Semi-automated Vocabulary Building for Structured Legal English

Shashishekar Ramakrishna and Adrian Paschke

Department of Computer Science, Freie Universität Berlin,
Königin-Luise-Str. 24-26,
14195 Berlin, Germany
shashi792@gmail.com, paschke@inf.fu-berlin.de

Abstract. Structured English has been applied as computational independent language for defining business vocabularies and business rules, e.g., in the context of OMG's Semantics and Business Vocabulary Representation (SBVR). It allows non-technical domain experts to engineer knowledge in natural language, but with an underlying semi-formal semantics which eases the automation of machine transformation into formal knowledge representations and logic-based machine interpretation. We adapt this approach to the legal domain in order to support legal domain experts in their task to build legal vocabularies and legal rules in Structured English from legal texts. In this paper we contribute with a semi-automated vocabulary and rule development process which is supported by automated suggestions of legal concepts computed by a semantic legal text analysis. We implement a proof-of-concept in the KR4IPLaw tool, which enables legal domain experts to represent their knowledge in Structured English. We evaluate the proposed approach on the basis of use cases in the domain of IP and patent law.

Keywords: Controlled Natural Language, SBVR, Structured English, Legal Norms, LegalRuleML.

1 Introduction

Typically there exists a gap concerning the understanding of the knowledge from a particular domain between a domain expert and a knowledge engineer who models such domain knowledge - often in a structured, formal language- for its use in (semi-)automated reasoning. Such a problem can be easily seen in the legal domain, wherein, the cost associated with not reducing such gap is substantially high [1].

Structured English (SE) has be applied as Computational Independent Model (CIM) language for defining business vocabularies and business rules. For instance, in the context of OMG's Semantics and Business Vocabulary Representation (SBVR), SE provides an efficient solution to the problem [2]. It allows non-technical domain experts to engineer knowledge (vocabularies and rules) in natural language, but with an underlying semi-formal semantics defined in the

A. Bikakis et al. (Eds.): RuleML 2014, LNCS 8620, pp. 201–215, 2014.

SBVR standard, which eases the automation of machine transformation into formal knowledge representations (KRs)[1] and logic-based machine interpretation. However, the problem is still, that the manual modelling of the legal vocabularies from legal text is one of the most time consuming parts of the legal knowledge engineering [4]. Non-existence of a global public/privately shared vocabulary, makes the task of building legal vocabulary more tedious and time consuming. In this paper we contribute with a semi-automated vocabulary and rule development process which is supported by automated suggestions of legal concepts computed by a semantic legal text analysis.

The paper is structured as follows. Section 2, introduces and compares a subset of existing controlled natural languages for their use in legal domain. Sections 3 and 4 deal with the the use of SBVR-SE and in Section 5 we illustrate with an example how SBVR-SE could be used as a semi-formal KR format to represent legal information. In Section 6 we compare a subset of existing knowledge extraction tools for their use in legal domain and propose a recommender system to semantically enrich the legal information represented in previous sections. Section 7, deals with the transformation of legal knowledge from a Computational Independent Modelling (CIM) layer to a Platform Independent Modelling (PIM) layer and finally to a Platform Specific Modelling (PIM) layer. Section 8, concludes the paper and presents some future directions.

2 Controlled Natural Language, 'CNL'

Controlled Natural Language (CNL) is a subset of natural language that can be accurately and effectively processed by a computer, because it avoids semantic ambiguity and supports natural language processing with its controlled grammar. Although controlled CNLs are expressive enough to allow natural usage by a non-specialist.

There exists a wide variety of CNL's, amongst them we consider a subset of CNLs to study their applicability to our problem domain of semi-formal knowledge engineering:

- Attempto Controlled English (ACE): ACE is a CNL which includes restricted syntax and a restricted semantics (of base (English) language) described by a small set of construction and interpretation rules [5].
- SBVR Structured English: SBVR-SE is a CNL originally developed for representing business rules. It is more reliable for automatic interpretation due to its high syntax restrictions. It ignores the grammatical structure followed by its peer base language when representing the same rule/statement.
- Drafters Language: Drafters Language is a CNL originally developed for DRAFTER-II system. It works on a conceptual authoring approach which provided a relatively simple pseudo-text to specify a complex configuration of action and object entities and the relations between them [6].

[1] E.g., in [3], we define a transformation process with a modal first-order semantics.

– Massachusetts Legislative Drafting Language: Is a CNL developed for describing legal texts (originally for Massachusetts Senate). It provides a uniformity in drafting style by specifying a restricted syntax, restricted semantics and restricted document structure [7].

