A Query System for Extracting Requirements-related Information from Legal Texts

Amin Sleimi*, Marcello Ceci*, Nicolas Sannier*, Mehrdad Sabetzadeh*, Lionel C. Briand*†, John Dann§
*SnT Centre for Security, Reliability and Trust, University of Luxembourg, Luxembourg
†School of Electrical Engineering and Computer Science, University of Ottawa, Canada
§Central Legislative Service (SCL), Ministry of State, Luxembourg
Email: {sleimi, ceci, sannier, sabetzadeh, briand}@svv.lu, john.dann@scl.etat.lu

***Abstract*—Searching legal texts for relevant information is a complex and expensive activity. The search solutions offered by present-day legal portals are targeted primarily at legal professionals. These solutions are not adequate for requirements analysts whose objective is to extract domain knowledge including stakeholders, rights and duties, and business processes that are relevant to legal requirements. Semantic Web technologies now enable smart search capabilities and can be exploited to help requirements analysts in elaborating legal requirements.**

In our previous work, we developed an automated framework for extracting semantic metadata from legal texts. In this paper, we investigate the use of our metadata extraction framework as an enabler for smart legal search with a focus on requirements engineering activities. We report on our industrial experience helping the Government of Luxembourg provide an advanced search facility over Luxembourg's Income Tax Law. The experience shows that semantic legal metadata can be successfully exploited for answering requirements engineering-related legal queries. Our results also suggest that our conceptualization of semantic legal metadata can be further improved with new information elements and relations.

***Index Terms*—Legal Requirements, Legal Metadata, Natural Language Processing, Smart Search, Question Answering.**

I. Introduction

Many information systems in domains such as healthcare, finance and taxation have to comply with the various laws and regulations that are pertinent to these domains. Nowadays, regulations like the General Data Protection Regulation [1] introduce provisions on systems that previously had only sparse regulatory constraints. As a consequence, when eliciting requirements for such systems, requirements analysts often have to examine the relevant laws in order to identify the software-related concepts and the statements that lead to legal requirements.

Support in searching the law is provided by legal publishers, but only for legal professionals. This kind of support for legal advice is inadequate for requirements analysts, who have different concerns and objectives.

One way to help requirements analysts in their understanding of the law and in the derivation of legal requirements is by enabling them to search the law based on semantic metadata. Examples of such search include looking for (1) the stakeholders of a system, (2) the stakeholders' rights and duties, and (3) the relationships that hold between the stakeholders and the system entities [2].

In a previous collaboration with the Government of Luxembourg [3], we proposed a conceptual model of semantic legal metadata for requirements engineering (RE). Our set of metadata provides information about the statements and the phrases contained in legal provisions. We further devised an approach to automatically extract our proposed metadata types using natural language processing (NLP).

In this paper, we organize the semantic legal metadata extracted using our previously developed solution into a knowledge base whose intended purpose is to support a legal query system in the context of requirements elaboration. We provide an implementation of such a query system using Semantic Web technologies. We then utilize our implementation for conducting an industrial feasibility analysis, using Luxembourg's Income Tax Law as a case study. This law is in French; but, throughout this paper, we use English translations for the excerpts we borrow from the law for exemplification. Finally, we reflect on the lessons learned from our experience.

Our work focuses on the following two research questions:

- **RQ1: Is our existing conceptual model for semantic legal metadata expressive enough to provide an adequate answer to the questions that a requirements analyst may ask when identifying and elaborating legal requirements?**
  RQ1 investigates the questions that a requirements analyst may ask, and how she can formulate them in a query system.
- **RQ2: Does our metadata-based query system yield accurate results?**
  RQ2 is aimed at measuring the accuracy of the results returned by the queries, as well as building insights into how these queries should be posed so as to obtain answers that are as precise and complete as possible.

Contributions. In light of the relevant literature in RE and artificial intelligence and law (AI and Law), we identify five questions related to legal requirements elaboration. We transform these questions into templates of queries for a knowledge base containing semantic legal metadata, and assess the accuracy of the query system based on selected queries.

Our results show that semantic metadata can be successfully leveraged for retrieving high-level information such as definitions, articles and prescriptions. Nevertheless, the results also indicate that there is room for improving the metadata that underlies our query system. In particular, we observe that the metadata should be enhanced with certain additional information in order to enable finer-grained analysis of legal provisions at the level of phrases. We believe that the experience we have gained through our work is a useful stepping stone toward providing computerized assistance in the specification of legal requirements.

Structure. The remainder of the paper is organized as follows. Section II discusses background and related work. Section III introduces our legal query toolchain. Sections IV and V address RQ1. Section VI addresses RQ2. Section VII discusses threats to validity. Section VIII concludes the paper.

II. Background and Related Work

In this section, we review the relevant literature on RE, specially requirements mining, and on AI and Law, specially legal knowledge representation.

A. Search Systems in RE

Mining requirements. Using NLP and machine learning for identifying and deriving requirements from textual sources of information has received a lot of attention in recent years. Strands of work include requirements gathering from (1) requests for proposals [4], (2) appstore reviews [5], (3) Twitter feeds [6], [7], (4) user manuals [8] and (5) log files [9]. However, being concerned with feature extraction, these contributions do not target legal analysis or the development of regulated systems, and, more importantly for what concerns this paper, they are not targeted at querying a knowledge base looking for specific information on a given concern or topic.

Legal requirements analysis. There is considerable research on extracting semantic information from legal provisions with the objective of helping with legal compliance. Breaux and Antón [10] propose an upper-level ontology aimed at classifying statements and their constituents. Maxwell and Antón [2] propose a taxonomy of rights, duties, actors and rules' preconditions for elaborating compliance rules. Massey [11] uses a taxonomy of legal concepts for traceability mapping of requirements to legal texts. Frameworks like Nomos [12], GaiusT [13], NomosT [14] and LegalGRL [15] are aimed at representing legal provisions as goal models. Apart from GaiusT and NomosT, none of these contributions provide tool support for automatically extracting semantic information. In addition, since these threads of work aim at supporting requirements analysts in eliciting legal requirements from specific legal provisions, they do not address the issue of retrieving such provisions in the first place.

Query systems in RE. Query systems in RE are seen as enablers for the analysis of large systems, in particular in the context of traceability management. Mäder and Cleland-Huang [16] propose VTML, a graphical modeling language for visualizing and querying traceability links. Sannier and Baudry [17] propose the INCREMENT tool for the analysis of safety standards and regulations. In this work, standards and regulations can be represented as models, and their content can be searched through a query system based on information retrieval. However, the work only considers structural elements, with a shallow level of provision classification. Pruski et al. [18] propose TiQi, a framework to convert natural languages queries into SQL for querying traceability links. Kanchev et al. [19] propose the Canary approach to query a database of RE-related annotations of online discussions. Canary enables a requester to find discussions related to a given requirement as well as argumentation elements for prioritizing requirements. Again, the granularity level of the metadata is rather shallow as it only considers RE objects (requirement and solution), argumentation objects (support and rebuttal), user information, and the scoring of discussions.

B. Legal Search and Analysis in AI and Law

Opijnen and Santos [20] identify two types of IT systems in the legal domain: (1) legal expert systems (LES) and (2) systems for legal information retrieval (LIR). While LES rely on Semantic Web technologies (taxonomies, controlled vocabularies, legal ontologies) to provide a specific answer to a query, LIR is more concerned with retrieving relevant legal documents (or parts thereof) in larger corpora.

