



# A Semi-automated Ontology Construction for Legal Question Answering

Biralatei Fawei<sup>1</sup> · Jeff Z. Pan<sup>1</sup> · Martin Kollingbaum<sup>1</sup> · Adam Z. Wyner<sup>2</sup>

Received: 18 February 2019 / Accepted: 11 August 2019 / Published online: 6 September 2019  
© The Author(s) 2019

## Abstract

The internet and the development of the semantic web have created the opportunity to provide structured legal data on the web. However, most legal information is in text. It is difficult to automatically determine the right natural language answer about the law to a given natural language question. One approach is to develop systems of legal ontologies and rules. Our example ontology represents semantic information about USA criminal law and procedure as well as the applicable legal rules. The purpose of the ontology is to provide reasoning support to a legal question answering tool that determines entailment between a pair of texts, one known as the background information (Bg) and the other question statement (Q), so whether Bg entails Q based on the application of the legal rules. The key contribution of this paper is the methodology and the semi-automated legal ontology generation tool, a clear and well-structured methodology that serves to develop such criminal law ontologies and rules (CLOR).

**Keywords** Ontology · Legal rules · Bar examination

## Introduction

Legal knowledge is usually expressed with domain-specific terminology and conveyed in textual form. Moreover, its expression and presentation do not provide a standard structure for a machine to use and reason with. The extraction and mining of terms and concepts of the domain along with relations and rules amongst them will lead to some knowledge representation model. Ontologies and the semantic web have created a means of representing domain-specific conceptual knowledge which can be used to facilitate the semantic capabilities for

---

✉ Biralatei Fawei  
r01bjf15@abdn.ac.uk

<sup>1</sup> Department of Computing Science, University of Aberdeen, Aberdeen, UK

<sup>2</sup> School of Law and Department of Computer Science, Swansea University, Swansea, UK

question answering systems. As such, ontologies are used to capture relevant knowledge in a standard way and provide a more generally accepted understanding of a field of study. The general vocabulary of an ontology defines the terms and relations which are usually organized in a hierarchical structure. However, capturing human created semantic information from the text for machine processing is not a linear process. The manual method of extracting and classifying legal text according to classes and object properties involves reading textual documents from different sources that deals with a given legal topic area.

To apply the law to a given legal case, a human legal expert needs to have the right expert knowledge of legal concepts to make a judicial decision. Our ultimate aim is to automate this process of judging a legal case. To allow legal reasoning, we take a legal ontology to be a core element in this process as it links all the necessary legal elements of such a case and supports an automated legal decision process. In this paper, we present an ontology engineering methodology along with a semi-automated approach of legal ontology generation from a collection of legal documents alongside with legal rules to provide reasoning support to an automatic legal question answering system. The main contribution of this paper is the methodology and the semi-automation of the 15 steps out of the 18 step legal ontology construction model for criminal law ontology. The tool uses Stanford parser to preprocess the input text and produce semantic triples. However, the semi-automated tool involves human intervention in the creation of a resource. The resources are then used in the automated process.

The research takes the perspective of a textual entailment task to question answering as it is used in the US Bar exam [1], our benchmark of choice. More formally, we can state that given a theory text  $T$  and hypothesis text  $H$ , we can determine whether or not from  $T$  one can infer  $H$  [2, 3]. The original bar exam questions are organized in the form of background information (Bg), which is the theory  $T$ , and multiple-choice statements (Q), each of which we take as an hypothesis  $H$ . The objective is to select the correct  $H$ , given  $T$ . That is, given the background information, one must accept one and reject the other three multiple-choice question statement based on the application of the law. For example, one of the bar exam questions looks like the following (with the options for an answer listed subsequently):

7. After being fired from his job, Mel drank almost a quart of vodka and decided to ride the bus home. While on the bus, he saw a briefcase he mistakenly thought was his own, and began struggling with the passenger carrying the briefcase. Mel knocked the passenger to the floor, took the briefcase, and fled. Mel was arrested and charged with robbery.

*Mel should be:*

- a. acquitted, because he used no threats and was intoxicated.
- b. acquitted, because his mistake negated the required specific intent.
- c. convicted, because his intoxication was voluntary.
- d. convicted, because mistake is no defense to robbery.

**Table 1** Sample bar examination criminal law question (adapted)

Bg After being fired from his job, Mel drank almost a quart of vodka and decided to ride the bus home. While on the bus, he saw a briefcase he mistakenly thought was his own, and began struggling with the passenger carrying the briefcase. Mel knocked the passenger to the floor, took the briefcase, and fled. Mel was arrested and charged with robbery. The mistake of Mel negated the required specific intent.

Q Mel should be acquitted.

Note, however, that to make the source text into something suitable for the textual entailment task, we have revised the multiple-choice questions into Bg and individual Q pairs, where any background information in the source Q is put into the Bg. For example, as shown in Table 1, we took option b, made a proposition ‘Mel should be acquitted’ as Q, and introduced ‘his mistake negated required specific intent’ as part of the background [4].

A range of approaches can be applied to the textual entailment task, e.g. machine learning, lexical and syntactic information and semantic dependencies.

However, these techniques lack the sort of legal knowledge and reasoning required to determine entailment in the text representing bar examination questions. The novelty of the research is the criminal law and procedure ontology construction procedure [5] and its semi-automated implementation alongside with legal rules to reason towards a conclusion. In designing the ontology and rules, the legal knowledge is modelled from the US bar examination preparatory materials: bar exam curriculum, course material, Strategic and Tactics for the multistate bar exam (MBE) [6] and other legal textbooks [7, 8] as well as some practical knowledge from legal practitioners. In addition, we adapted relevant commonsense knowledge related to or useful for our task from existing common-sense knowledge bases [9, 10] since some of the example problems, e.g. the above, incorporate commonsense knowledge. The research progresses work on the significant and problematic link between legal and common-sense knowledge, wherein a legal concept such as specific intent is realized with respect to common-sense actions.