To compare the efficiency of different CNL's we use the evaluation methodology as proposed by Kuhn [8]. The evaluation is done based on four parameters described below:

– Precision: Shows the degree to which the meaning of text can be directly retrieved from its textual form.
– Expressivity: Describes the range of propositions that a certain language is able to express.
– Naturalness: Describes how close the language is to its base English (base language of considered problem domain) language.
– Simplicity: Describes simplicity/complexity of exact and comprehensive language description.

Fig 1, compares the four CNL's based on the four parameters discussed above. From the Figure we see that two out of four CNL's, i.e. SBVR-SE and ACE seem to fulfill the requirements required to represent our problem domain. Legal practitioners being both the authors and end-users of CNL based systems, we need to add another evaluation parameter 'learning curve'. From a legal practitioners' point of view, the learning curve involved in ACE seemed to be higher than that involved in SBVR-SE.

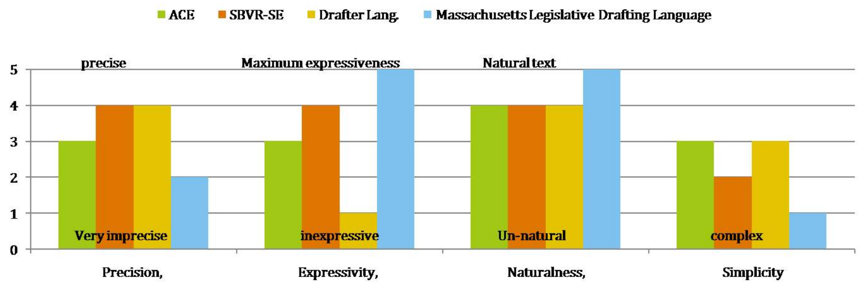


Fig. 1. Comparision of CNL's

3 SBVR Structured English

The OMGs Model Driven Architecture (MDA) [9] provides a basis for representing information on different layers of KR models (CIM, PIM and PSM). Semantic Business Vocabulary and Business Rules, SBVR [2], is an ISO terminological dictionary (vocabulary) for defining business concepts and rules. SBVR works on the Computational Independent Model (CIM) layer of the OMGs MDA. It

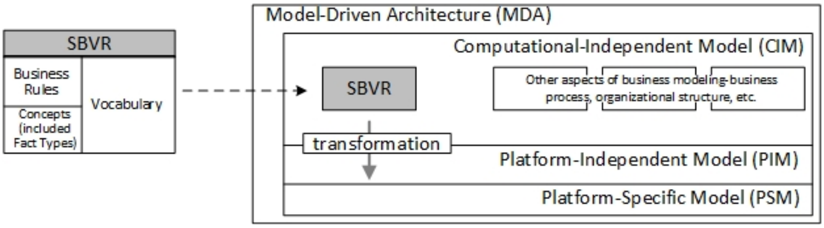


Fig. 2. SBVR position in MDA (adapted from [2])

suggests the use of Structured English (SE), a computational-independent English (natural) with a structured syntax for representing business vocabularies and business rules. SBVR captures the structural and behavioral aspects of business processes, as well as the policies that should guide the business behavior in certain situations. A core idea of business rules formally supported by SBVR is the following: Rules build on facts, and facts build on concepts as expressed by terms. Terms express business concepts; facts make assertions about these concepts; rules constrain and support these facts [2]. Fig 2 depicts the relation of SBVR and OMGs MDA.

4 Semi-formal KR in Legal Domain

The power of SBVR is disclosed by the fact that the SBVR specification itself was formally written in SBVR Structured English, 'SSE' [2]. The use of SBVR in legal domain was first proposed by Johnsen and Berre [10] [11]. In [4] we showed how OMGs MDA could be viewed in the domain of patent law, wherein, we provided the first ideas on using SBVR SE in patent law domain. We adapt the approach of the OMG Semantic Business Vocabulary and Business Rules [2] (OMG SBVR) standard to the patent law domain. Fig 3 gives an overview of it.

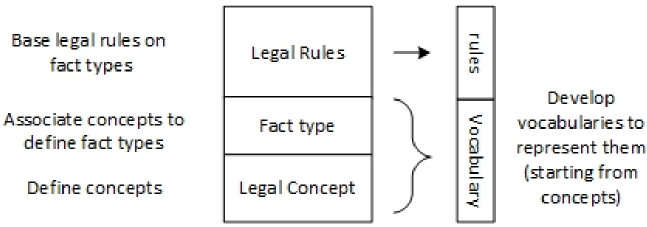


Fig. 3. Building legal vocabulary

SBVR defines the vocabulary and rules for describing the legal semantics using SSE. Even though SSE does not provide all the expressivity required for translating the procedural rules into a formal reasoning, the simple approach of SSE helps the end users (i.e. the domain experts and legal practitioners) to define their legal vocabularies and rules in a more understandable manner, which at the same time can be also interpretable by the computer. Like in SBVR, we define the legal (procedural/substantive) rules in a structured natural language (a Structured English syntax) using predefined **legal vocabularies**, consisting of **legal concepts** (concepts which have a meaning in the legal tradition, e.g. claim construction vocabulary) in template-based **legal rules**.

