of the cases, such as when it is a
    generic citation
```

and we provide the LLM with manually annotated examples that we present in Table 1. The approach resulted in the creation of 2.979 `AMENDS` edges, 201 `ABROGATES` edges, 1.843 `IS_LEGAL_BASIS_OF` edges and 28.000 `CITES` ones. We manually evaluated samples of such reclassification and noted that, as desired, the approach was quite conservative and accurate, meaning it re-classifies edges in safe cases, leaving doubtful cases with the generic `CITES` label. In future iterations, such accurate classifications can be used to further reclassify additional edges by fine-tuning dedicated LLMs that are suited for the classification task, such as adopting a BERT model that demonstrates state-of-the-art performance in various domains. [12, 49]

3.5 Nodes Properties Enrichment

As discussed in Section 3.2, recent laws published by the Library of Congress are provided with useful document-level metadata assigned as node properties of the KG, while older laws do not share the same availability of metadata. In this work, we focus on the *subjects* attribute, describing the topics regulated by an act, which, for recent laws, are inserted by experts and are available for 7857 acts out of the 17k acts (i.e., nodes) that compose our database. This attribute is crucial for performing structured queries that enable fast access to documents of interest or to build and enhance retrieval systems that can easily understand the content of an act through the topics.

To support such studies, we developed an approach to integrate missing information that typically characterizes older laws. In particular, given the availability of acts whose subjects were manually annotated, we obtain a full representation of the subjects of all laws of our KG by fine-tuning a light LLM model capable of extracting subjects from texts. We used the same configuration described in Table A.1 to fine-tune a second Mistral7B model, a *Subject Extractor*, utilizing the same machine. The evolution of the train and validation losses are available in the Appendix (Figure A.2). Given the easier task, w.r.t. edge extraction, we achieve a better validation loss of around 0.83 with the need for fewer steps before convergence. The fine-tuned model is also publicly available at [3].

In Figure 3, we report the most frequent subjects, combining both metadata manually inserted by the Library of Congress and extending such knowledge with our approach to older laws. Such subjects naturally relate to general legal aspects such as politics, government, congress, and the economy. Although less frequent, most extracted subjects reflect the specific details of the laws examined. Out of the 7717 different subjects extracted, only 5771 are common to at most 10 laws, and 2817 are found in just a single law. For example, some of the single-law subjects are *"Cigars"* or *"Printing paper"*.

4 APPLICATIONS

In this section, we provide an overview of some of the applications we unlock by constructing a structured graph of the US legislation and, in particular, by adopting the Property Graph schema [13].

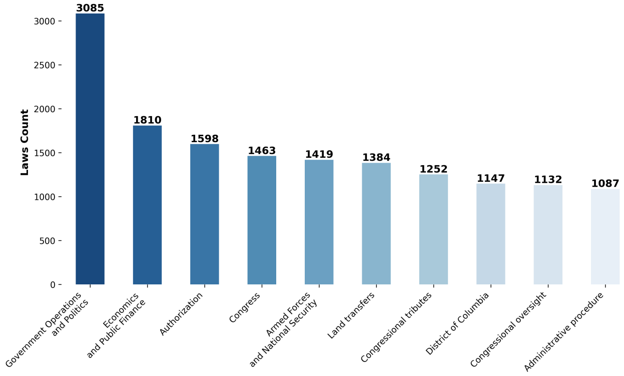

Figure 3: Top 10 most frequent subjects extracted from the text of laws. For each subject, the number of laws published with that specific subject is reported.

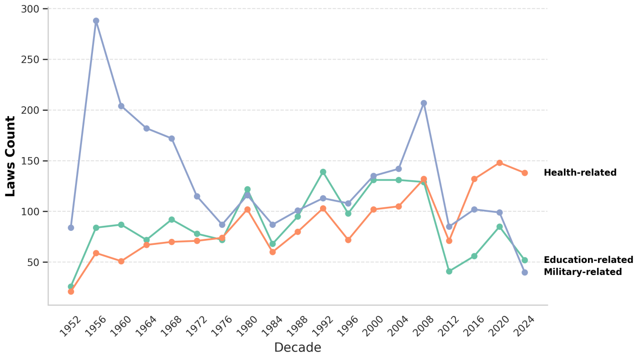

Figure 4: Temporal evolution of laws grouped into macro topics based on the subjects that are assigned to each graph node. For each macro topic, the number of laws published in a 4 years period with that specific subject is reported.