Legal expert systems (LES). Examples of LES that rely on semantic metadata are abundant and the large majority of them are based on legal ontologies built using OWL or RDF. For instance, Quaresma and Rodrigues [21] propose a question answering system for the Portuguese criminal law. This approach relies on Prolog and is paired with an ontology supporting the semantic analysis and the pragmatic interpretation of the questions. The approach has nevertheless not been tested on judicial texts but rather on newspaper articles, and the results, although encouraging, are not high quality enough for practical applications. Other examples rely on rule languages such as LegalRuleML [22]: Wyner et al. [23] perform manual annotation on legal texts in order to answer a set of queries concerning the legal semantics of the provisions; Gandon et al. [24] provide an ontological extension of LegalRuleML to support SPARQL queries that go beyond the expressiveness of OWL 2. These systems are nonetheless only presented at the level of proof of concept and are not implemented in a concrete use case.

Legal information retrieval (LIR). In Legal information retrieval, we distinguish between (1) systems based on ontologies [25], and (2) systems using NLP technologies.

Within the first type of systems, the Légilocal system [26] and the Nomothesia platform [27] propose solutions for authorities to manage local regulations implementing national laws in France and Greece, respectively. Their conceptual models cover legal document types as well as structural, geographical and topographical metadata, but do not provide semantic metadata about the content of the provisions.

Within the second type of systems, relevant contributions include Do et al. [28], Adebayo et al. [29] and Collarana et al. [30], all of which aim to retrieve relevant documents. They do not employ a particular conceptual model to formalize the content of the law, and therefore are not able to answer specific queries about content. In Eunomos [31], Boella et al. developed a conceptual model combined with NLP capabilities. Eunomos is a document management system that

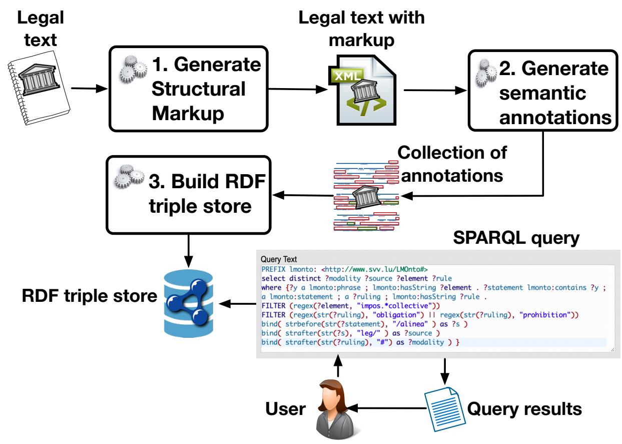

Fig. 1. Our Toolchain

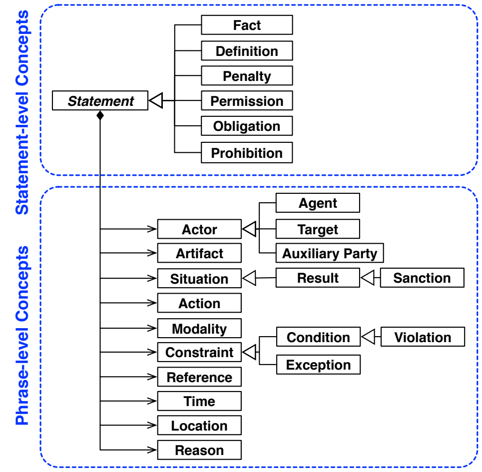

Fig. 2. Our Conceptual Model for Semantic Legal Metadata

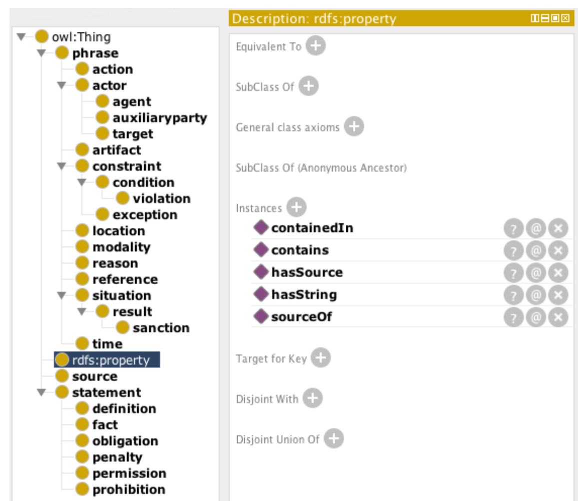

Fig. 3. The RDF Schema in Protégé

is mainly focused on vocabulary building; more specifically, it is aimed at reconciling and aligning vocabularies across the European legislation landscape. The conceptual model of Eunomos focuses on domain keywords, domains, and cross-references between articles.

Our approach attempts to bridge the gap between approaches with deep semantic and interpretation capabilities, but almost no tool support, and approaches that provide some support for automatic metadata generation, but lacking means for semantic analysis. In particular, we rely on a domain-independent conceptual model of semantic legal metadata with automated support for metadata extraction from legal texts.

III. Our Toolchain

In this section, we describe our toolchain for a query system based on legal metadata. This toolchain has been developed in collaboration with Luxembourg's Central Legislative Service (*Service Central de Législation*, hereafter SCL) – the government agency responsible for the publication of all legislative acts in Luxembourg through the online official portal Legilux (http://legilux.public.lu).

The overall workflow of the toolchain is depicted in Fig. 1. The first step is to identify the structure of an input legal text and convert the text into a markup document in XML. This step leverages our existing infrastructure for generating structural metadata [32]. The generated markup document includes annotations for provisions at the article level (using Uniform Resource Identifiers - URIs) as well as for cross-references. These structural annotations are essential for providing traceability between the legal text fragments and the legal statements expressed therein. Resources are named using ELI templates [33]. ELI (the European Legislation Identifier) is an EU-endorsed initiative aimed at providing a unified legal referencing mechanism. Its ultimate goal is to facilitate access, exchange and reuse of legal knowledge across the EU member states.

The second step of our approach is semantic metadata extraction. Here, the markup document from the first step is converted into individual statements. Each statement is subsequently processed in order to automatically extract semantic metadata for the statement itself as well as the phrases contained therein. The metadata annotations produced in this step follow the conceptual model developed in our previous work [3] and shown in Fig. 2. In this paper, we do not elaborate further on the conceptual model; the reader can find definitions, examples and discussions in our prior work.

The third step is concerned with building a knowledge base that can be queried. Here, for the representation of our metadata, we have chosen RDF (Resource Description Format) – a metadata model and a W3C recommendation since 1999 [34]. Our RDF schema, shown in Fig. 3, is a direct implementation of the conceptual model of Fig. 2. Fig. 4 presents a snippet of the schema and introduces two predicates aimed at building the RDF graph. The first one, *contains* (with its inverse *containedIn*), links a statement and the phrases enumerated therein, while the second one, *hasSource* (with its inverse *SourceOf*), links a statement and its source, i.e. the article to which the statement belongs.