An ontology and rule set for criminal law and procedure is large, complex, and evolving. Our contribution develops an interesting and relevant fragment, which can be developed further. In addition, an important research contribution is our incremental methodology for the criminal legal ontology and rule (CLOR) development, wherein we start with some initial ontology and rules and build upon them to account for further bar examination questions. The idea is that within this process, we come to identify specific or repeated patterns of legal reasoning, which then lead towards further generalization and application of legal rules.

This is demonstrated later, where an initial system is developed on the basis of a limited set of data and then applied to further examples which had not be considered in the initial system.

The current paper implements the ontology construction methodology defined in [5], which provided an initial criminal law ontology along with SWRL rules to draw inferences as well as provided preliminary results from an initial experiment. In addition, the approach used NLP techniques to extract textual information

from the source text. However, in that paper the ontology and rules were manually implemented.

The rest of the paper is organized as follows. “[Related work](#)” discusses legal ontologies and closely related works. “[Problem statement](#)” describes the problem statement. The methodology applied in constructing the ontology is explained in “[Methodology](#)”, and the implementation of the methodology is discussed in “[Implementation of legal ontology generation tool](#)”. “[Legal rule acquisition and representation](#)” presents legal rules acquisition and presentation; an illustration of how the rules are applied to ontological information is in “[Application of the ontology](#)”. “[Ontology evaluation](#)” outlines how the ontology was evaluated. We conclude with some discussion in “[Conclusion](#)”.

## Related Work

With the widespread adoption of ontologies for different applications, several ontology development techniques have been proposed and applied. Previous works have constructed ontologies in various domain. El Ghosh et al. [11] presented a semi-automatic ontology construction technique. The approach combines the top-down and bottom-up strategy of ontology construction. The top-down strategy models the upper or core module of the ontology which comprises of the conceptual structure of the criminal domain. This part reuses existing ontologies and extracts similar and complementary information, while the bottom-up strategy captures relevant legal concepts and relations from textual sources using some NLP techniques. Hwang et al. [12] describe a technique for an automatic ontology construction from a structured text (databases). The approach captures legal concepts and relations from the Chinese Laws and Regulations Database and then constructs a law ontology. For concept and relationship extraction, NLP and data mining techniques are implemented. The essence is to extract legal keywords automatically with the respective definitions. The extracted keywords and relations extracted are used to build the law ontology. Deng and Wang [13] develop an ontology for maritime information in Chinese. For relevant concepts and relations extraction, a weight calculator is applied to calculate the weight of each term with respect to the maritime domain. The essence is to identify words with a higher weight as suitable classes in the construction of the maritime domain ontology. Johnson et al. [14] presented a law enforcement ontology construction, from a collection of thousands of sanitized emails. These emails are gathered from law enforcement investigators throughout 3 years.

Osathitporn et al. [15] describe an ontology for Thai criminal legal code with concepts about crime, justification, and criminal impunity. It aims to help users to understand and interpret the legal elements of criminal law. However, the focus of the ontology as well as its structural and hierarchical organization differs from an ontology for legal question answering. Bak et al. [16] describe an ontology as well as rules that capture and represent the relationship existing between legal actors and their different roles in money laundering crime. It includes relational information about companies, entities, people, and actions. Ceci and Gangemi [17] present an

OWL2-DL ontology library that describes the interpretation a judge makes of the law in providing a judgment while engaged in a legal reasoning process to adjudicate a case. This approach is based on a theoretical model and some specific patterns that use some newly introduced features of OWL2. This approach delivers meaningful legal semantics while the link to the source document is strongly maintained (that is, fragments of the legal texts). Gangemi et al. [18] describe how new legal decision support systems can be created by exploiting existing legal ontologies. Legal ontology design patterns were proposed in [19], wherein they applied conceptual ontology design patterns (CODEPs). However, this work differs from legal question answering in which legal rules need to be applied to facts extracted from legal text to reason with to determine an answer.

Several ontology development methodologies have been proposed. However, these different methodologies have not delivered a complete ontology development standard as in software engineering. Suarez-Figueroa et al. [20] present the NeOn ontology development methodology. NeOn is a scenario-based approach that applies a different insight into existing ontology construction methodologies. However, this approach does not specify a particular workflow for the ontology development, rather it recognizes nine scenarios for collaborative ontology construction, re-engineering, alignment, and so on. De Nicola and Missikoff [21] proposed the Unified Process for ONtology (UPON Lite), an ontology construction methodology that depends on an incremental process to enhance the role of end users without requiring any specific ontology expertise at the heart of the process. The approach is established with an ordered set of six steps. Each step displays a complete and independent artefact that is immediately available to end users, which serves as an input to the subsequent step. This whole process reduces the role of ontology engineers.

An overview of ontology design patterns was presented in [22] exploring how ontologies are constructed in the legal domain. Current approaches on ontology development can be categorized as either “top-down” or “bottom-up”. The manual development of ontologies from scratch by a knowledge engineer and with the support of domain experts is known as the top-down approach [23], which is later used to annotate existing documents. When an ontology is extracted by automatic mappings or extraction rules or by machine learning from vital data sources [23], then this is regarded as a bottom-up approach. Much of the research works on legal data harmonization, applying a standardized formal language to express legal knowledge, its metadata, and its axiomatization. With respect to a top-down approach, Hoekstra et al. [24] present the Legal Knowledge Interchange Format (LKIF), an alternative schema that can be seen as an extension of MetaLex. It is more expressive than OWL and includes LKIF rules that support axiomatization. Related, Athan et al. [25] propose the LegalRuleML language that is an extension of the XML-based markup language known as RuleML. It can be applied for expressing and inferencing over legal knowledge. In addition, Gandon et al. [26] proposed an extension of the LegalRuleML that supports modelling of normative rules. There has not been an instantiation of LKIF and LegalRuleML at scale or used for formalizing or annotating the content of a legal corpora either automatically or manually. Also, different theoretical approaches have argued that laws can be formally defined and reasoned with by applying non-classical logics like defeasible logic or deontic logic, of which

their application involves the manual encoding of some specific parts of a legislative document that may not scale to a full legal corpus [27, 28].