– **Legal Noun concepts**, which correspond to legal concepts.
– **Legal Verb concepts**, which correspond to relationships between legal concepts.
– **Legal rules**, which constrain these relationships so that they can be used to define consistent and complete arguments.

Legal concepts represented by noun concepts must be explicitly defined with the intended semantics given in an authoritative source or otherwise acknowledge by implicit pragmatic understanding (the ordinary natural language meaning of the term used). Verb concepts can only use such recognized noun concepts as their terms. The legal rules can then be constructed using the "*if ... then ...*", "*at least*", "*each*" as well as definitional alethic and behavioral deontic legal norm modalities (*"obliged", "permitted" ...*), etc. The following example in the next section illustrates its use.

5 Example

To illustrate the use of SSE in the legal domain, we consider legal (procedural) rules followed by an examiner in evaluating the essential subject matter requirement as defined in Paragraph ¶ 7.33.01 of United States Patent Law [12] - which states as follows

¶ *7.33.01 Rejection, 35 U.S.C. 112, 1st Paragraph, Essential Subject Matter Missing From Claims (Enablement)*

Claim [1] rejected under 35 U.S.C. 112, first paragraph, as based on a disclosure which is not enabling. [2] critical or essential to the practice of the invention, but not included in the claim(s) is not enabled by the disclosure.

1. *This rejection must be preceded by form paragraph 7.30.01 or 7.103.*
2. *In bracket 2, recite the subject matter omitted from the claims.*

Using SBVR Structured English:

Legal Concepts: *Noun concepts defined in green and individual noun concepts are defined in dark-green starting with capital letters.*

claim	
Definition	Define the invention and are what aspects are legally enforceable
Dictionary basis	patentlaw
Source	USPTOGlossary
General Concept	patent

building on the same lines, we obtain other legal concepts like:
examiner office_action paragraph statement argument date drawing
applicant effective_feature invention
essential_subject_matter_requirement

Paragraph_7_33_01

Legal Facts: *Verb concepts are defined in blue.*
office_action *includes* paragraph
claim *is_rejected_under* essential_subject_matter_requirement
office_action *include* statement
applicant *conceals* effective_feature
effective_feature *is_about* the invention
examiner *applies* **Paragraph_7_33_01**
examiner *rejects* the claim
examiner *rejects* the claim

Legal (procedural) rules: *for ¶ 7.33.01.*

1. It is obligatory that examiner *rejects* the claim and office_action *includes* paragraphs **Paragraph_7_33_01** if claim *is_rejected_under* essential_subject_matter_requirement.
2. It is obligatory that office_action *include* statement and argument and date and drawing if office_action *includes* paragraph **Paragraph_7_33_01**.
3. It is obligatory that examiner *applies* **Paragraph_7_33_01** if applicant *conceals* effective_feature and effective_feature *is_about* the invention.

6 Semi-automated Vocabulary Building

In [4], we introduced a proof-of-concept implementation of the tool KR4IPLaw (Knowledge Representation for Intellectual property law). The long term goal of this tool is to provide a user interface, which can be easily handled by legal practitioners and be capable enough to provide all the necessary inputs for a knowledge engineer to model legal rules for (semi-/)automated reasoning thereafter. In this paper, we contribute with an additional conceptual functionality in the architecture of KR4IPLaw, which is provided by a terminology recommender system. Such a system complementary to KR4IPLaw helps to fill the gap between a legal vocabulary/rules built by legal practitioner and all possible concepts/rules which can be identified by the automated system. We strongly believe, that in the legal domain, as an effect of the pragmatics involved, it is rarely possible for a system to fully automate the entire process of building legal rules/vocabularies accurately. Human intervention in confirming the system's automatically generated results is needed in an iterative process during the whole knowledge engineering and formalization process. The recommender system proposed here should provide the required additional context information that can be derived out of the legal context in which a legal vocabulary is built (e.g., case-law, definitions, synonyms etc. pertaining to the section/legal text under consideration). We divide the terminology recommender system into two parts, first one providing legal concepts, i.e., identification of new concepts and semantic enrichment of existing legal concepts. The second one is working on generating the legal facts and building legal rules based on legal facts. In this paper, we mainly concentrate on the first part.