For querying the RDF triple store, we use SPARQL (SPARQL Protocol and RDF Query Language), the most

```
<rdfs:Class rdf:ID="definition">
  <rdfs:subClassOf rdf:resource="#statement"/>
<rdfs:property rdf:ID="contains">
  <rdfs:domain rdf:resource="#statement"/>
  <rdfs:range rdf:resource="#phrase"/>
</rdfs:property>
<rdfs:property rdf:ID="hasSource">
  <rdfs:domain rdf:resource="#statement"/>
  <rdfs:range rdf:resource="#source"/>
</rdfs:property>
```

Fig. 4. Excerpt from the RDF Schema

popular query language for RDF graphs and a W3C recommendation since 2008 [35]. Two factors were decisive in the choice for this technical implementation: (1) it is a convenient and scalable way to handle a large amount of metadata, and (2) SCL has significant experience with these two technologies and is already using them in the Legilux portal.

IV. Most Relevant Questions to Legal RE

Before analyzing the adequacy of our conceptual model for answering RE-related questions, we first need to identify these questions. In this section, we analyze a typical scenario as well as examples from the literature in order to identify a list of high-level questions that could be of interest to a requirements analyst working with legal texts.

RE questions. There is a long history of research regarding the kind of questions that software developers and requirements analysts are likely to ask during RE activities and for software evolution [36], [37], [38], [9]. Recently, Malviya et al. [39] have classified relevant questions for RE activities into nine families according to their purposes, among which we deem Business Rule Analysis (family 1), Requirements Elicitation (family 3), Process (family 5), Quality Assessment (family 7), Risk Management (family 8) and Stakeholder Analysis (family 9) to be the most relevant to legal requirements analysis.

We now describe a typical legal requirements elaboration scenario by listing the four essential knowledge extraction activities that an analyst needs to perform when dealing with a domain that is heavily regulated, e.g., taxes, trade, or data protection and privacy.

1) First, the analyst needs to **extract the relevant concepts of the domain** from the underlying legal texts [40].
2) For each relevant domain concept, the analyst then has to extract the applicable authoritative **definition**, in order to align her understanding of the domain with what is envisaged by the law.
3) Once the relevant domain concepts have been identified and defined, the analyst needs to extract the **prescriptions and conditions that apply** to these concepts in order to elaborate legal requirements.
4) Finally, she may be interested in extracting the possible consequences of breaching the law in order to **assess risks** and prioritize requirements [41].

Next, we elaborate each of the activities presented above and identify the practical questions that the analyst may ask in order to extract the required information. We support our choice of questions with examples of similar questions from the literature, with their wording adapted to the tax domain.

Domain concept extraction. **Domain concepts** broadly include the stakeholders of a system, the objects the system handles, and the processes it has to perform or take part in [40].

The elicitation of domain concepts corresponds to multiple questions by Malviya et al. [39], notably those having to do with business rule analysis (family 1, e.g., *"list all business objectives"*) and stakeholder identification (family 9, e.g., *"for a given requirement, who are the stakeholders of interest?"* and *"what kind of users are going to use the system?"*).

We therefore introduce the first question for our query system (Q1): **What are the relevant concepts of the domain?**

Domain concept definition. Quaresma et al. [21] and Gandon et al. [24] propose questions such as *"what is a taxpayer?"*. These questions are aimed at retrieving **definitions** and indicative statements [42] from which the analyst can derive dictionaries or taxonomies of concepts [43], [10]. Jackson's questions [40], *"what do we mean by 'y is a company'?"*, *"what do we mean by 'z is a kind of commercial profit'?"*, and *"what do we mean by 'y realizes z'?"* go even further in that they effectively attempt to build a domain model.

In addition, identifying the terms that lack an authoritative definition allows answering the questions of Malviya et al. [39] concerning stakeholder identification (see above) and project glossary extraction (family 7, e.g.,*"find all ambiguous words in the requirements"* and *"are there weak words in the document?"*).

We introduce our second question for our query system (Q2): **What are the definitions for a given domain concept?**

Prescriptions and conditions that apply. After retrieving and classifying all relevant domain concepts, the analyst needs to find in the law all the restrictions and constraints related to these concepts. This means identifying the obligations, permissions and prohibitions (from now on collectively referred to as **prescriptions**) that involve these concepts.

Concrete examples are provided by Collarana et al. [30] (*"what shall a company do with regard to tax obligations?"*) and Wyner et al. [23] (*"what prohibitions apply to foreign companies?"* and *"what obligations have been placed on which entities, e.g., resident taxpayer?"*). Extracting prescriptions related to specific concepts corresponds to answering the questions of Malviya et al. [39] aimed at requirements elicitation (family 3, e.g., *"which requirements are related to requirement x?"* and *"need to know the regulatory compliance requirements pertinent to process x"*), and at reviewing requirements to uncover errors or inconsistencies (family 7, e.g., *"did I miss any requirements from stakeholders?"*).

We introduce our third question (Q3): **What are the prescriptions for a given domain concept?**

Interestingly, restrictions and constraints on a domain concept do not only entail prescriptions but also **conditions** and **exceptions** which determine whether that concept is included or excluded from the area of application of a given prescription. For example, the statement *"if the transfer profit [...]*

includes a capital gain realized on an immovable property, the capital gain may, upon request, be immunized [...]" includes a clause for "capital gain" ("realized on an immovable property"). This clause does not express the prescription introduced by the statement *"the capital gain may [...] be immunized"*, but rather identifies the subset of capital gains to which the prescription applies.

Capturing the legal conditions and exceptions is linked to the following question in Malviya et al. [39]: *"what type of constraints are embedded in this rule?"* (family 1). Capturing these conditions and exceptions is important not only in order to know the conditions of validity of a prescription in a given context, but also to understand which constraints apply to the business processes that would need to be implemented in the system-to-be. From the analysis of conditions and exceptions, the analyst will be able to extract, among other things, the time or duration constraints related to activities, the input conditions that trigger an activity, and information about the sequencing of different activities. This is exemplified by Robertson & Robertson [44] according to which *"time constraints can be imposed to enable the product to meet a window of opportunity [...] or to satisfy many other scheduling demands"*.

Observing the important role that conditions and exceptions play during requirements elaboration, we introduce a fourth question (Q4): **Which conditions and exceptions apply to a given domain concept?**

Risk assessment. Malviya et al. [39] identify risk management (family 8) and more particularly compliance analysis as important RE purposes. However, the question shown as an example of compliance analysis (*"what are the regulations to comply with?"*) is too abstract and does not capture the essence of legal risk [45], [46], which is not merely the identification of the laws to comply with, but also the risk of losses from non-compliance. In the law (and in our conceptual model), **sanctions** identify the concrete consequences of breaching a legal requirement: as such, *sanctions* are the source of legal risks. Wyner et al. [23] ask the general question *"what are all the offenses and associated penalties?"*.

Following a similar line of reasoning, we introduce the fifth question (Q5): **What are the sanctions for a given breach?**

Our question is more specific than the one in Wyner et al. [23] and Malviya et al. [39] in that it retrieves only the sanctions related to a given offense.

The five questions introduced in this section may not be sufficient to gain a complete understanding of the law from the perspective of a legal expert. Nonetheless and from an RE standpoint, being able to answer these questions is a critical step towards having a systematic and reliable process for the elaboration of legal requirements.

V. Adequacy of Semantic Metadata for Extracting Requirements-related Information

In this section, we first map onto our conceptual model of Fig. 2 the key notions that underlie the questions identified in Section IV. We then convert the questions into SPARQL queries for automation purposes. By doing so, we assess whether our conceptual model is sufficiently expressive to support the extraction of requirements-related information from legal texts.