## Problem Statement

While current approaches to textual entailment or question answering contain one or two steps of reasoning in dealing with general knowledge in deciding entailment or answering questions, they lack the sort of legal knowledge and reasoning required for deep textual entailment task as well as the capability to provide the line of logical reasoning that leads to the answer.

In general, given background information and a possible answer in the form of an implicit (Yes/No) question about that information, how can a machine process that information and reason with it to arrive at a true or false answer based on the background information? For example, an example derived from the Bar exam source material is in Table 1. To meet the research objective, we are motivated to develop a semi-automatic criminal law ontology generation tool as well as rules that provide a reasoning support to an automatic legal question answering tool in answering the USA bar examination questions.

## Methodology

In this section, we present our methodology, first with some general points, then with more specific considerations. This methodology consists of 18 steps that lead to the creation of a legal ontology and a corresponding set of rules. Besides, we describe each of the 18 steps with an example using the sample bar examination question in “[Introduction](#)” as the input text.

We selected source material about criminal law and legal procedures from exam preparation material [6, 8, 29, 30], information from domain experts, and twelve randomly selected bar exam questions (questions 7, 15, 61, 66, 76, 98, 101, 102, 103, 107, 115 and 117) from a set of 200 questions [1]. The bar exam questions come with an answer key, which constitutes the benchmark for our methodology.

The selected questions contain criminal law and procedural notions such as acquit, robbery, larceny, felony murder, arson, drug dealing and motion moving in criminal procedure. The idea is to ensure that all the information necessary for applying the law is extracted and represented in the ontology. That means, we systematically analyse the questions to identify and extract concepts, properties and relationships relevant for applying the legal rules for making legal decisions.

Due to the challenging nature of ontology and rule authoring [31, 32], we decompose the analysis into a series of simpler competency questions (CQ) [33, 34], each of which is aimed at collecting some specific information and can be used to ensure quality control of the knowledge base [35, 36]. The domain expert seeks to answer the questions with respect to the corpus of bar exam questions and answers. These questions play a crucial part in the knowledge acquisition phase of the ontology development life cycle, as they describe the requirements

**Table 2** Sample competency question

1.	What are the elements of robbery?
2.	Under what conditions should $x$ be convicted?
3.	Under what conditions should $x$ be acquitted?
4.	What are the differences between robbery and $X$ (such as theft)?
5.	What are the similarities between robbery and $X$ (such as theft)?
6.	What element(s) of robbery is necessary?
7.	What element(s) of robbery is sufficient?
8.	What elements(s) of robbery is optional?
9.	What are the defenses for robbery?
10.	What are the penalties for robbery?
11.	What is the sentencing for robbery?

of the intended ontology (see sample competency questions in Table 2). Next, we created a methodology consisting of 18 steps (see Fig. 1).

Some steps process the text to provide material for further analysis, e.g. Select all nouns. Other steps filter or process information, e.g. Identify relevant nouns (given some notion of relevance) and Identify atomic and definable classes (given some notions of atomic and definable), and yet other steps further select information in response to particular competency questions. Thus, for each step, we process or seek to identify specific information from the bar exam question material and extract it into an ontology.

Steps 1 and 2: We identified and created competency questions relevant for extracting necessary information from the textbooks describing law and procedures [6, 8, 29, 30]. For example, the relevant information for competency questions 1, 2, and 3 above could be retrieved from these textbooks:

1. The elements of robbery are: “property is taken from the person or presence of the owner; and the taking is accomplished with the application of physical force or putting the owner in fear. A threat of harm will suffice” [6, 30].
2. To convict someone, the crime has at least three elements: criminal act (*actus reus*), criminal intent (*mens rea*), and occurrence = act + intent [8].

Using the competency questions and other elements of our methodology, we extract legal concepts from these texts for our ontology.

Step 3: We start by identifying and collecting all the nouns in a particular bar exam question (question 7) without minding their relationships, the overlap between them, the characteristic attributes of the nouns or whether the nouns should be in a class or not. We want to know the elements of a crime which we would like to reason with. For example, the following nouns were identified and collected from the bar exam question text (see Table 1)—*Job, Mel, Quart, Vodka, Bus, Home, Briefcase, Passenger, Floor, Robbery, Threat, Intoxication,*