Subject	Law Count	Amended Percentage
Administrative procedure	226	20.8
Authorization	186	11.6
Congressional oversight	173	15.3
Advisory bodies	170	24.8
Agriculture and Food	167	16.7
Government Operations and Politics	164	5.3
Economics and Public Finance	149	8.2
Congress	145	9.9
Appropriations	140	13.3

Table 3: Top 10 most frequent subjects extracted from the text of amended laws. The Law Count column shows the number of amended laws from which the subject is extracted. The Amended Percentage column displays the percentage of amended laws relative to the total number of laws that include that subject.

Naturally, the advantage of having a structured database of legislative knowledge empowers users to write queries that allow the derivation of advanced insights in a straightforward way within the U.S. legislative system, such as retrieving portion of legislation but also, for instance, measuring the government activity on a certain subjects by grouping, filtering and counting nodes. Other tools developed for Property Graphs, as Association Rules and Triggers, could also be used to conduct better knowledge discovery and update the state of the graph [31].

A natural application of the graph model refers to the possibility of users to graphically visualize patterns or structures of U.S. legislation, allowing them to interactively navigate the legislative corpus, by expanding nodes and traversing edges, also visually [42]. Then, since our pipeline integrates missing properties in older laws, such as subjects, we unlock analysts, especially in economics, to conduct temporal analysis on the portion of the graph of interest, which can be easily derived through graph queries. For instance, by isolating laws (thus nodes) of a certain period, we characterize the activity of each Congress over the years.

More advanced insights can be obtained by combining semantics and graph models with network analysis tools: by projecting parts of the graph through queries we facilitate the exploration and analysis of the network topology, by applying centrality or clustering techniques that account for the specific semantic of the graph. For instance, one could be interested in analysing only centrality referred to certain subjects or analyse clusters according to the types of edges, an analysis that requires a rich data model like ours to construct the networks,

4.1 Property Graph Queries

As a representative example of the query-based insights our graph data model can unlock, we present two paradigmatic queries that, leveraging the graph data model and the graph query language, empower users of a tool for a better exploration of the legislation.

4.1.1 Lawmaking Patterns. We show in Figure 4 subjects grouped by three custom macro-topics of interest. These groups are *health*, *education*, and *military*. By tracking such groups, the activity of different governments can be analyzed and scrutinized. In Appendix A.3, we demonstrate how to use Cypher, the graph query language for Property Graph, to conduct this kind of analysis.

4.1.2 Evolution of Amendment Activity. An interlinked and easily queryable graph also opens new ways for evaluating the broader implications of lawmaking. For instance, by combining subjects with references, our resource can be used to provide insights into the evolution of U.S. laws in terms of identifying which topics are more likely to be revised. As a representative use case of such analysis, we present in Table 3 the ten most frequent subjects extracted from amended laws, including both the count of amended laws and the percentage of amended laws relative to the total number of laws with each specific subject. The average percentage of amended laws per subject is 26.4%, with a standard deviation of 25.7%, indicating significant variation in the behavior across different subjects.

Law Title	Publication Year	PageRank Score
Omnibus Crime Control and Safe Streets Act	1968	57.32
General Government Matters Appropriation Act	1960	56.51
Elementary and Secondary Education Act	1965	45.57
Wild and Scenic Rivers Act	1968	39.49
Immigration and Nationality Act	1952	21.39
Public Law 86-643	1960	19.53
Designation of Great Hall of the Capitol Visitor Center as Emancipation Hall	2007	14.10
Higher Education Act	1965	13.59
Public Law 100-230	1988	13.38
Small Business Act	1958	11.94

Table 4: Top 10 most central laws within the largest connected component of the graph. The Publication Year column shows the year in which each law was published. The PageRank Score column displays the score given by the PageRank algorithm to each reported law.

4.2 Network Analysis

By constructing the legislative graph to reflect the underlying semantics of United States legislation, we enable the direct application of network analysis techniques to gain deeper insights into its structure. This includes exploring connectivity and community structures within the citation network of laws, as well as understanding how topics extracted from legal texts are related through the connections between laws.

Such analysis has been largely exploited by legislative and social science literature: for example, to highlight the most influential documents and laws within the legal domain under analysis [24, 35]; and also to study the evolution of a legislative system through the joint work of both legal and community analysis [40]. In the following paragraphs, inspired by such literature, we provide some examples of such techniques, aiming to demonstrate the utility of having a complete representation of the U.S. legislative system in a graph, as derived from our LLM-assisted pipeline.