In [4], we already showed how legal practitioners/domain experts either define case based legal vocabularies from scratch or use the preagreed legal vocabulary stored in a central public/privately-shared repositories (such as OntoMaven [13] [14]) and build legal rules based on it as shown before.

For the purpose of legal concept recommendation, we consider a small subset of the available Semantic-Knowledge Extraction (S-KE) tools suitable for its application to our considered legal domain:

– AlchemyAPI: AlchemyAPI [15], is a tool which employs the methods of deep linguistic parsing, statistical natural language processing, and machine learning for named entity extraction, keyword extraction, fact and relation extraction, document categorization, concept tagging and language detection. It builds upon semantic web functionality, AlchemyAPI concepts and entities are linked to DBpedia, Freebase, OpenCyc, GeoNames etc. It is available as a Web application or as a REST service.
– DBpedia Spotlight: A tool for automatically annotating entities in text as DBpedia resources, providing a solution for linking unstructured information sources to the Linked Open Data cloud through DBpedia. It is available as a Web application, as a REST service or as downloadable source [16]. Also language specific versions exist, e.g. DBPedia German[2].
– NERD: NERD [17] proposes a Web framework which unifies numerous named entity extractors using the NERD ontology which provides a rich set of axioms aligning the taxonomies of these tools. Extractors supported by NERD are AlchemyAPI, DBpedia Spotlight, OpenCalais, etc..
– FRED: A tool for automatically producing RDF/OWL ontologies and linked data from natural language sentences. It links the extracted knowledge to both lexical linked data and datasets. It is available as a Web application or as a REST service [18].

Based on [19], a feature based comparison of the considered S-KE tools is as shown in Table 1. Where, NER refers to Named Entity Recognition, DIS refers

[2] http://www.corporate-semantic-web.de/dbpedia-deutsch-spotlight.html

Table 1. Feature based comparison of the semantic-knowledge extraction tools for legal concept recommendation (adapted from [19])

	Topic	NER	DIS	TAX	REL	SemRole	Events	Frames
AlchemyAPI	Yes	Yes	Yes	No	Yes	No	No	No
DBpedia Spotlight	No	Yes	Yes	Yes	No	No	No	No
NERD	No	Yes	Yes	No	No	No	No	No
FRED	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes

to the sense disambiguation feature, TAX refers to Taxonomy identification capability and SemRole refers to the identification of semantic roles against an extracted concept.

From Table 1, we can see that FRED offers more features than its considered counterparts. Based on performance evaluation of the S-KE tools as shown in [19], FRED at the time of review provides better results than DBpedia, AlchemyAPI and NERD. For our proof of concept implementation and evaluation we make use of FRED, and adapt it to the legal domain so that it can be used as a legal concept recommender system, working in conjunction with the existing KR4IPLaw tool. FRED considers a legal sections/text given in its base (English) language as an input to produce semantic data and ontologies with a quality closer to what is expected at-least from average linked data-sets and vocabularies by passing through DRS produced by Boxer. It includes Named Entity Resolution (based on Apache Stanbol) and Word Sense Disambiguation

FRED offers several functionalities [20] as required by any legal recommender system. Some functionalities supported by it are as stated below.

- Captures accurate semantic structures,
- Represents complex relations,
- Supports integration of sophisticated lexical reasoners (like OpenNLP, VerbNet, FrameNet)
- Supports open information extraction,
- Maps natural language to RDF/OWL and
- Links the extracted knowledge to both lexical linked data and datasets (WordNet, DB-pedia and other foundational ontologies)

Fig 4, shows a snippet of the output for a legal text out of the paragraph ¶ 7.30.01.

With the legal text provided as an input, the next step requires the extraction of the required semantic information out of the obtained RDF/OWL ontology. The extracted information is thereafter used to enrich the existing legal vocabulary. The required information is extracted using SPARQL queries and then mapped to legal vocabulary with the help of a mapping scheme as proposed in Table 2.