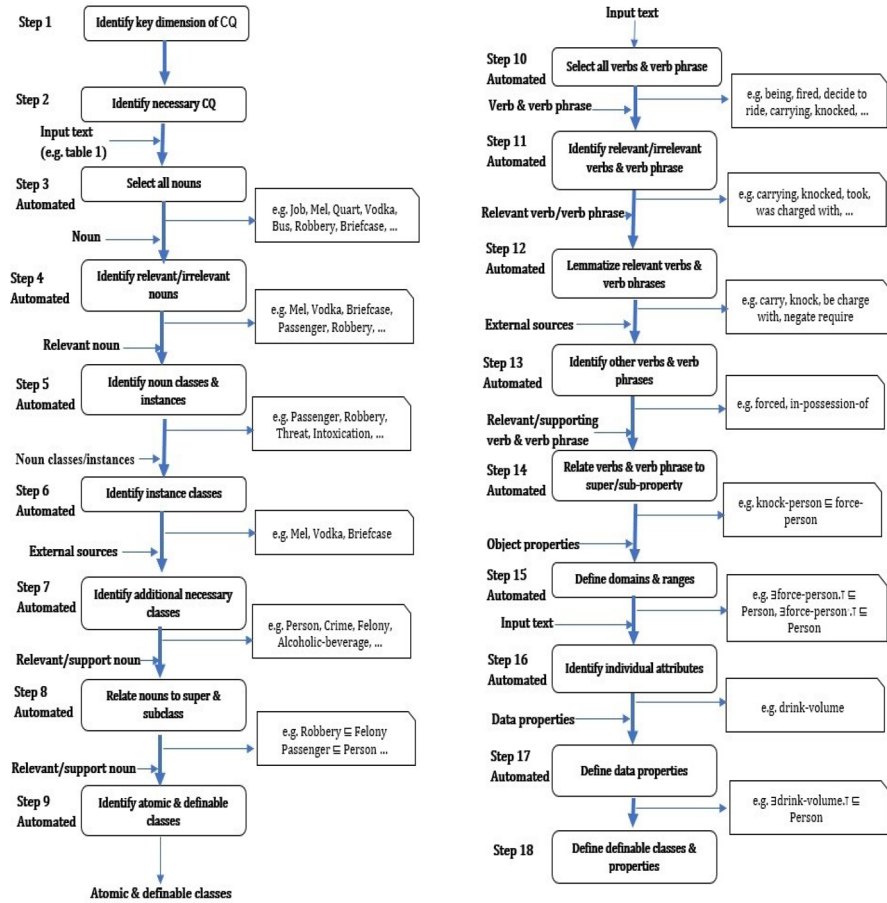


Fig. 1 Ontology design procedure

*Mistake, Defense, Voluntary action, and Intent* are extracted in relation to the elements of robbery and elements of crime as in CQ 1 and CQ 2 above along with some useful legal key terms.

Step 4: We separate the relevant nouns from the irrelevant ones (see Fig. 1). For example, the relevant nouns *Mel, Vodka, Briefcase, Passenger, Robbery, Threat, Intoxication, Mistake, Defense, Voluntary-action, Intent* are extracted in relation to the elements of robbery and elements of crime as in CQ 1 and CQ 2 above, along with some useful legal key terms. Relevant nouns are the nouns that match the elements of crime identified from the competency questions. In our example running text, the competency question is *What are the elements of robbery?* The textbook answer is: *“A person forcibly steals property and commits robbery when, in the course of committing a larceny, he uses or threatens the immediate use of physical*



*force upon another person*".<sup>1</sup> We identify the nouns in this answer, e.g. property and physical force. Moreover, we identify related nouns using terminological relations derived from WordNet to get more specific classes of nouns, e.g. classes related to property are *money*, *personal property*, *intangible property*, and things in action. Some of the classes may have further subclasses, e.g. personal property includes *books*, *cd*, *jewellery*, and so on. Such nouns are the model nouns. Thus, to identify the relevant nouns in our target text, we first extract all the nouns from the text, then we filter them through the model; that is, if an extracted noun is a model noun or a legal keyword, then it is a relevant noun, and any other noun is not relevant. The irrelevant ones *Job*, *Bus*, *Home*, *Floor* may be relevant to other crimes, but are not relevant to reason with in this particular robbery crime (question 7). Once we are able to identify all the relevant concepts, we can then apply them for legal reasoning while discarding the irrelevant ones.

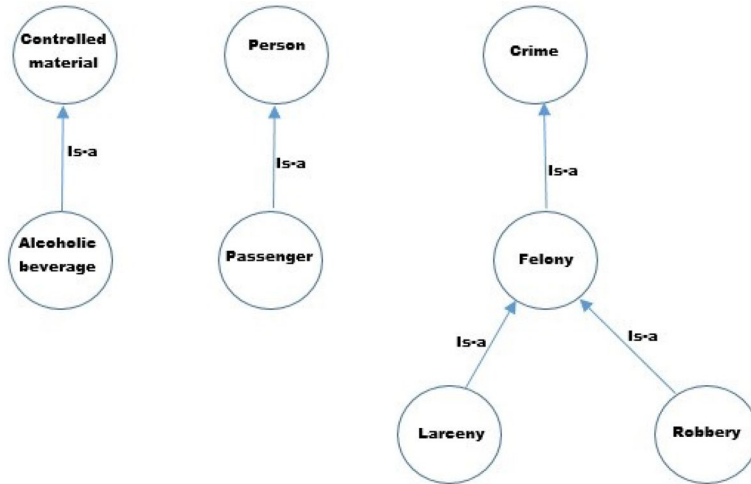
Step 5: After identifying the relevant and irrelevant nouns, from the relevant ones we determine the type of nouns which we could describe as classes and instances (see Fig. 1). We identified the nouns *Passenger*, *Robbery*, *Threat*, *Intoxication*, *Mistake*, *Defense*, *Voluntary-action*, and *Intent* as classes, whereas *Mel*, *Vodka* and *Briefcase* are ground level objects, which are instances of a class.

Step 6: Here, we identify the classes of the objects *Mel*, *Briefcase* and *Vodka* as *Person*, *Property* and *Alcoholic-beverage*, respectively. *Robbery* is described as forcible stealing [30]. It means a person taking something of value from another person by applying force, threat or by putting the person in fear. From our text, *Mel* forcefully collected the briefcase from the passenger who was in possession of the briefcase by knocking the passenger down on the floor. As such, we extract *Mel* as the person and *briefcase* as the valuable thing or property. Likewise, *vodka* is a fermented liquor that contains ethyl alcohol which corresponds to the concept of *Alcoholic beverage*.