TABLE I
Mapping between Questions and Metadata Types in our Conceptual Model

Question	Related Metadata Types in our Conceptual Model
Q1. What are the relevant concepts of the domain?	actor, agent, target, auxiliary party, artifact, situation, location, time
Q2. What are the definitions for a given domain concept?	definition
Q3. What are the prescriptions for a given domain concept?	obligation, prohibition, permission
Q4. Which conditions and exceptions apply to a given domain concept?	constraint, condition, exception
Q5. What are the sanctions for a given breach?	penalty, violation, sanction

A. Mapping the Questions onto the Existing Metadata Types

The relationship between the questions and the metadata types in our conceptual model is explained below and summarized in Table I.

Q1 (*What are the relevant concepts of the domain?*) aims at characterizing all the relevant domain concepts in a legal text. Jackson [40] identifies domain concepts as stakeholders, objects and processes. In our conceptual model, stakeholders correspond to the phrase-level metadata type **actor** and its subconcepts. Objects correspond to **artifact**. Processes correspond to **situation**. In some circumstances, **location** and **time** may also represent relevant information to elicit. In order to limit the results of the query, one can ask more specific questions related to specific metadata types, e.g., "*what are the relevant stakeholders in the domain?*".

Q2 (*What are the definitions for a given domain concept?*) aims at retrieving definition(s) of a domain concept in a given legal text, e.g., the definition of "special expense" in Income Tax Law. In our conceptual model, answering Q2 means retrieving all statements annotated as **definition** and containing the domain concept of interest (e.g., "special expense").

Q3 (*What are the prescriptions for a given domain concept?*) aims at retrieving all the statements that express a legally enforceable order involving (but not necessarily targeting) a domain concept, e.g., a "resident taxpayer". These statements often provide important information for deriving legal requirements. In our conceptual model, answering Q3 means retrieving **obligation**, **permission** and **prohibition** statements containing the domain concept of interest.

Q4 (*Which conditions and exceptions apply to a given domain concept?*) aims at retrieving any text segment that makes a certain domain concept (ir)relevant to the law, thus making the law (in)applicable to that concept. In our conceptual model, answering Q4 means retrieving phrase-level concepts which are typed as **constraint**, **condition** or **exception** and which are related to the domain concept of interest, e.g., "a resident taxpayer who is not married".

```
1  PREFIX lmonto: <http://www.svv.lu/LMOnto#>
2  SELECT DISTINCT ?concept ?modality ?verbatim ?source
3  WHERE {?x a lmonto:phrase ; lmonto:hasString ?concept .
4  ?req lmonto:contains ?x ; a lmonto:statement ; a ?metadata_type ;
5  lmonto:hasString ?verbatim ; lmonto:hasSource ?s .
6  FILTER (regex(?concept, 'joint taxation'))
7  FILTER (regex(str(?metadata_type), "obligation")
8  || regex(str(?metadata_type), "permission")
9  || regex(str(?metadata_type), "prohibition"))
10 BIND ( strafter(str(?metadata_type), "#") as ?modality )
11 }
```

Fig. 5. The SPARQL Query for Q3 on "Joint Taxation"

Q5 (*What are the sanctions for a given breach?*) retrieves penalties (e.g., "to pay a EUR 12.500 fine") that are associated with a specified breach, e.g., "to forge a certificate". In our conceptual model, answering Q5 means retrieving the **sanctions** that are related to a specific **violation** within a **penalty** statement, e.g., "*The one who has forged a certificate [...] shall pay a fine ranging from EUR 251 to EUR 12.500*".

B. Translating the Questions into SPARQL Queries

We translate the questions identified in Section IV into SPARQL query templates. The questions conform to the RDF schema presented in Section III. Due to limited space, we do not present all our query templates here. Instead, we discuss only one of the templates, namely that of Q3 from Section IV. The other templates are similar.

Fig. 5 shows the template for Q3, instantiated for the concept of "joint taxation". Our template covers the SELECT, WHERE and BIND parts of the query, while the FILTER part must be specified manually.

Regarding the SELECT part (line 2), we are interested in retrieving the concept to be queried (?concept), the type of prescription that contains it (?modality), the verbatim (i.e. the original text) of the prescription (?verbatim) and its source (?source, i.e., the ELI resource). Regarding the WHERE part and the conditions for triggering a result (lines 3 to 5), we look for phrases that contain the queried concept and for statements of certain type(s) that contain these phrases. The FILTERs (lines 6 and 7) contain the parameters of the query (e.g., the concept of "joint taxation") as well as the metadata types of interest (e.g., *obligation* and/or *permission* and/or *prohibition*), specified manually. The query can be fine-tuned by specifying a selection of metadata types instead of all possible ones, e.g., by selecting only *obligations* in the query filters. The last line (BIND) is a simple post-processing directive aimed at displaying the string of the metadata type (e.g., "obligation") instead of its verbose description in RDF.

Answer to RQ1. We have been able to map onto the conceptual model of Fig. 2 all the elements that we have identified to be of interest for posing requirements-related questions over legal texts. By doing so, we provide confidence that our conceptual model is a suitable basis for developing an RE-oriented query system for legal texts.

VI. Accuracy of the Query System

In this section, we first report on the evaluation of our query system over a case study. We then reflect on our observations and lessons learned.

A. Case Study Description

Description. The main goal of our case study is to investigate the accuracy of our query system. The study was performed in collaboration with Luxembourg's Central Legislative Service (SCL). SCL already employs a range of Semantic Web technologies for legal text processing, and has considerable prior experience with legal metadata for coordinating and consolidating legal texts. SCL has shown interest in investigating the use of semantic legal metadata for the interactive querying of the law by various interested parties including lay individuals, legal experts, and software and business process analysts.

Data collection procedure. For our case study, SCL proposed to focus on the modified "Law of 4 December 1967 on Income Tax", in short the *Income Tax Law* (ITL). This law is the basis for Luxembourg's taxation system and has implications for the IT systems of the country's national tax administration bodies. The text is 210 pages long and has 241 articles in its 2018 version. On its own, the law does not cover the entire income tax domain as several secondary legislative texts further elaborate on specific aspects of the law. The law nevertheless already provides good coverage of the tax calculation policies that need to be implemented in eGovernment applications [47]. This characteristic makes ITL particularly relevant to requirements analysts. ITL is also reasonably contained in size: we can thus rely on human experts for high-quality manual analysis with reasonable effort.

To process the text of the law, we followed the metadata extraction process described in Fig. 1. Overall, we extracted ≈1770 statements, including ≈19000 semantic metadata items. In the process, some phrases were cloned in an attempt to provide self-contained sentences when lists are present, this being standard pre-processing in NLP tasks [48].

In our previous work [3], we assessed the accuracy of our semantic metadata extraction rules, which showed high – yet not perfect – accuracy (overall precision of 87.4% and overall recall of 85.5%). While we do keep track of the NLP errors in our evaluation of the results in Section VI-B, we do not reflect on the exact nature and root causes of these errors, noting that our conclusions about NLP are the same as those reached and presented in our previous work.

For the purposes of our case study, we focus on two topics: (1) taxes related to commercial activities, more precisely, the concept of "commercial profit", and (2) taxation of households, more precisely, the concepts of "indigenous income" and "joint taxation", where the latter is a phenomenon strictly related to the former. These are important topics that affect a large portion of Luxembourg's tax system. They are also addressed in various parts of the law, meaning that an analyst cannot, in normal conditions, scope the search by focusing on a small portion of the law.