As a part of the evaluation, we adapt the performance evaluation of NLP tools, proposed by Hirschman and Thompson [21] and its derived methodology proposed in [22]. We assume that a legal practitioner builds a legal vocabulary

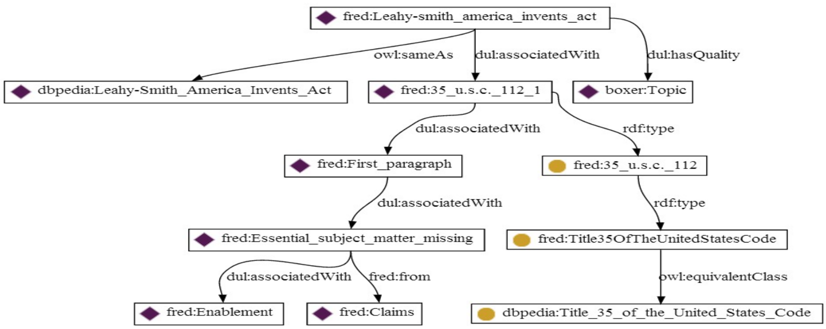


Fig. 4. FRED's output for a legal text

Table 2. Mapping scheme: Parsed legal text to legal (SBVR) vocabulary

RDF class	⟼	Legal (Noun) Concept
rdfs:subClassOf	⟼	General Concept
owl:sameAs	⟼	Synonyms
owl:equivalentClass	⟼	Synonyms
wn:lang	⟼	Language
wn:gloss	⟼	Legal Concept Definition
⟨ boxer⟩: hasModality	⟼	Necessity/Possibility
boxerpos/pennpos: 'v'/'VB'	⟼	Legal (verb) concept
boxerpos/pennpos: 'n'/'NN'/'NNS'	⟼	Legal (Noun) Concept
boxerpos/pennpos: 'np'/'NNP'/'NNPS'	⟼	Legal (Individual Noun) Concept

from scratch to suit, e.g., case-law requirements as an alternative to use/build the existing (pre-agreed) shared vocabulary.

Figure 5, shows a Venn diagram depicting different terms (and its relations) used in this methodology. Building legal arguments (based on legal rules) being the main concern in this evaluation study, a legal practitioner is only interested in the concepts required to build legal rules and rule-based arguments (i.e., $N_{legal(Noun)}$ concepts, $N_{Legal(verb)}$ concept, and $N_{legal(Indv)}$concepts). In this

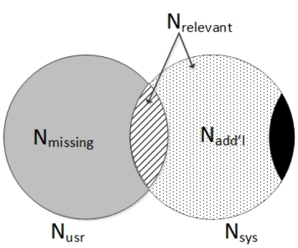


Fig. 5. Venn diagram

evaluation we study to which extent the system is capable of filling/enriching the semantic information attached to each legal concept.

N_{usr} here denotes the inputs from the user in building the legal vocabulary, (where N denotes the number of respective items added). N_{sys} denotes the systems effort in identifying the information/items related to this section of legal text under consideration. $N_{relevant}$ refers to the items that are relevant/meaningful out of the identified items by the system (i.e N_{sys}). The relevance of an item is determined by a domain expert. $N_{missing}$ refers to the difference in number between the items that are relevant and the items that were used/identified by the user/legal practitioner. $N_{add'l}$ refers to the additional relevant items identified by the system which are currently not used by the user. To evaluate the efficiency of such systems, we consider two parameters, $Eff_{sys\ vs.\ relevant}$ and $Eff_{relevant\ vs.\ add'l}$ as shown below:

$$Eff_{sys\ vs.\ relevant} = \frac{N_{relevant}}{N_{sys}} \times 100\% \Rightarrow SystemReliability \quad (1)$$

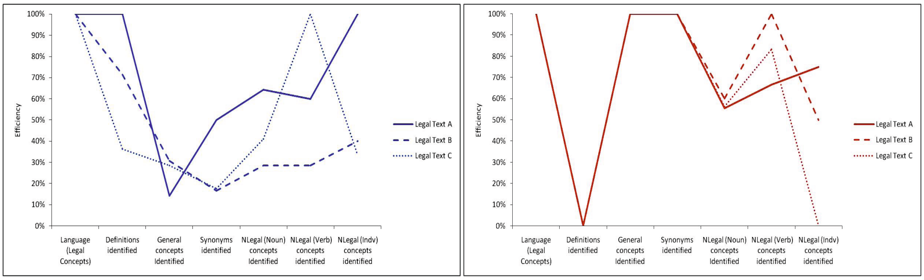
$$Eff_{relevant\ vs.\ add'l} = \frac{N_{add'l}}{N_{relevant}} \times 100\% \Rightarrow SystemIntelligence \quad (2)$$

wherein;
$Eff_{sys\ vs.\ relevant}$ denotes the efficiency of the system in identifying relevant/meaningful items in a given legal passage/text and $Eff_{relevant\ vs.\ add'l}$ refers to the efficiency of the system in providing additional information out of its identified relevant items. We consider the example shown in the last section as an input to the recommender system. Table 3 shows a chart comprising of both inputs from the user as well as from the system. The efficiency of the system is as shown in Figure 6 (i.e. Legal Text A).