Step 7: While creating the class hierarchy, it is necessary to identify other classes, which are not in the selected bar exam questions, but are needed to create clear class hierarchies (see Fig. 1). For example, classes such as *Person*, *Alcoholic-beverage*, *Crime*, *Felony*, and *Controlled-material* are created as conceptual "covers" of the particular terms in our examples. More generally, the task is to classify a set of named entities in the texts as persons, organizations, locations, quantities, times, and so on. Here, *Mel* is a name of a person and, therefore, a *Person* concept. *Alcoholic beverages* like liquor are *controlled materials*; therefore, we create the *controlled-material* class as a superclass of the *alcoholic-beverage* class, and define *vodka* as an instance of this class.

Step 8: In creating the class hierarchy, the class *Robbery* is a subclass of *Felony* (Rv F) and *Felony* a subclass of *Crime* (Fv C) (see Fig. 2). Furthermore, *has-committed-robbery* (HCRvJ), *should be acquitted* (SBAvJ), and *should-be-convicted* (SBCvJ) are subclasses of the *Judgement* class, and *Alcoholic-beverage* is a subclass of *Controlled-material* (ABvCM).

<sup>1</sup> <http://ypdcrime.com/penal.law/article160.htm>.



**Fig. 2** Fragments of our legal ontology

Step 9: The above identified concepts are classified into atomic and defined classes. Atomic classes have no definitions and are used types of instances. These are self-explanatory concepts and cannot be derived using other classes or properties. For example, *Mel* is a person and so *Mel* is a member of the *Person* class. Definable classes can be defined by using other classes and properties. For example, an *Offense* is defined as consisting of both *a guilty act* and *a guilty mind*. Often definable classes do not have direct instances; instead, objects can be classified as their instances by reasoning. Here, the definable classes are *has-committed-robbery*, *should-be-acquitted*, and *should-be-convicted*. To define the definable classes, we need to use properties (cf. the next steps).

Step 10: For object property identification, we start by identifying and extracting all the main verbs in the text (see Fig. 1). We do not consider verb phrases—a verb together with objects. Such objects are related to subjects in the ontology as below. For example, from the text we identify *being fired*, *decide to ride*, *carrying*, *knocked*, *took*, *was charged with*, and *negated required*.

Step 11: Amongst the extracted verbs, we determine the relevant ones by identifying the ones that link the identified nouns together in our earlier concept identification phase. The ones that do not link the selected concepts are the irrelevant ones. Here, in relation to our example text in Table 1 and the element of robbery and crime as in CQ 1 and CQ 2 above, the following verbs are relevant: *carrying*, *knocked*, *took*, *was charged with*, and *negated required*. They link together the concepts identified earlier. These relations are helpful in defining the elements of robbery and crime in which criminal law and procedural rules can be applied. Furthermore, verb phrases such as “being fired”, “decide to ride” and some others are irrelevant since they do not link the extracted concepts together.

Step 12: The relevant verbs extracted are lemmatized to eliminate inflectional forms except the “to be” verbs. For example, from the extracted relations, we have

*carry, knock, take, be charge with, and negate require*. We keep compound verbs, which are those together with selected prepositions.

Step 13: Other verbs that may be useful and necessary for linking some of the relevant concepts are identified to answer our competency question, for example, *forced* and *in-possession-of*.

Step 14: The retrieved verbs are then related into super and sub-property relations, thereby creating the object-property hierarchy. It is important to point out that due to the peculiarity of legal text, verbs that define a unary relationship are classified as classes. Such class names may also appear as a relation where that verb describes a binary or n-relationship. For example, the main verb *arrested* in the text *Mel was arrested* describes a unary relationship. To solve this peculiarity, the verb *arrested* is identified as a class in its base form. This means we have *Arrested* as a class. However, in a case of binary relationship, for example, *Mel was arrested for robbery*, the main verb *arrestedFor* is treated as a relation linking Mel to robbery as *Mel was arrested for robbery*. As such, the verb assumes a class position when it defines a unary relationship and an object property when it defines a binary relationship.

Step 15: We define domains and ranges of the identified relations as well as the characteristics as a way of restricting the relation. Since, object properties connect individuals from the domain to individuals from the range. For example, the relation *carry-property* has *Person* class as domain and *Property* as range ( $\exists \text{carry-property} . > v \text{ Person}, \exists \text{carry-property} \bar{.} > v \text{ Property}$ ); *forceperson* has *Person* class as domain and range ( $\exists \text{force-person} . > v \text{ Person}, \exists \text{forceperson} \bar{.} > v \text{ Person}$ ). Also, relation hierarchies are created to relate them into superproperties and subproperties. For example, the relation *knock person* is a subproperty of *force-person* (*knock-person*  $v$  *force-person*).

Step 16: In same way, we identify datatype properties. These are the properties that link individuals to datatypes. Here, we identify *drink-volume* as datatype property.

Step 17: From the datatype properties, we identify the respective domains and ranges. For example, the domain and range of the datatype property *drink volume* are *Person* ( $\exists \text{drink-volume} . > v \text{ Person}$ ) and *xsd:string* ( $\exists \text{drink-volume} \bar{.} > v \text{ xsd:string}$ ), respectively.

Step 18: Here, we define the definable classes, which can be defined using OWL axioms or SWRL rules (cf. “[Legal rule acquisition and representation](#)”). Rules are often more intuitive to construct. Similar to definable classes, there are definable properties too, which can be defined using SWRL rules (cf. “[Legal rule acquisition and representation](#)”).