Analysis procedure. We perform with respect to our concepts of interest a detailed examination of three questions, Q2, Q3 and Q4, of the five posed in Section IV. Specifically, when instantiated for the concepts of interest, these three questions will address the following: the definitions of "commercial profit" and "indigenous income" (Q2), the prescriptions that apply to "commercial profit" and "joint taxation" (Q3), and the conditions (and exceptions) that apply to "commercial profit" and "joint taxation" (Q4). Q3 is evaluated at two different levels of granularity: at the article level (Q3.1) and at the metadata level (Q3.2). Q3.1 enables us to examine whether our query system is able to identify the articles containing relevant information, whereas Q3.2 allows us to measure the ability of the query system to provide detailed information by returning relevant verbatim statements from the law. In total, we evaluate eight queries, that is, four queries (Q2, Q3.1, Q3.2 and Q4) for each topic.

Each question was independently analyzed by a different pair of authors among the first three authors. All authors have prior experience in legal informatics, with the second and last authors being legal experts. For each question, the first three authors manually investigated the text in order to retrieve the relevant elements together with the location where these elements appear. The retrieved elements were then compared, and discrepancies in the results were discussed among the three authors to form a ground truth for each question.

In order to build the SPARQL queries, we instantiated with the chosen concepts the templates discussed in Section V. We then evaluated the accuracy of these SPARQL queries by comparing their results against the ground truth.

In this study, we are also interested in measuring the effort required for an analyst to manually retrieve relevant information. To this end, we kept track of the time taken for the construction of the ground truth for each query. This enables us to provide a preliminary indication of the effort that could be saved by using our query system as opposed to a fully manual approach.

Our accuracy analysis is based on the following notions:

- A returned result is *relevant* if it is present in the ground truth. Relevant results count as true positives (TP).
- A returned result is *irrelevant* if it does not appear in the ground truth. Irrelevant results count as false positives (FP).
- A result is *missed* if it is not returned by the query but appears in the ground truth. Missed results count as false negatives (FN).

We measure the accuracy of our query system using the standard precision and recall metrics. Precision is computed as $|TP|/(|TP| + |FP|)$ and recall as $|TP|/(|TP| + |FN|)$.

Finally, we perform an error analysis over the FPs and FNs to identify potential areas for improvements. Specifically, we manually investigate the results in order to assess whether the errors could possibly have arisen from (1) NLP-related issues, (2) our set of extraction rules, (3) shortcomings in our conceptual model, or (4) the query system. In this paper, we discuss the errors related to only the last three points; as for the NLP-related issues, we refer the reader to our previous work [3] where we provide detailed discussions.

We make the following remarks about the two questions, Q1 and Q5, which we do not evaluate in depth here:

- Q1 retrieves a total of 4306 concepts when executed over ITL. A thorough vetting of all these concepts was not possible due to their broad scope. Nevertheless, to ensure the overall quality of the results, the first three authors collaboratively reviewed a random subset of 430 concepts from the output of Q1 (i.e., 10% of all the retrieved concepts). They deemed 376 of the concepts as being TPs and 54 as being FPs, thus giving a precision of ≈87,4%. Naturally, since we did not examine the Q1 results in their entirety, we cannot analyze recall. Nevertheless, there is no reason to suspect issues with recall for Q1, given the promising results from our previous work [3] for all the individual metadata types that Q1 retrieves.
- Q5 yielded no result for ITL. Our manual analysis of the law confirmed that the law is not concerned with stating penalties; this function is fulfilled by secondary legal acts.

B. Results

The results of the evaluation over the eight queries are presented in Table II. Columns 2 to 4 report the size of the ground truth for each query and the approximate evaluation time (rounded up or down to the nearest five minutes) spent by the pair of analysts who manually answered that query. Columns 5 through 8 report the results from the query system and their evaluation. Columns 9 and 10 report the accuracy measures for each query. Although we provide percentages for the precision and recall scores of all the queries, we note that where the results in the ground truth are few, these scores are not good indicators. Below, we discuss the accuracy of each query based on Table II.

Observations on the ground truth. Overall, the analysts needed on average ≈74.7 minutes to analyze the law for each query. While building the ground truth, it emerged that manually identifying the precise text spans in the law took the most time and effort, whereas identifying the information at article level was easier. Another interesting observation was that legal drafting practices can complicate the precise identification of relevant text segments in the provisions. This explains the gap observed between the two analysts in Q3.1 on "joint taxation" and in Q3.2 on "commercial profit", where one analyst had more difficulty precisely identifying relevant information in the law.

Results from Q2 queries. Regarding the search for *definitions*, our query has only one FN, where the concept of "commercial profit" is conveyed through the general notion of "profit", and this was not accounted for by our query. We elaborate this point in lesson learned L1 in Section VI-C.

We also have two FPs, which are due to the presence of the concept in a definition statement that defines another concept. To illustrate, consider the statement *"The following are considered to be indigenous incomes of non-resident taxpayers:[...] commercial profit within the meaning of Articles 14 and 15"*. This statement is a definition of "indigenous income", but not a valid definition of "commercial profit". This raises the issue of identifying the right *subject* of a statement. We elaborate this issue further in Section VI-C (see L4).

TABLE II
STATISTICS FOR THE QUERIES IN OUR CASE STUDY

Query	Ground Truth			Query Results	Relevant (TP)	Irrelevant (FP)	Missed (FN)	Precision	Recall
	Results	Effort							
		R1	R2						
Q2 - What are the definitions for commercial profit?	4	25'	35'	4	3	1	1	75,0%	75,0%
Q2 - What are the definitions for indigenous income?	3	60'	40'	4	3	1	0	75,0%	100,0%
Q3.1 - What articles contain prescriptions for commercial profit?	9	75'	105'	15	8	7	1	53,3%	88,9%
Q3.1 - What articles contain prescriptions for joint taxation?	19	60'	120'	21	19	2	0	90,5%	100,0%
Q3.2 - What are the prescriptions for commercial profit?	11	75'	105'	22	10	12	1	45,5%	90,9%
Q3.2 - What are the prescriptions for joint taxation?	30	60'	120'	33	26	7	4	78,8%	86,7%
Q4 - Which conditions and exceptions apply to commercial profit?	13	65'	120'	26	4	22	9	15,4%	30,8%
Q4 - Which conditions and exceptions apply to joint taxation?	23	75'	55'	50	17	33	6	34,0%	73,9%

Overall, the results show that our query is adequate for retrieving definitions from which the requirements analyst can later derive a dictionary or taxonomy of domain concepts.

Results from Q3 queries. Regarding the retrieval of *prescriptions*, our query system shows good recall scores at the level of both articles (Q3.1, ≈94.5% on average) and statements (Q3.2, 90% on average). The only FN in Q3.1 and Q3.2 for the query on "commercial profit" is, similarly to Q2, related to a prescription for the more general domain concept of "profit".

Regarding the four prescriptions for "joint taxation" that are not retrieved in Q3.2, the error analysis shows that they are due to NLP errors during metadata extraction.

As indicated by Table II, precision varies from 45.5% to 90.5%, depending on the query and its granularity. In particular, precision decreases when we search for information at the statement level, the retrieved results being finer-grained. Regarding the nine FPs in Q3.1, two are related to NLP errors. Five FPs are concerned with other domain concepts as discussed above for Q2. Another two FPs are related to the retrieval of delegation statements, which we elaborate momentarily.