4.2.1 Network Structure and Influential Laws. As a first example, we can analyze the structure of the resulting graph and identify the most central nodes, by considering all types of citations, thus using the entire graph. The full graph contains 17,961 laws, with a dominant connected component, i.e. the largest subgraph in which a path exists between every pair of nodes, comprising 6,496 laws. The graph contains only four other components with more than five nodes, none exceeding ten, while the remainder are singletons or pairs. Focusing on the dominant component, i.e., on the portion of legislation that plays a more important role, the PageRank algorithm can be applied to identify influential laws within the largest component. In Table 4, the top-ranked nodes are reported: they cover a diverse range of topics -some procedural or bureaucratic (*General Government Matters Appropriation Act, 1961*), others more operational (*Omnibus Crime Control and Safe Streets Act of 1968*, *Elementary and Secondary Education Act of 1965*)— but all are notably older statutes, consistent with the expectation that foundational laws tend to accumulate more references over time.

4.2.2 Communities and Subjects Classification. The graph structure also allows us to explore the internal structure of the law network through the application of community detection algorithms. Understanding how laws cluster together can lead to a better classification of their subjects. For instance, as communities form around related laws, a single subject may appear in multiple communities, revealing overlapping semantic clusters. This helps distinguish between broadly shared topics—often associated with general legal themes—and more specific topics that appear in only a few clusters, highlighting the unique focus of those communities. This unique focus can help refine existing legal classifications, especially when different labels describe similar or related topics, and by grouping related subjects into broader thematic categories, we can reduce fragmentation and improve interpretability.

We clustered the United States laws into 1,530 distinct communities using the Label Propagation algorithm. Most of these communities are very small (one or two nodes), but we find a few mid-sized groups and one very large community containing over 1,000 laws. Then, we cross-referenced the detected communities with the subject extracted in Section 3.5. For example, we identified a community of nine law nodes that collectively touch on 46 distinct topics, all loosely related to nature and the environment. These topics range from general semantic themes—such as *Animals* and *Climate Change and Greenhouse Gases*—to more legally defined areas like *Public Lands and Natural Resources*. This demonstrates how network-based community detection can complement traditional classification schemes and reveal meaningful groupings across legal and semantic dimensions.

4.2.3 Temporal Evolution of Subject Centrality. By combining references and subjects, our legislative graph can unlock more interesting graph analysis.

First, the flexibility of the property graph data model allows us to link subjects based on references, thus deriving a network of co-occurrences among subjects that informs about which subjects are somehow consistently dependent on others, unveiling hidden patterns. This can be done by performing **projections** on the graph:

```
MATCH (l:Law)
UNWIND l.subjects as sub
WITH count(l.lawID) as countL, sub
CREATE (s:Subject {name: sub});
```

```
MATCH (l:Law)
UNWIND l.subjects as sub
WITH l, sub
MATCH (s:Subject)
WHERE s.name = sub
MERGE (l)-[:RELATED_TO]->(s);
```

```
MATCH (s1:Subject)<-[:RELATED_TO]-(l1:Law)-[:CITES]->
      (l2:Law)-[:RELATED_TO]->(s2:Subject)
MERGE (s1)-[r:CITES_SUB]-(s2)
ON CREATE SET r.lawIDFrom = l1.lawID, r.lawIDTo = l2.lawID
```

where we first convert subjects to nodes, and we merge edges to connect laws with their respective subjects. Then, by traversing laws citations, we can create co-occurrence relationships among subjects, thus creating a powerful network of subject interactions to be investigated.