## Implementation of Legal Ontology Generation Tool

In this section, we present an implementation of a semi-automatic legal ontology generation tool. The tool follows 15 out of 18 fine-grain legal ontology construction steps defined in [5] to generate a legal ontology (see Fig. 1). The main thing in the implementation phase is the automation of 15 steps out of the 18 step legal ontology

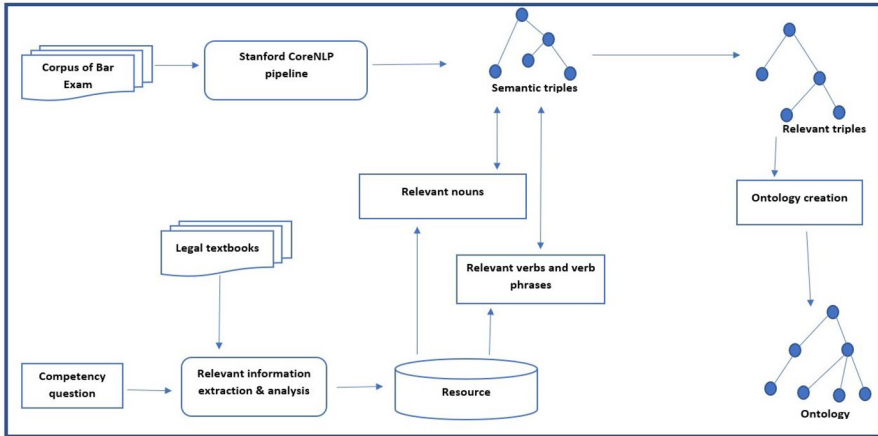


Fig. 3 A semi-automatic legal ontology generation tool

construction model with some human intervention for steps 1, 2 and 18 to extract structured information from an unstructured bar examination text to automatically generate an ontology for legal question answering.<sup>2</sup> Figure 3 shows the semi-automated legal ontology generation tool for the legal question answering task. In the following subsections, we will explain our methodology on how we implemented the 15 steps as well as the human intervention.

### Competency Question Generation and Analysis

Application of the law to real-life scenarios require both common-sense and legal knowledge. Here, we manually create a resource which will be used in the automated process. From Fig. 1 in “Methodology”, we implement steps 1 and 2

by carefully generating well structured coherent competency questions useful for extracting specific information relevant for answering the Bar examination questions. Hence, we analysed the answers derived from our structured competency questions to extract essential legal and common-sense knowledge (concepts and relations) useful for legal reasoning. The intention here is to use the competency questions as indicators of which keywords we want to extract to build a resource. Building a resource containing both legal and common-sense terminologies is essential in that they serve to enhance the identification and extraction of relevant concepts and relations in the automated process. Moreover, it is important to note that the legal vocabulary is derived from standard legal materials, e.g. textbooks and dictionaries. Common-sense terminology is that which has in and of itself no intrinsic legal import. For instance, *robbery* is a legal term, while a *briefcase* is a non-legal term. If a person is robbed of a briefcase, then briefcase is a property in the legal setting. For

<sup>2</sup> <https://github.com/bfawei/Research>.

example, given the competency questions 1 (*What are the elements of robbery?*) as shown in Table 2, we extract relevant information from law textbooks:

E1 “A person is guilty of robbery if he steals, and immediately before or at the time of doing so, and to do so, he uses force on any person or puts or seeks to put any person in fear of being then and there subjected to force” [8].

E2 “Person forcibly steals property and when, in the course of the commission of the crime or of immediate flight therefrom, he or another participant in the crime displays what appears to be a pistol, revolver, rifle, shotgun, machine gun or other firearm.”<sup>3</sup>

The relevant and necessary concepts and relations are identified from law textbooks and are extracted for the construction of a legal resource. The identification and extraction process is manually carried out in consultation with legal professionals to ensure fairness in keyword selection. The consultation is to enable us extract important and relevant concepts and relations as well as other salient information along with some useful heuristic knowledge from court experience in the construction of the resource.

Here, we identify and extract key concepts as well as action words or relations from the different units as well as terminological relations from WordNet. See some of the selected keywords from the unit of texts (E1 and E2) and terminological relations derived from WordNet:

Verb and verb phrase: commit, attempt to kill, kill, force, take, apply, etc.—  
Nouns: Person, Theft, Larceny, Robbery, Property, Threat, Fear, etc.

To identify and extract key concepts and relations useful for constructing legal ontologies, we need to process the source text. One step of this process is the identification of words in the text which are relevant for constructing the knowledge base. For this, we use the resource, which is a list of words consulted when processing the source text. The manually extracted keywords from steps 1 and 2 are then used to build the resource related to the particular crime. We then use this resource as a lookup list in the automated process of extracting relevant concepts and relations (see Fig. 3).

We manually extracted both nouns, verbs and verb phrases. The extracted nouns, verbs and verb phrases are used to create a resource containing legal concepts and relations. With this resource, we would be able to delineate from a given legal text the set of nouns that are relevant and the ones that are irrelevant. The relevant ones must match in some ways to our list of legal concepts (resource of legal keywords) either by direct or synonym match. For example, suppose we have the concept (money) in our resource of legal concepts and relations, and from a legal text we have (money and cash). The word money would have a direct match with concept in our resource (money  $\equiv$  money) while cash will match as a synonym (cash  $\equiv$  money).

<sup>3</sup> <http://ypdcrime.com/penal.law/article160.htm#p160.15>.