Regarding the 19 FPs in Q3.2, two of them are due to NLP errors. Ten FPs are concerned with other domain concepts. The remaining seven FPs are related to the retrieval of the following statements:

- **Delegation statements**, which give powers to a secondary (legislative or administrative) legal instrument to specify or implement a given provision. An example such statement from the ITL is: *"A Grand-Ducal Regulation may extend to the partners taxed jointly the regulatory provisions [...] applicable to the spouses taxable jointly"*.
- **Party-to-the-law statements**, which express a legal requirement through the extension (or restriction) of the area of application of another legal provision [49]. An example from the ITL is: *"The provisions of Title I of this law are applicable for the determination of the taxable income and the net income of which it is composed [...]."*

Although it is important from a legal perspective, the information contained in delegation and party-to-the-law statements is often only marginally relevant to a requirements analyst since such information does not provide the details of the concrete prescriptions for the domain concept. We discuss the implications of delegation and party-to-the-law statements in Section VI-C (see L3).

Overall, the results show that our query is adequate for retrieving prescriptions related to domain concepts, but in order to increase precision we need finer-grained information in the conceptual model.

Results from Q4 queries. Q4 differs from the previous questions in that it is not aimed at retrieving entire statements or articles but precise phrases (*conditions* or *exceptions*).

The accuracy for Q4 is low: we have a total of 55 (33+22) FPs and 15 (9+6) FNs. The FNs shown in Table II are explained as follows: (1) four conditions are contained in statements that were already FNs in Q3; (2) another four are due to an erroneous NLP extraction given the complexity of the statements in question; finally, (3) seven conditions use common linguistic patterns for which no extraction rules exist due to a design decision (the resulting extraction rules would have been too generic and we wanted to avoid generating too many FPs).

As for the 55 FPs, 26 FPs come from statements that were already FPs in Q3. In these cases, solving the issues observed for Q3 (i.e., fixing the NLP errors and adding new statement types) would increase the precision for Q4 as well (on average, precision would increase from 21/76 ≈27.6% to 21/50 = 42%). A further 26 FPs are due to the fact that the retrieved conditions are valid conditions but concern a different concept from the one specified in the query. For instance, consider the following statement, which is an FP for Q4 on joint taxation: *"Remuneration paid to a relative other than a spouse who is taxed jointly with the operator is deductible as an operating expense if it is due under a service contract that meets the conditions to be specified by a Grand-Ducal Regulation."* Here, the conditions "if it is due under a service contract" and "the conditions to be specified" are not related to the concept of joint taxation but to the remuneration and to the service contract, respectively. The remaining three cases are due to NLP errors, resulting in *actions* being incorrectly tagged as *conditions* or *exceptions*.

Overall, the results show that our conceptual model needs to better handle the relationships between metadata in order to answer detailed questions such Q4. We further discuss this point in Section VI-C (see L5).

Answer to RQ2. Our query system can provide accurate results when searching for statement-level and article-level information (Q2 and Q3). Nevertheless, further work needs to be done for successfully answering queries that are aimed at retrieving phrase-level information (Q4).

C. Observations and Lessons Learned

In this section, we present the observations and lessons learned from our case study.

Observations concerning a domain taxonomy. Q2 and Q3 on "commercial profit" showed the importance of having a domain taxonomy for managing the existing hierarchy of terms and concepts that affect the queries. In the law, the concept of *commercial profit* is defined as a kind of *profit* alongside various other types of profit including *profit from agriculture*, *profit from forestry*, and *profit from independent activity*. These concepts also share subconcepts, such as *divestment profit*. Not knowing these relationships entails the risk of (1) missing general prescriptions on *profit* that span all the subconcepts, (2) missing prescriptions for *divestment profit* related to *commercial profit*, or (3) erroneously accounting for *divestment profits* that are not related to *commercial profit* but to other types of profit, e.g., agricultural profit.

Lesson learned 1 (L1): Having a domain taxonomy or an ontology available would enable easier exploration of the law and make the querying of the RDF graph easier. Building such a taxonomy (or ontology) can be facilitated by Q2 queries.

Observations concerning cross-references. In our queries, some of the returned results contained cross-references. In certain cases, the full content of a definition or prescription could only be retrieved by following those cross-references. Cross-references may contain information that has a direct impact on legal requirements [50], [51]. It is thus important that requirements analysts carefully consider and inspect cross-references during requirements elaboration. To help with this, our structural markup generator (Fig. 1) already detects and resolves cross-references.

Automatically navigating and analyzing cross-references can improve the quality of legal query results. However, doing so also raises the question of how far to extend the analysis: indeed, the targeted provision might in turn contain more cross-references, which should also be resolved and analyzed, with the risk of drifting too far from the initial scope of the analysis. Maxwell et al. [50] and Sannier et al. [52], among others, have taken steps in the direction of (automatically) interpreting cross-references. Despite these interesting contributions, more work is required before cross-references can be handled automatically and sufficiently accurately for questions-answering purposes.

Lesson learned 2 (L2): At this stage, from a practical perspective, it seems preferable to provide the cross-references as additional information and let the analyst decide how to handle them.

During the analysis of our results, we encountered two particular types of cross-references: (1) cross-references that delegate the implementation of a prescription to another legal text, and (2) cross-references that modify the application area of another provision. The presence of such cross-references affects the classification of the statements that contain them, as we elaborate next.

Observations concerning statement types. Statement may delegate the specification or implementation of a prescription to a future legal document, or modify the area of application of a statement. Although such a statement can be understood as an obligation, a permission or a prohibition, it should be considered as a delegation statement or as a party-to-the-law statement. From a legal standpoint, party-to-the-law statements have the effect of a prescription, but the sentence itself does not include the information that would enable the precise identification of the prescription, since this information is located elsewhere (i.e., in the referenced legal provision). Ideally, useful information would come from resolving the corresponding cross-reference [51]. However, performing this analysis and providing the information through the query system would require rethinking both our conceptual model of semantic legal metadata and the extraction rules we have developed for metadata extraction. We therefore leave this to future work.

Nevertheless, it would be useful to identify these statements in order to filter them out when they are deemed irrelevant by the analyst. This identification can be achieved by adding two boolean attributes (*isDelegation* and *isPartyToTheLaw*) to all statement-level metadata types in our conceptual model.

Lesson learned 3 (L3): Adding the notions of *delegation statement* and *party-to-the-law statement* in our conceptual model would offer easier exploration of the law, and provide a filtering mechanism to the analyst. Those statements can be detected by looking for cross-references within the *subject* of the statement. We elaborate on *subject* next.

Observations concerning the subject of a statement. The notion of *subject* in the literature identifies the addressee or main target of a legal provision [10], [12]. Linguistically, it corresponds most of the time to the semantic subject of the main clause. In our current conceptual model, this notion is addressed through the *agent* metadata type, which, however, can only specify actors. This notion could instead encompass all possible phrase-level concepts that can appear as addressees of the law, i.e., *actors*, *situations*, and *artifacts*. This way, when the actual human addressee is not explicitly mentioned in the statement, labeling as *subject* the addressed artifact or a situation would provide a first clue toward the identification of the real addressee. For example, consider the statement *"Compensation paid to a close relative other than the spouse taxable jointly with the operator is deductible as an operating expense [...]"*. Here, the *subject* is "compensation". However, one correct interpretation of the statement would be *"the taxpayer can deduct compensations paid to a close relative other than the spouse taxable jointly with the operator [...]"*, where the addressee is "the taxpayer paying the compensation".