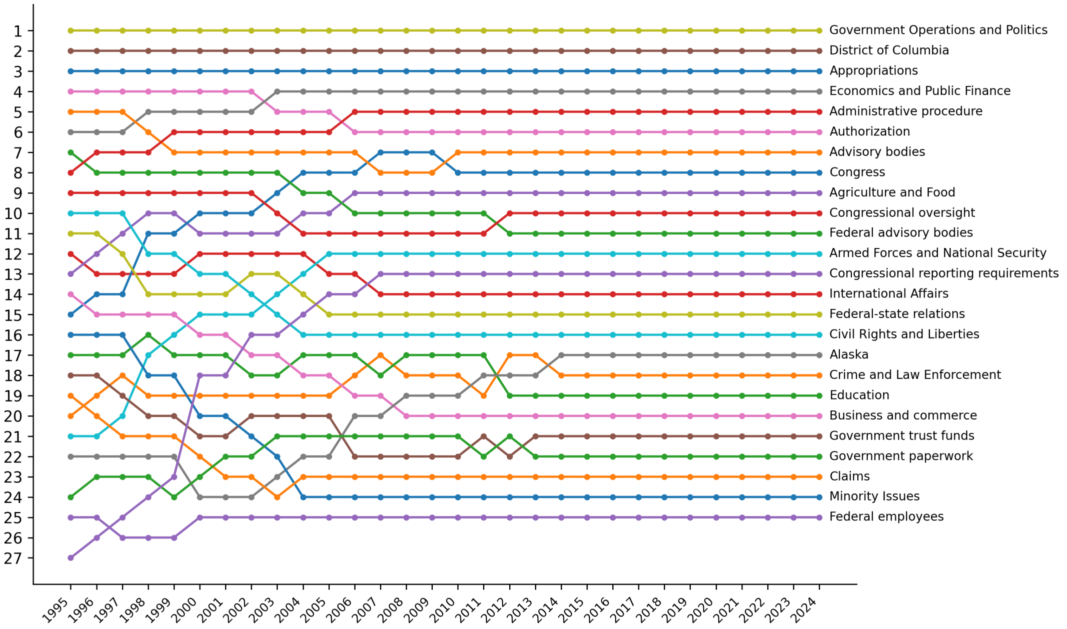

Figure 5: Temporal evolution of the centrality of the 25 most central subjects according to PageRank across the last 30 years.

In the case of the U.S. legislative graph, this step created over 5,000 subject nodes and added more than 1.5 million new edges capturing the semantic and functional relationships within the legislative systems. On this expanded network, temporal centrality metrics can be used to evaluate the evolving importance of each subject throughout the history of the U.S. legislative system, for instance, by applying the PageRank algorithm to measure the centrality of each subject over time, taking into account only the nodes and edges that had been introduced up to the year under analysis.

Figure 5 reports the top 25 most central subjects over the past 30 years. Notably, while the top three subjects have consistently held their positions across decades, from 2015 onward, the entire top-25 ranking has remained unchanged—a marked contrast to earlier years, where significant fluctuation was observed beyond the top three. This plateau in subject centrality suggests a stabilization or ossification of thematic focus in recent legislative activity, marking a potentially significant shift in the structural dynamics of U.S. lawmaking. While many of the most frequently occurring subjects extracted directly from the text of the laws also appear among the top-ranked in terms of centrality, the two rankings do not fully align. This divergence highlights how incorporating graph structure and LLM-based edge construction enables a more refined analysis of the U.S. legislative system.

5 CONCLUSIONS AND FUTURE WORK

In this paper, we present an LLM-assisted pipeline that constructs the widest Knowledge Graph representing the U.S. Congressional legislation since the 50s, for which digitalized documents from the Library of Congress are available. Compared to previous works, we adopt a recently published unifying database schema based on the property graph model, integrating and enriching it with powerful LLMs capable of transforming unstructured low-quality textual data into structured information. By modeling the data in a graph, we allow interested stakeholders to easily query information about the U.S. legislation, supporting traditional users working with legislative data and pushing network-based studies on such knowledge or developing graph-based search systems that synergize with KGs to achieve better searches. We also provided some relevant application examples inspired by computer law and social science literature, which would primarily benefit from the constructed graph.

In future iterations, we aim to expand the pipeline by integrating other legislation sources, such as Congress resolutions or Presidential acts that are a relevant part of the U.S. legislation.

Resources. The Property Graph of the U.S. legislation is available on Zenodo [14]. The fine-tuned LLMs, the Mistral-7B for edge finding and the Mistral-7B for subject extraction, are available on HuggingFace [2, 3].

ACKNOWLEDGMENTS

Andrea Colombo kindly acknowledges INPS for the funding of his Ph.D. program.

A APPENDIX

A.1 LLM Fine-tuning Parameters

Parameter	Value
Batch Size	4
Learning Rate	5e-05
Training Time	6 h
Best Validation Loss	1.01
Optimization	Adam
Training set size	18k
Warm-up fraction	0.03
Fine-tuning Technique	LoRa

Table A.1: Fine-tuning parameters and settings to fine-tune the Mistral model, based on the LoRa technique [21].