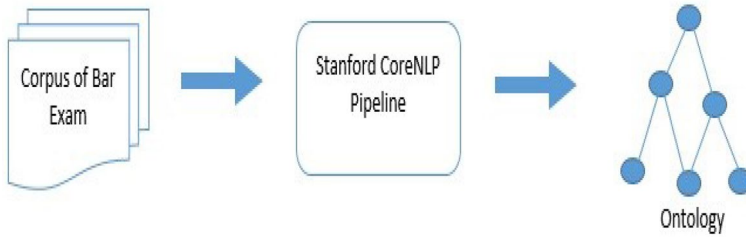


Fig. 4 From text to ontology

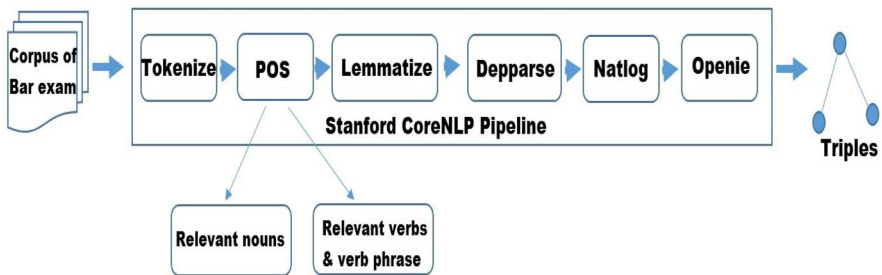


Fig. 5 Input text processing

## Input Text Processing

Stanford NLP is a combination of varying processing components, which are targeted at collecting specific information from documents in the different processing steps. The essence is to take the annotations produced by the pipeline to provide a simple way of relating the NLP and domain annotations to concepts and relations in creating an ontology (see Fig. 4). However, it is important to emphasize that our semi-automatic legal ontology learning tool does not extend the Stanford parser but only uses it to preprocess the input text and produce semantic triples.

We implement step 3 by applying the Stanford CoreNLP [37] to perform tokenization, part-of-speech tagging (POS) and named entity recognition (see Fig. 5). The essence is to identify and extract both legal and common-sense keywords useful to reason with to answer the Bar exam questions. The tagged tokens from the input text are matched against the resource built from steps 1 and 2 (see Step 4 Fig. 1). If a token in the text is a noun and matches against the resource, then it is extracted as a relevant noun and if it is a verb or verb phrase, then it is extracted as a relevant verb or verb phrase (see Step 11 Fig. 1). Also, to identify the main entities which the actions in the text refers to, we applied the Named Entity Recognition System (NERS) to identify and extract textual information referring to someone or something. This is important, since the interest of the law is to protect lives and properties. We obtain the set of named entities in the input texts into a predefined classification like the name of persons, organizations, locations, quantities, times, etc. With the NERS we are able to identify the instance classes. For example, using

the resource and NERS for extracting the relevant concept and relation from the text “*Mel should be acquitted.*”, *Mel* will be identified as *Person* and *should be acquitted* as a relevant verb phrase. Hence, the proper noun *Mel* becomes an instance of the *Person* concept (see Step 6 Fig. 1). We implemented the English model of the Stanford-corenlp which contains (CoreAnnotations.NamedEntityTagAnnotation) annotation. However, the tokens that do not match either the elements in the resource or NERS are identified as irrelevant and are discarded. (see Step 5 Fig. 1).

Then, we performed lemmatization and named entity recognition [37] for steps 6 and 12, respectively (see Fig. 5). Lemmatization is carried out on the verbs and verb phrases to define the lemmas of the given word or phrase, thereby eliminating the various forms of words that have the same core meaning (see Step 12 Fig. 1). For example, *takes* and *took* are lammatized to *take*.

For steps 7, 8, 13, and 14, we adopt WordNet 2.1 database. Here, after the lemmatization each token is looked up in the WordNet database to search for hypernyms. If a hypernym is found, it is identified and extracted as additional and necessary concept or relation. In the same way, the extracted hypernyms are useful for creating the class hierarchy.

In addition, we then applied dependency parsing, Natural Logic in NLP (natlog) and Open Information Extraction (openie) for steps 15, 16, and 17. Dependency parsing identifies a sentence and attaches a syntactic structure to the sentence. It generates two argument predicates that expresses the relationship (syntactic relations) between the words in the sentences. For example, from the sentence “Mel knocked the passenger to the floor”, the Stanford Parser generates:

```

Universal dependencies, enhanced
nsubj(knocked-2, Mel-1)
root(ROOT-0, knocked-2)
det(passenger-4, the-3)
dobj(knocked-2, passenger-4)
case(floor-7, to-5) det(floor-7,
the-6) nmod(to(knocked-2,
floor-7)

```

The natlog component works based on a model of natural logic to do inferencing on natural language text. It decomposes an inference problem into chunks of atomic edits connecting the premises to the hypothesis. It recursively traverses through a dependency tree and predicts if an edge produces an independent clause which is logically entailed by the original. From the independent clauses maximally simple relation triples are extracted while maintaining the necessary context [38]. The openie extracts relational tuples, specifically binary relations from a natural language text [39]. With this, we are able to identify relational triples as subject-predicate-object statements from the source text. The relevant triples are identified by matching each of the elements (Subject-PredicateObject) to the resource containing extracted nouns and relations (see Step 15 Fig. 1). Matching triples are extracted as relevant triples. Consequently, if a triple (subject-predicate-object) relation relates

an object to its characteristics. Then, the predicate in the tuple is extracted as data property and the respective subject and object are selected as the domain and range of the predicate.

However, if a matching word is not found, the word is looked up in WordNet dictionary to obtain synonyms. The WordNet synonym is then matched against the resource. The reason for the WordNet look up is to capture predicates having different lexical expressions. For example, suppose we have two words *possess* and *own* as predicate and relation from triple and resource, respectively. We can assert that the two concepts *possess* and *own* do not match lexically. Hence, we search for the synonyms or hypernyms of the word *possess* from WordNet and match them with the resource *own* based on exact match. If a match is found, we identify the predicate as relevant and as a sub-property of the matching relation in the list. The approach is helpful in capturing strings of words, for example, we have *go to* and *attend* in triple and resource, respectively. By this approach, we are able to capture *go to* as a synonym of *attend*.