The addressee may also correspond to a different element than the subject of the main clause, e.g., a *target*, and less commonly, an *auxiliary party*. Consider for instance the following statement: *"It is allowed for operators with regular accounts to include in the net assets invested goods [...]"*. Here, the addressee, namely, "operator", is not the linguistic subject of the sentence. This happens not only with impersonal verbs (i.e., verbs with no determinate subject), as in the example, but also with party-to-the-law statements, discussed above.

Lesson learned 4 (L4): Capturing the *subject* of a statement requires enhancing the conceptual model with a boolean attribute (*isSubject*) added to *actors*, *situations* and *artifacts*, as well as defining and implementing new extraction rules aimed at identifying the correct addressee of the legal provision.

Observations concerning fine-grained analysis. Looking at the results of Q4, we learned that, in order to successfully retrieve all the conditions related to a given domain concept while discarding those that are not, it is necessary to improve the conceptual model with relationships between metadata types. In practice, we need to account for the relationships between *actors*, *artifacts* and *situations* on one side and *constraints* on the other side.

Lesson learned 5 (L5): It seems useful to link *constraints* and their subconcepts, namely *conditions* and *exceptions*, to their related phrase-level concepts in our conceptual model. This requires an extension of the conceptual model as well as new extraction rules.

VII. Threats to Validity

The validity considerations most pertinent to our work are internal and external validity, as we discuss below.

Internal validity. The first threat to internal validity is related to the risk of misinterpreting (or having changing interpretations of) the provisions in the law when elaborating the ground truth for each query. This risk is minimized by the analysts having background in legal analysis and compliance. Second, while elaborating our queries and criteria for evaluation, we avoided as much as possible restricting alternative legal interpretations, in order to leave the final decision on the interpretation to the analyst. Third, each question was analyzed by a pair of analysts and the results were discussed and reconciled among all the analysts.

Another threat to internal validity is related to the alignment between the questions that we identified in Section IV and the SPARQL queries that we built. We note that the questions are simple, and thus there is a limited risk of misinterpretation. There remains, though, the risk that the query does not fully cover the initial question as observed in the results for Q4, due to potential limitations in the conceptual model. If present, such limitations would however also apply to a manual search, which, in the case of Q4, would leave the identification of conditions and exceptions totally in the hands of the analyst.

External validity. The main threat to external validity has to do with the generalizability of our results. Due to the effort-intensive nature of the tasks in our study (e.g., building the ground truth), we evaluated our queries on two topics only ("commercial profit" and "joint taxation"), among the many different topics that would need to be covered in relation to the Income Tax Law. There is a risk that our observations and suggestions for improvements would not readily generalize to other topics. Further studies that cover other legal domains and a more comprehensive list of topics therefore remain essential for validating the general applicability of our results.

A second threat to external validity is related to the size of the corpus. The law over which we posed our queries in this paper does not cover the entirety of its underlying domain (taxation) as there exists considerable secondary legislation providing implementation and enforcement details. Going for a larger corpus could have an impact, since the number of elements to retrieve and the ones that would actually be retrieved by a query system will inevitably increase. This gives rise to the risk that the analyst may be overwhelmed by large result sets. This risk is, however, only relevant for very broad queries such as Q1. In such situations, the analyst would still be able to scope the search to a specific context or document and thus obtain result sets of manageable sizes.

VIII. Conclusion

In this paper, we described an industrial experience aimed at helping requirements analysts to query legal texts. The work is a follow-up to our previous research on automated legal metadata extraction [3]. To build a query system, we convert the extracted metadata into RDF triples and populate a knowledge base using the resulting triples. We identified five important questions that requirements analysts are likely to ask when elaborating legal requirements. We proposed SPARQL query templates corresponding to each question and evaluated the accuracy of the templates through a case study on Luxembourg's Income Tax Law. Finally, we drew several lessons learned to guide future work.

Our analysis suggests that our conceptualization of legal metadata is a useful basis for smart legal search in the context of RE. Further, our empirical results show that we can accurately query for relevant information at the article and sentence level. At the same time, the results pinpoint areas for further improvement. First, we observe that certain drafting practices in legal texts pose challenges for our query system. Second, we identify possible enhancements to our legal metadata information such as an attribute identifying the concrete subject of a statement and additional relationships between metadata types.

Our future work includes implementing the lessons learned and evaluating their effect on our approach through additional case studies. Another interesting area for future work is investigating whether our existing query system can be augmented with techniques that can automatically derive legal requirements and compliance rules from legal texts.

Acknowledgments. Supported by SCL and the Luxembourg National Research Fund (FNR) under grants PUBLIC2-17/IS/11801776 and PoC16/11554296.

References

[1] The European Parliament and the Council of the European Union, "General data protection regulation," https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32016R0679.