A.2 Ref-Finder Fine-Tuning

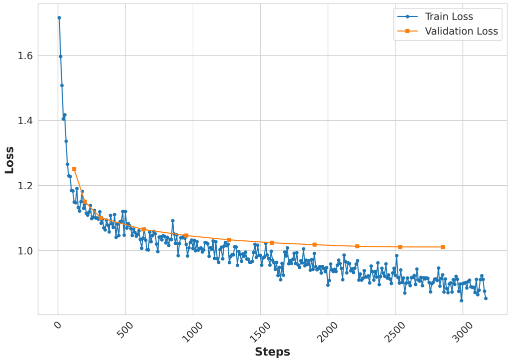

Figure A.1: Training and validation loss curves for the fine-tuning of the Ref-Finder. The blue line represents the training loss at each step, while the orange line represents the validation loss, which converges after around 2000 steps.

A.3 Subject Extractor Fine-Tuning

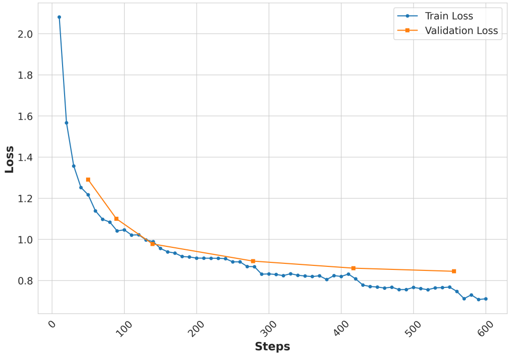

Figure A.2: Training and validation loss curves for the fine-tuning of the Subject Extractor.

A.4 Querying the graph with Cypher

The Cypher query to create macro-topics statistics as in Figure 4 can be formulated as follows:

```
MATCH (l:Law)
WHERE (ANY(x in l.subjects WHERE lower(x) contains
    'education') OR ANY(x in l.subjects WHERE lower(x)
     contains 'school'))
RETURN l.publicationDate.year as year, 'Education-related'
        AS subject

UNION ALL

MATCH (l:Law)
WHERE (ANY(x in l.subjects WHERE lower(x) contains
    'medical') OR ANY(x in l.subjects WHERE lower(x)
     contains 'health'))
RETURN l.publicationDate.year as year, 'Health-related'
    AS subject

UNION ALL

MATCH (l:Law)
WHERE (ANY(x in l.subjects WHERE lower(x) contains
    'military') OR ANY(x in l.subjects WHERE lower(x)
     contains 'armed forces'))
RETURN l.publicationDate.year as year, 'Military-related'
    AS subject
```

REFERENCES

[1] Andrea Colombo. 2025. Dataset US-Public-Laws-Citations. https://doi.org/10.57967/hf/4252

[2] Andrea Colombo. 2025. US-Laws-Reference-Extractor. https://doi.org/10.57967/hf/4254

[3] Andrea Colombo. 2025. US-Laws-Subjects-Extractor. https://doi.org/10.57967/hf/4253

[4] Vito Walter Anelli, Eros Brienza, Marco Recupero, Francesco Greco, Andrea De Maria, Tommaso Di Noia, and Eugenio Di Sciascio. 2023. Navigating the Legal Landscape: Developing Italy's Official Legal Knowledge Graph for Enhanced Legislative and Public Services. In *Proceedings of the Italia Intelligenza Artificiale - Thematic Workshops co-located with the 3rd CINI National Lab AIIS Conference on Artificial Intelligence (Ital IA 2023), Pisa, Italy, May 29-30, 2023 (CEUR Workshop Proceedings)*, Fabrizio Falchi, Fosca Giannotti, Anna Monreale, Chiara Boldrini, Salvatore Rinzivillo, and Sara Colantonio (Eds.), Vol. 3486. CEUR-WS.org, 223–228. https://ceur-ws.org/Vol-3486/87.pdf

[5] I. Angelidis, I. Chalkidis, C. Nikolaou, P. Soursos, and M. Koubarakis. 2018. Nomothesia: A linked data platform for Greek legislation. In *Proceedings of MIREL 2018 Workshop on MIning and REasoning with Legal texts, Luxembourg, 30-08-2018.* https://cgi.di.uoa.gr/~koubarak/publications/2018/nomothesia-linked-data.pdf

[6] Renzo Angles, Angela Bonifati, Stefania Dumbrava, George Fletcher, Alastair Green, Jan Hidders, Bei Li, Leonid Libkin, Victor Marsault, Wim Martens, Filip Murlak, Stefan Plantikow, Ognjen Savkovic, Michael Schmidt, Juan Sequeda, Slawek Staworko, Dominik Tomaszuk, Hannes Voigt, Domagoj Vrgoc, Mingxi Wu, and Dusan Zivkovic. 2023. PG-Schema: Schemas for Property Graphs. *Proc. ACM Manag. Data* 1, 2, Article 198 (June 2023), 25 pages. https://doi.org/10.1145/3589778

[7] Nadia Banteka. 2019. A Network Theory Approach to Global Legislative Action. *Seton Hall L. Rev.* 50 (2019), 339.