Table 3. Recommender system outcome analysis

	N_{usr}	N_{sys}	$N_{relevant}$	$N_{missing}$	$N_{add'l}$
Language (Legal concepts)	0	1	1	0	1
Definitions identified	0	4	4	8	0
General concepts identified	0	14	2	NA	2
Synonyms identified	0	4	2	NA	2
$N_{Legal(Noun)}$concepts identified	12	14	9	8	5
$N_{Legal(Verb)}$concepts identified	6	5	3	5	2
$N_{Legal(Indv)}$concepts identified	1	4	4	0	3

Figure 6 gives the results of the evaluation on two additional legal paragraphs (denoted here as legal texts 'B' and 'C'). Specifically, Fig 6a, shows the efficiency $Eff_{sys\ vs.\ relevant}$ and Fig 6b, shows the efficiency $Eff_{relevant\ vs.\ add'l}$. The second part of the recommender system involving (semi/-)automatized building of legal rules is still an open research question. There have been several works in automatic extraction of SBVR business rules [23] [22] [24] [25]. Adapting it to



(a) $Eff_{sys\ v.\ relevant}$ (b) $Eff_{relevant\ v.\ add'l}$

Fig. 6. Efficiency evaluation

legal domain has shown high inconsistencies between the actual legal texts to its constructed legal rules. The architecture of FRED is designed to allow the use of domain specific legal lexical resources, which includes the knowledge base (legal vocabulary) built during the semi-formal representation of the procedural legal rules.

7 CIM to PIM to PSM

Moving from computational independent layer (i.e. SBVR-SE) to platform independent layer requires storing the semantically enriched legal vocabulary and rules in a machine oriented format. Legal vocabularies (Legal Concepts and Legal facts) are mapped into an OWL2 ontology. In [26] [27] [28], authors have proposed a possible mapping scheme for such transformations. For interchanging the legal rules in a platform independent way, we propose to translate them into XML using the language family of 'RuleML' [29] as expression language. In particular, we make use of two complementary OASIS standards-OASIS Legal Document Markup Language, 'LegalDocML' [30] and OASIS Legal Rule Markup Language, 'LegalRuleML' [31] [32] in combination with Reaction RuleML [33] [34] for the said transformation. The details of this semantic transformation process are out of scope of this paper. They can be found in [3].

For reasoning with such transformed legal rules using legal knowledge bases, we use Prova [35] [36] [37] [38], as a rule engine. Prova is both a Semantic Web rule language and a high expressive distributed rule engine. It, supports the execution of declarative (legal) rules including scoped reasoning [37] [36] [38], Rule-Based Data Access (RBDA) to external semantic web data via SPARQL, and Ontology-Based Data Access (OBDA) with DL typed reasoning [39]. For the purpose of ontology reasoning on-top of legal knowledge bases (domain ontologies), Prova integrates SPARQL-DL API [40], a subset of SPARQL tailored for ontology-specific requests related to OWL and it is more expressive than existing DL query languages by allowing a mix of TBox, RBox, and ABox queries. It can be regarded as an OBDA interface to any ontology reasoner supporting OWL-API. Reasoning with legal rules in Prova is also out of the scope of this paper.

For examples on representing LegalRuleML in Prova we refer to the patent law use case[3] [41] and the copyright use case[4] of the LegalRuleML tutorials.

8 Conclusion and Future Directions

The paper in its first part explored the use of controlled natural languages as a bridge between a domain expert and a knowledge modeler in legal domain. We then showed with the help of an example on how SBVR Structured English, a controlled natural language, can be used in the legal domain (specifically for IP law representation). In the second part of the paper, we presented an extension of our KR4IPLaw system with a legal concept recommender system which supports the manual vocabulary building process by making automated suggestions. We implemented a proof-of-concept and studied the feasibility of the automation approach of semantically enriching legal vocabularies by means of case study examples. During the course of this studies, we identified some new and re-iterated some known research problems existing in the process of automation in legal domain. The long term goal of this KR4IPLaw project is to build a system which acts as a platform to model, represent, recommend, and reason about legal patent law knowledge.

Acknowledgements. The authors would like to thank the entire Corporate Semantic Web team at the Free University of Berlin for their constructive comments and suggestions. The authors would also like to thank Mr Spurthishekar for assisting in sentence structure and grammatical error corrections.