[2] J. C. Maxwell and A. I. Antón, "The production rule framework: developing a canonical set of software requirements for compliance with law," in *Proceedings of IHI'10*, 2010, pp. 629–636.
[3] A. Sleimi, N. Sannier, M. Sabetzadeh, L. C. Briand, and J. Dann, "Automated extraction of semantic legal metadata using natural language processing," in *Proceedings of RE'18*, 2018, pp. 302–311.
[4] A. Falkner, C. Palomares, X. Franch, G. Schenner, P. Aznar, and A. Schoerghuber, "Identifying requirements in requests for proposal," in *Proceedings of REFSQ'19*, 2019, pp. 176–182.
[5] N. Jha and A. Mahmoud, "Mining user requirements from application store reviews using frame semantics," in *Proceedings of REFSQ'17*, 2017, pp. 273–287.
[6] E. Guzman, M. Ibrahim, and M. Glinz, "A little bird told me: Mining tweets for requirements and software evolution," in *Proceedings of RE'17*, 2017, pp. 11–20.
[7] G. Williams and A. Mahmoud, "Mining twitter feeds for software user requirements," in *Proceedings of RE'17*, 2017, pp. 1–10.
[8] T. Quirchmayr, B. Paech, R. Kohl, H. Karey, and G. Kasdepke, "Semi-automatic rule-based domain terminology and software feature-relevant information extraction from natural language user manuals," *Empirical Software Engineering*, 2018.
[9] I. T. Koitz and M. Glinz, "A fuzzy galois lattices approach to requirements elicitation for cloud services," *IEEE Transactions on Services Computing*, vol. 11, no. 5, pp. 768–781, 2018.
[10] T. D. Breaux and A. I. Antón, "Analyzing regulatory rules for privacy and security requirements," *IEEE Transactions on Software Engineering*, vol. 34, no. 1, pp. 5–20, 2008.
[11] A. Massey, "Legal requirements metrics for compliance analysis," Ph.D. dissertation, North Carolina State University, Raleigh, North Carolina, USA, 2012.
[12] A. Siena, J. Mylopoulos, A. Perini, and A. Susi, "Designing law-compliant software requirements," in *Proceedings of ER'09*, 2009, pp. 472–486.
[13] N. Zeni, N. Kiyavitskaya, L. Mich, J. R. Cordy, and J. Mylopoulos, "GaiusT: supporting the extraction of rights and obligations for regulatory compliance," *Requirements Engineering*, vol. 20, no. 1, pp. 1–22, 2015.
[14] N. Zeni, E. A. Seid, P. Engiel, S. Ingolfo, and J. Mylopoulos, "Building large models of law with NómosT," in *Proceedings of ER'16*, 2016, pp. 233–247.
[15] S. Ghanavati, D. Amyot, and A. Rifaut, "Legal goal-oriented requirement language (legal GRL) for modeling regulations," in *Proceedings of MISE'14*, 2014, pp. 1–6.
[16] P. Mäder and J. Cleland-Huang, "A visual language for modeling and executing traceability queries," *Software and System Modeling*, vol. 12, no. 3, pp. 537–553, 2013.
[17] N. Sannier and B. Baudry, "INCREMENT: A mixed MDE-IR approach for regulatory requirements modeling and analysis," in *Proceedings of REFSQ'14*, 2014, pp. 135–151.
[18] P. Pruski, S. Lohar, W. Goss, A. Rasin, and J. Cleland-Huang, "Tiqi: answering unstructured natural language trace queries," *Requirements Engineering*, vol. 20, no. 3, pp. 215–232, 2015.
[19] G. M. Kanchev, P. K. Murukannaiah, A. K. Chopra, and P. Sawyer, "Canary: Extracting requirements-related information from online discussions," in *Proceedings of RE'17*, 2017, pp. 31–40.
[20] M. van Opijnen and C. Santos, "On the concept of relevance in legal information retrieval," *Artificial Intelligence and Law*, vol. 25, no. 1, pp. 65–87, 2017.
[21] P. Quaresma and I. Pimenta Rodrigues, "A question answer system for legal information retrieval," in *Proceedings of JURIX'05*, 2005, pp. 91–100.
[22] M. Palmirani, G. Governatori, A. Rotolo, S. Tabet, H. Boley, and A. Paschke, "Legalruleml: Xml-based rules and norms," in *Proceedings of RuleML'11*, 2011, pp. 298–312.
[23] A. Wyner, F. Gough, F. Levy, M. Lynch, and A. Nazarenko, "On annotation of the textual contents of scottish legal instruments," in *Proceedings of JURIX'17*, 2017, pp. 101–106.
[24] F. Gandon, G. Governatori, and S. Villata, "Normative requirements as linked data," in *Proceedings of JURIX'17*, 2017, pp. 1–10.
[25] K. T. Maxwell and B. Schafer, "Concept and context in legal information retrieval," in *Proceedings of JURIX'08*, 2008, pp. 63–72.
[26] N. Mimouni, A. Nazarenko, and S. Salotti, "Answering complex queries on legal networks: A direct and a structured IR approaches," in *Proceedings of AICOL'17*, 2017, pp. 451–464.
[27] I. Chalkidis, C. Nikolaou, P. Soursos, and M. Koubarakis, "Modeling and querying greek legislation using semantic web technologies," in *Proceedings of ESWC'17*, 2017, pp. 591–606.
[28] P. Do, H. Nguyen, C. Tran, M. Nguyen, and M. Nguyen, "Legal question answering using ranking SVM and deep convolutional neural network," *CoRR*, vol. abs/1703.05320, 2017.
[29] K. J. Adebayo, L. D. Caro, G. Boella, and C. Bartolini, "An approach to information retrieval and question answering in the legal domain," in *Proceedings of JURISIN'16*, 2016.
[30] D. Collarana, T. Heuss, J. Lehmann, I. Lytra, G. Maheshwari, R. Nedelchev, T. Schmidt, and P. Trivedi, "A question answering system on regulatory documents," in *Proceedings of JURIX'18*, 2018, pp. 41–50.
[31] G. Boella, L. D. Caro, L. Humphreys, L. Robaldo, P. Rossi, and L. van der Torre, "Eunomos, a legal document and knowledge management system for the web to provide relevant, reliable and up-to-date information on the law," *Artificial Intelligence and Law*, vol. 24, no. 3, pp. 245–283, 2016.
[32] N. Sannier, M. Adedjouma, M. Sabetzadeh, L. C. Briand, J. Dann, M. Hisette, and P. Thill, "Legal markup generation in the large: An experience report," in *Proceedings of RE'17*, 2017, pp. 302–311.
[33] "ELI - the European Legislation Identifier," http://eur-lex.europa.eu/eli-register/about.html.
[34] World Wide Web Consortium, "Resource Description Framework (RDF)," https://www.w3.org/RDF/.
[35] ——, "SPARQL Query Language for RDF," https://www.w3.org/TR/rdf-sparql-query/.
[36] B. Nuseibeh and S. M. Easterbrook, "Requirements engineering: a roadmap," in *Proceedings of ICSE-FOSE'2000*, 2000, pp. 35–46.
[37] D. Zowghi and C. Coulin, "Requirements elicitation: A survey of techniques, approaches, and tools," in *Engineering and Managing Software Requirements*, A. Aurum and C. Wohlin, Eds. Springer, 2005, ch. 2, pp. 19–46.
[38] T. Fritz and G. C. Murphy, "Using information fragments to answer the questions developers ask," in *Proceedings of ICSE'10*, 2010, pp. 175–184.
[39] S. Malviya, M. Vierhauser, J. Cleland-Huang, and S. Ghaisas, "What questions do requirements engineers ask?" in *Proceedings of RE'17*, 2017, pp. 100–109.
[40] M. Jackson, "The world and the machine," in *Proceedings of ICSE'95*, 1995, pp. 283–292.
[41] A. K. Massey, P. N. Otto, and A. I. Antón, "Prioritizing legal requirements," in *Proceedings of RELAW'09*, 2009, pp. 27–32.
[42] P. Zave and M. Jackson, "Four dark corners of requirements engineering," *ACM Transanctions on Software Engineering and Methodololgy*, vol. 6, no. 1, pp. 1–30, Jan. 1997.
[43] N. Kiyavitskaya, N. Zeni, T. D. Breaux, A. I. Antón, J. R. Cordy, L. Mich, and J. Mylopoulos, "Automating the extraction of rights and obligations for regulatory compliance," in *Proceedings of ER'08*, 2008, pp. 154–168.
[44] S. Robertson and J. Robertson, *Mastering the Requirements Process*. Addison-Wesley Professional, 2006.
[45] B. of International Settlements, "Basel II: International convergence of capital measurement and capital standards: a revised framework," 2004.
[46] M. Laycock, *Operational Risk Reporting Standards (ORRS) Edition 2011*, ORX Association, 2012.
[47] G. Soltana, N. Sannier, M. Sabetzadeh, and L. C. Briand, "Model-based simulation of legal policies: Framework, tool support, and validation," *Software & Systems Modeling*, vol. 17, no. 3, pp. 851–883, 2018.
[48] F. DellOrletta, S. Marchi, S. Montemagni, B. Plank, and G. Venturi, "The SPLeT2012 shared task on dependency parsing of legal texts," in *Proceedings of SPLeT'12*, 2012, pp. 42–51.
[49] M. Ceci, T. Butler, L. O'Brien, and F. Al Khalil, "Legal patterns for different constitutive rules," in *Proceedings of AICOL'18*, 2018, pp. 105–123.
[50] J. C. Maxwell, A. I. Antón, P. P. Swire, M. Riaz, and C. M. McCraw, "A legal cross-references taxonomy for reasoning about compliance requirements," *Requirements Engineering*, vol. 17, no. 2, pp. 99–115, 2012.
[51] N. Sannier, M. Adedjouma, M. Sabetzadeh, and L. C. Briand, "An automated framework for detection and resolution of cross references in legal texts," *Requirements Engineering*, vol. 22, no. 2, pp. 215–237, 2017.
[52] ——, "Automated classification of legal cross references based on semantic intent," in *Proceedings of REFSQ'16*, 2016, pp. 119–134.