[8] Paheli Bhattacharya, Kripabandhu Ghosh, Arindam Pal, and Saptarshi Ghosh. 2022. Legal case document similarity: You need both network and text. *Information Processing and Management* 59, 6 (Nov. 2022), 103069. https://doi.org/10.1016/j.ipm.2022.103069

[9] Michael James Bommarito and Daniel Martin Katz. 2009. Properties of the United States Code Citation Network. *SSRN Electronic Journal* (2009). https://doi.org/10.2139/ssrn.1502927

[10] Romain Boulet, Pierre Mazzega, and Danièle Bourcier. 2010. *Network Analysis of the French Environmental Code.* Springer Berlin Heidelberg, 39–53. https://doi.org/10.1007/978-3-642-16524-5_4

[11] Romain Boulet, Pierre Mazzega, and Danièle Bourcier. 2011. A network approach to the French system of legal codes—part I: analysis of a dense network. *Artificial Intelligence and Law* 19, 4 (Nov. 2011), 333–355. https://doi.org/10.1007/s10506-011-9116-1

[12] Qingyu Chen, Jingcheng Du, Alexis Allot, and Zhiyong Lu. 2022. LitMC-BERT: Transformer-Based Multi-Label Classification of Biomedical Literature With An Application on COVID-19 Literature Curation. *IEEE/ACM Trans. Comput. Biol. Bioinformatics* 19, 5 (May 2022), 2584–2595. https://doi.org/10.1109/TCBB.2022.3173562

[13] Andrea Colombo. 2024. Leveraging Knowledge Graphs and LLMs to Support and Monitor Legislative Systems. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management* (Boise, ID, USA) *(CIKM '24)*. Association for Computing Machinery, New York, NY, USA, 5443–5446. https://doi.org/10.1145/3627673.3680268

[14] Andrea Colombo. 2025. [dataset] Knowledge Graph of the US Legislation. https://doi.org/10.5281/zenodo.14858340

[15] Andrea Colombo, Anna Bernasconi, and Stefano Ceri. 2025. An LLM-assisted ETL pipeline to build a high-quality knowledge graph of the Italian legislation. *Information Processing and Management* 62, 4 (2025), 104082. https://doi.org/10.1016/j.ipm.2025.104082

[16] Andrea Colombo, Francesco Cambria, and Francesco Invernici. 2025. Legislative Knowledge Management with Property Graphs [TO APPEAR]. In *Transforming Graph Data Workshop, First Edition - co-located with EDBT 2025 - March 2025, Barcelona, Spain.*

[17] Jack Cushman, Matthew Dahl, and Michael Lissner. 2021. eyecite: A Tool for Parsing Legal Citations. *Journal of Open Source Software* 6, 66 (2021), 3617. https://doi.org/10.21105/joss.03617

[18] Souripriya Das, Jagannathan Srinivasan, Matthew Perry, Eugene Inseok Chong, and Jayanta Banerjee. 2014. A Tale of Two Graphs: Property Graphs as RDF in Oracle. In *Proceedings of the 17th International Conference on Extending Database Technology, Athens, Greece, March 24–28*. https://doi.org/10.5441/002/edbt.2014.82

[19] Alin Deutsch, Nadime Francis, Alastair Green, Keith Hare, Bei Li, Leonid Libkin, Tobias Lindaaker, Victor Marsault, Wim Martens, Jan Michels, Filip Murlak, Stefan Plantikow, Petra Selmer, Oskar van Rest, Hannes Voigt, Domagoj Vrgoč, Mingxi Wu, and Fred Zemke. 2022. Graph Pattern Matching in GQL and SQL/PGQ. In *Proceedings of the 2022 International Conference on Management of Data* (Philadelphia, PA, USA) *(SIGMOD '22)*. Association for Computing Machinery, New York, NY, USA, 2246–2258. https://doi.org/10.1145/3514221.3526057

[20] Rinke Hoekstra, Joost Breuker, Marcello Di Bello, and Alexander Boer. 2007. The LKIF core ontology of basic legal concepts. *CEUR Workshop Proceedings* 321 (2007), 43–63. https://ceur-ws.org/Vol-321/paper3.pdf 2nd Workshop on Legal Ontologies and Artificial Intelligence Techniques, LOAIT 2007 ; Conference date: 04-06-2007 Through 04-06-2007.