References

1. Palmirani, M., Contissa, G., Rubino, R.: Fill the gap in the legal knowledge modelling. In: Governatori, G., Hall, J., Paschke, A. (eds.) RuleML 2009. LNCS, vol. 5858, pp. 305–314. Springer, Heidelberg (2009)
2. OMG: Semantics of Business Vocabulary and Business Rules (SBVR)- Version 1.2. Technical Report November, Object Management Group (2013)
3. Ramakrishna, S., Paschke, A.: A Process for Knowledge Transformation and Knowledge Representation of Patent Law. In: RuleML 2014. LNCS, vol. 8620, pp. 308–325. Springer, Heidelberg (2014)
4. Ramakrishna, S., Paschke, A.: Bridging the gap between Legal Practitioners and Knowledge Engineers using semi-formal KR. In: The 8th International Workshop on Value Modeling and Business Ontology, VMBO, Berlin (2014)

[3] http://2013.ruleml.org/presentations/RuleML2013Tutorial_PaschkePatentLaw.pdf

[4] https://lists.oasis-open.org/archives/legalruleml/201208/msg00040/LegalRuleML-palmirani2012_-RuleML2012v3.pdf, slide 39-54.

5. Fuchs, N.E., Schwitter, R.: Attempto controlled english (ace). arXiv preprint cmp-lg/9603003 (1996)
6. Paris, C., Linden, K.V.: DRAFTER: An Interactive Support Tool for Writing Multilingual Instructions. IEEE Computer, Special Issue on Interactive NLP (July 1996)
7. Sullivan, D.E.: Legislative drafting and legal manual (2003)
8. Kuhn, T.: Controlled Natural Language and Opportunities for Standardization. In: International Workshop on Terminology, Languages, and Content Resources (June 2013)
9. Bézivin, J., Gerbé, O.: Towards a precise definition of the OMG/MDA framework. In: Proceedings of the 16th Annual International Conference on Automated Software Engineering, ASE 2001, pp. 273–280. IEEE (2001)
10. Johnsen, A.S.: Semantisk modellering av juridisk regelverk med bruk av SBVR - en brobygger mellom jus og IT. Master thesis, University of Oslo (2011)
11. Johnsen, A.S., Berre, A.J.R.: A bridge between legislator and technologist - Formalization in SBVR for improved quality and understanding of legal rules. In: International Workshop on Business Models, Business Rules and Ontologies, Bressanone, Brixen, Italy (2010)
12. USC: Title 35 of the United States Code (1952)
13. Paschke, A.: OntoMaven API4KB - A Maven-based API for Knowledge Bases. In: 6th International Semantic Web Applications and Tools for the Life Science (SWAT4LS 2013), Edinburgh, UK, December 10-12 (2013)
14. Paschke, A.: OntoMaven. In: 9th International Workshop on Semantic Web Enabled Software Engineering (SWESE 2013), Berlin, Germany, December 2-5 (2013)
15. Turian, J.: Using AlchemyAPI for Enterprise-Grade Text Analysis. Technical report, AlchemyAPI (August 2013)
16. Mendes, P.N., Jakob, M., García-Silva, A., Bizer, C.: Dbpedia spotlight: Shedding light on the web of documents. In: Proceedings of the 7th International Conference on Semantic Systems. I-Semantics 2011, pp. 1–8. ACM, New York (2011)
17. Van Erp, M., Rizzo, G., Troncy, R.: Learning with the web: Spotting named entities on the intersection of nerd and machine learning. In: Proceedings of the 3rd Workshop on Making Sense of Microposts (# MSM 2013) (2013)
18. Draicchio, F., Gangemi, A., Presutti, V., Nuzzolese, A.G.: FRED: From Natural Language Text to RDF and OWL in One Click. In: Cimiano, P., Fernández, M., Lopez, V., Schlobach, S., Völker, J. (eds.) ESWC 2013. LNCS, vol. 7955, pp. 263–267. Springer, Heidelberg (2013)
19. Gangemi, A.: A comparison of knowledge extraction tools for the semantic web. In: Cimiano, P., Corcho, O., Presutti, V., Hollink, L., Rudolph, S. (eds.) ESWC 2013. LNCS, vol. 7882, pp. 351–366. Springer, Heidelberg (2013)
20. Gangemi, A., Draicchio, F., Presutti, V., Nuzzolese, A.G., Recupero, D.R.: A machine reader for the semantic web. In: International Semantic Web Conference (Posters & Demos), pp. 149–152 (2013)
21. Hirschman, L., Thompson, H.S.: Chapter 13: Overview of Evaluation in Speech and Natural Language Processing. In: Survey of the State of the Art in Human Language Technology, pp. 385–420 (1995)
22. Afreen, H., Bajwa, I.S.: Generating UML Class Models from SBVR Software Requirements Specifications. In: 23rd Benelux Conference on Artificial Intelligence (BNAIC 2011), Gent, Belgium, pp. 23–32 (2011)