[21] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685* (2021).

[22] ISO. 2024. ISO/IEC 39075:2024 - Information technology — Database languages — GQL. https://doi.org/ISO/IEC39075:2024

[23] Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7B. *arXiv preprint arXiv:2310.06825* (2023).

[24] Marios Koniaris, Ioannis Anagnostopoulos, and Yannis Vassiliou. 2018. Network analysis in the legal domain: a complex model for European Union legal sources. *Journal of Complex Networks* 6, 2 (2018), 243–268.

[25] legis graph. 2015. Legis-Graph: A Graph Database of Legislative Data. https://github.com/legis-graph/legis-graph. Accessed: 2025-02-08.

[26] Bo Li, Gexiang Fang, Yang Yang, Quansen Wang, Wei Ye, Wen Zhao, and Shikun Zhang. 2023. Evaluating ChatGPT's Information Extraction Capabilities: An Assessment of Performance, Explainability, Calibration, and Faithfulness. *CoRR* abs/2304.11633 (2023). https://doi.org/10.48550/ARXIV.2304.11633 arXiv:2304.11633

[27] Library of Congress. 2025. Bills and Resolutions. https://www.loc.gov/collections/century-of-lawmaking/articles-and-essays/statutes-and-documents/bills-and-resolutions/.

[28] Library of Congress API. 2025. API.Congress.Gov. https://github.com/LibraryOfCongress/api.congress.gov. GitHub repository.

[29] Yuliang Liu, Zhang Li, Mingxin Huang, Biao Yang, Wenwen Yu, Chunyuan Li, Xu-Cheng Yin, Cheng-Lin Liu, Lianwen Jin, and Xiang Bai. 2024. OCRBench: on the hidden mystery of OCR in large multimodal models. *Science China Information Sciences* 67, 12 (Dec. 2024). https://doi.org/10.1007/s11432-024-4235-6

[30] Faiza Loukil, Sarah Cadereau, Hervé Verjus, Mattéo Galfré, Kavé Salamatian, David Telisson, Quentin Kembellec, and Olivier Le Van. 2024. LLM-centric pipeline for information extraction from invoices. In *International Conference on Foundation and Large Language Models (FLLM2024)*.

[31] Davide Magnanimi, Andrea Colombo, Luigi Bellomarini, Anna Bernasconi, Stefano Ceri, and Davide Martinenghi. 2025. Enabling Light-Weight Reasoning via Cypher Triggers . In *2025 IEEE 41st International Conference on Data Engineering (ICDE)*. IEEE Computer Society, Los Alamitos, CA, USA, 4277–4290. https://doi.org/10.1109/ICDE65448.2025.00320

[32] Nada Mimouni. 2013. Modeling Legal Documents as Typed Linked Data for Relational Querying. In *First JURIX Doctoral Consortium and Poster Sessions in conjunction with the 26th International Conference on Legal Knowledge and Information Systems, JURIX 2013*, Vol. 1105.

[33] Nada Mimouni, Meritxell Fernàndez, Adeline Nazarenko, Danièle Bourcier, and Sylvie Salotti. 2013. A relational approach for information retrieval on XML legal sources. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Law* (Rome, Italy) *(ICAIL '13)*. Association for Computing Machinery, New York, NY, USA, 212–216. https://doi.org/10.1145/2514601.2514629

[34] Nada Mimouni, Adeline Nazarenko, and Sylvie Salotti. 2015. *A Conceptual Approach for Relational IR: Application to Legal Collections*. Springer International Publishing, 303–318. https://doi.org/10.1007/978-3-319-19545-2_19

[35] Thom Neale. 2013. Citation analysis of canadian case law. *J. Open Access L.* 1 (2013), 1.