23. Bajwa, I., Lee, M., Bordbar, B.: SBVR Business Rules Generation from Natural Language Specification. In: AAAI 2011 Spring Symposium AI for Business Agility, San Francisco, USA, pp. 2–8 (2011)
24. Martínez-Fernández, J.L., González, J.C., Villena, J., Martínez, P.: A preliminary approach to the automatic extraction of business rules from unrestricted text in the banking industry. In: Kapetanios, E., Sugumaran, V., Spiliopoulou, M. (eds.) NLDB 2008. LNCS, vol. 5039, pp. 299–310. Springer, Heidelberg (2008)
25. Chaparro, O., Aponte, J., Ortega, F., Marcus, A.: Towards the Automatic Extraction of Structural Business Rules from Legacy Databases. In: 2012 19th Working Conference on Reverse Engineering (WCRE), pp. 479–488 (October 2012)
26. Elisa, K., Mark, H.L.: Mapping SBVR to OWL2. Technical report, IBM Research Division, New York, NY (2013)
27. Reynares, E., Caliusco, M.A., Galli, M.R.: Automatable Approach for SBVR to OWL2 Mappings. In: XVI Ibero-American Conference on Software Engineering (CIbSE 2013), Montevideo, Uruguay (2013)
28. Karpovic, J., Nemuraite, L.: Transforming SBVR Business Semantics into Web Ontology Language OWL2: Main Concepts. In: In Proc. 17th International Conference on Information and Software Technologies, IT 2011, pp. 231–254 (2011)
29. Boley, H., Paschke, A., Shafiq, O.: RuleML 1.0: The Overarching Specification of Web Rules. In: Dean, M., Hall, J., Rotolo, A., Tabet, S. (eds.) RuleML 2010. LNCS, vol. 6403, pp. 162–178. Springer, Heidelberg (2010)
30. Gordon, T.F.: The Legal Knowledge Interchange Format (LKIF). Technical report, European project for Standardized Transparent Representations in order to Extend LegaL Accessibility Specific Targeted Research or Innovation Project (2008)
31. Palmirani, M., Governatori, G., Rotolo, A., Tabet, S., Boley, H., Paschke, A.: LegalRuleML: XML-Based Rules and Norms. In: Palmirani, M. (ed.) RuleML 2011 - America. LNCS, vol. 7018, pp. 298–312. Springer, Heidelberg (2011)
32. Athan, T., Boley, H., Governatori, G., Palmirani, M., Paschke, A., Wyner, A.: OASIS LegalRuleML. In: Proceedings of 14th International Conference on Artificial Intelligence and Law (ICAIL 2013). ACM (2013)
33. Paschke, A., Boley, H., Zhao, Z., Teymourian, K., Athan, T.: Reaction RuleML 1.0: Standardized Semantic Reaction Rules. In: Bikakis, A., Giurca, A. (eds.) RuleML 2012. LNCS, vol. 7438, pp. 100–119. Springer, Heidelberg (2012)
34. Paschke, A.: Reaction RuleML 1.0 for Rules, Events and Actions in Semantic Complex Event Processing. In: RuleML 2014. LNCS, vol. 8620, pp. 1–18. Springer, Heidelberg (2014)
35. Kozlenkov, A., Paschke, A.: Prova Rule Language Version 3.0 User's Guide (2010), http://prova.ws/index.html
36. Paschke, A.: Rules and logic programming for the web. In: Polleres, A., d'Amato, C., Arenas, M., Handschuh, S., Kroner, P., Ossowski, S., Patel-Schneider, P. (eds.) Reasoning Web 2011. LNCS, vol. 6848, pp. 326–381. Springer, Heidelberg (2011)
37. Paschke, A., Boley, H.: Rule Responder: Rule-Based Agents for the Semantic-Pragmatic Web. International Journal on Artificial Intelligence Tools 20(6), 1043–1081 (2011)
38. Paschke, A.: Rule based service level agreements: RBSLA; knowledge representation for automated e-contract, SLA and policy management. Idea Verlag GmbH (2007)

39. Paschke, A.: A Typed Hybrid Description Logic Programming Language with Polymorphic Order-Sorted DL-Typed Unification for Semantic Web Type Systems. In: Proceedings of the OWLED 2006 Workshop on OWL: Experiences and Directions, Athens, Georgia, USA, November 10-11. CEUR Workshop Proceedings, vol. 216, CEUR-WS.org (2006)
40. Sirin, E., Parsia, B.: SPARQL-DL: SPARQL Query for OWL-DL. In: 3rd OWL Experiences and Directions Workshop, OWLED 2007 (2007)
41. Paschke, A., Ramakrishna, S.: Legal RuleML Tutorial Use Case - LegalRuleML for Legal Reasoning in Patent Law (2013)