[36] OASIS. 2018. Akoma Ntoso Version 1.0 becomes an OASIS Standard. Available online at: https://www.oasis-open.org/news/announcements/akoma-ntoso-version-1-0-becomes-an-oasis-standard/, last accessed on 01.05.2024.

[37] Monica Palmirani. 2021. Lexdatafication: Italian Legal Knowledge Modelling in Akoma Ntoso. In *AI Approaches to the Complexity of Legal Systems XI-XII*, Víctor Rodríguez-Doncel, Monica Palmirani, Michał Araszkiewicz, Pompeu Casanovas, Ugo Pagallo, and Giovanni Sartor (Eds.). Springer International Publishing, Cham, 31–47. https://doi.org/10.1007/978-3-030-89811-3_3

[38] Archit Parnami and Minwoo Lee. 2022. Learning from few examples: A summary of approaches to few-shot learning. *arXiv preprint arXiv:2203.04291* (2022).

[39] Ali Sadeghian, Laksshman Sundaram, Daisy Zhe Wang, William F. Hamilton, Karl Branting, and Craig Pfeifer. 2018. Automatic semantic edge labeling over legal citation graphs. *Artif. Intell. Law* 26, 2 (June 2018), 127–144. https://doi.org/10.1007/s10506-018-9217-1

[40] Urška Šadl, Lucía López Zurita, and Sebastiano Piccolo. 2024. The European Community of Law and the Communities of Case-law: Understanding legal concepts and processes through the lens of community detection algorithms. *European law open* 3, 2 (2024), 431–455.

[41] Ababneh Sana and G. Suganthi. 2017. Modeling and Storage of XML Data as a Graph and Processing with Graph Processor. *World Congress on Computing and Communication Technologies (WCCCT)* (2017), 16–19. https://doi.org/10.1109/WCCCT.2016.14

[42] Galileo Sartor, Piera Santin, Davide Audrito, Emilio Sulis, and Luigi Di Caro. 2022. *Automated Extraction and Representation of Citation Network: A CJEU Case-Study*. Springer International Publishing, 102–111. https://doi.org/10.1007/978-3-031-22036-4_10

[43] R. Smith. 2007. An Overview of the Tesseract OCR Engine. In *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, Vol. 2. 629–633. https://doi.org/10.1109/ICDAR.2007.4376991

[44] Alan Thomas, Robert Gaizauskas, and Haiping Lu. 2024. Leveraging LLMs for Post-OCR Correction of Historical Newspapers. In *Proceedings of the Third Workshop on Language Technologies for Historical and Ancient Languages (LT4HALA) @ LREC-COLING-2024*, Rachele Sprugnoli and Marco Passarotti (Eds.). ELRA and ICCL, Torino, Italia, 116–121. https://aclanthology.org/2024.lt4hala-1.14/

[45] U.S. Government Publishing Office. 2025. U.S. Code. https://www.govinfo.gov/app/collection/uscode. GovInfo platform.

[46] MEB Veninga. 2024. *LLMs for OCR Post-Correction*. Master's thesis. University of Twente.

[47] Xiang Wang, Yang Hong, Harry Chang, KyoungSoo Park, Geoff Langdale, Jiayu Hu, and Heqing Zhu. 2019. Hyperscan: a fast multi-pattern regex matcher for modern CPUs. In *Proceedings of the 16th USENIX Conference on Networked Systems Design and Implementation* (Boston, MA, USA) *(NSDI'19)*. USENIX Association, USA, 631–648.

[48] Yaqing Wang, Quanming Yao, James T. Kwok, and Lionel M. Ni. 2020. Generalizing from a Few Examples: A Survey on Few-shot Learning. *ACM Comput. Surv.* 53, 3, Article 63 (jun 2020), 34 pages. https://doi.org/10.1145/3386252

[49] Hamada M Zahera, Ibrahim A Elgendy, Rricha Jalota, Mohamed Ahmed Sherif, and E Voorhees. 2019. Fine-tuned BERT Model for Multi-Label Tweets Classification.. In *Proceedings of the Twenty-Eighth Text REtrieval Conference,{ TREC} 2019, Gaithersburg, Maryland, USA, November 13-15, 2019*. 1–7. https://trec.nist.gov/pubs/trec28/papers/DICE_UPB.IS.pdf