### Ontology Modelling Primitives

For the legal ontology creation, the basic modelling primitives adopted in this research are: concepts, instances, taxonomies, relations. From the relevant triples which are extracted and translated into RDF triples, the single and two or three word subjects and objects are extracted and used to create ontology classes. For example, Person, Robbery, Larceny, Police officer, Wrist watch and so on. For class hierarchy creation (subclass-of-relations), we define our taxonomy based on some basic intuitions of the law. For example, our classification of criminal offenses is based on severity of the offense and the respective punishment involved for which someone is convicted of the crime can receive. As such, in our context of use, criminal offenses are classified into capital offense, felonies, misdemeanors, felony-misdemeanors and infractions<sup>4</sup>. Also, the law is meant to protect lives and property. Properties include personal and real properties, substance or things of value like money, computer program, and so on that can be charged or compensated for<sup>5</sup>. For common-sense knowledge concepts, we adopt the WordNet-based technique to extract respective hypernyms in creating class hierarchy.

In addition, ontology instances are identified based on proper and common nouns. A common noun is the standard name for person, place and thing. This is important in that most of the bar examination questions may not refer to a specific person, place or thing. For example, “The homeless girl broke into Homer’s house at night.” We can identify girl, Homer, house and night as common nouns. These nouns can then be selected as noun instances in the text.

<sup>4</sup> <https://open.lib.umn.edu/criminallaw/chapter/1-4-classification-of-crimes/>

<sup>5</sup> [https://www.lawserver.com/law/state/new-york/ny-laws/ny\\_penal\\_law\\_155-00](https://www.lawserver.com/law/state/new-york/ny-laws/ny_penal_law_155-00)



## Legal Rule Acquisition and Representation

Rules can be used to express definable classes and properties. In our case, we captured criminal law and procedure rules from bar examination preparatory material [6–8, 40] and in consultation with domain experts. The expression of legal rules in SWRL is not a simple task and requires interpreting and formalizing the source text.

The acquired rules were then expressed in the Semantic Web Rule Language (SWRL), which makes use of the vocabulary defined in our OWL ontology. The rules trigger in a forward chaining fashion. The essence is to ensure a consistent way of reasoning to exploit both the ontology and rules to draw inferences. SWRL rules are in the form of Datalog, where the predicates are OWL classes or properties. Moreover, rules may interact with OWL axioms, such as domain and range axioms for properties. For example, given the legal rule:

$$\text{own\_property}(?x, ?pr) \wedge \text{decide\_to\_steal}(?y, ?pr) \wedge \text{take\_property}(?y, ?pr) \\ \wedge \text{differentFrom}(?x, ?y) \rightarrow \text{has\_committed\_larceny}(?y)$$

The property *own property* has a domain of Person ( $\exists \text{own property} . > \text{v Person}$ ) and a range of Property ( $\exists \text{own property}^- . > \text{v Property}$ ) as defined in the ontology. They add implicit constraints on variable *?x* and *?pr*, which must be instances of Person and Property, respectively.

All atoms in the premises need to be satisfied for the rule to be triggered. For example, for the crime of robbery, suppose P1 is *taking*, P2 is *by force*, P3 *use of weapon*, and P4 *robbery*. Suppose we have the legal rule (simplified by removing the variables):  $P1 \wedge P2 \wedge P3 \rightarrow P4$ . Assuming that only P2 and P3 hold in the knowledge base, then we cannot assert that robbery. The fact that there was an application of force on someone and the presence of weapon does not constitute a robbery, since taking is not involved. Martin and Storey [8] describe the elements of robbery as “theft by force or putting or seeking to put any person in fear of force.” Therefore, the elements: theft and force are the main focus and must be explicitly defined in the rule. The extracted and transformed robbery rule from [8, 40] and *Panel Law art 160* in relation to our ontological concepts and properties is given as:

$$\text{in\_possession\_of}(?y, ?pr) \wedge \text{force\_person}(?x, ?y) \wedge \text{take\_property}(?x, ?pr) \\ \wedge \text{differentFrom}(?x, ?y) \rightarrow \text{has\_committed}(?x, \text{robbery}).$$

Due to domain and range axioms, the variables *?x* and *?y* are instances of the Person class while “*?pr*” is an instance of the Property class. The rule can be read as:

If person *?y* is in possession of property *?pr* and person *?x* forced *?y* and take property *?pr* and *?x* is different from *?y* then *?x* has committed robbery.

Also, Martin and Storey describe the elements of crime as “*actus reus + mens rea = offense*” [8]—the concurrence of the two elements *actus reus* and *mens rea*. We translate these elements into rules, where an *offense* is:

$$has\_committed(?x, ?y) \wedge has\_intent(?x, ?i) \rightarrow guilty\_of\_offense(?x)$$

Here, due to the domain and range axioms from the ontology,  $?x$  is an instance of the Person class,  $?y$  is an instance of the Crime class, and  $?i$  is an instance of the Intention class. The atom *has committed*( $?x$ ,  $?y$ ) corresponds to the *actus reus* and *has intent*( $?x$ ,  $?i$ ) to *mens rea* as the elements of crime. The rule can be read as:

“If person  $?x$  has committed a crime  $?y$  and person  $?x$  had intention  $?i$  to commit a crime  $?y$ , then person  $?x$  is guilty of an offense”.

A more complex example enables reasoning to acquittal. Note the chaining of rules between conclusions and premises, where the conclusion of rule (a) is a premise of rule (c), and the conclusion of rule (c) is a premise of rule (d).

- (a)  $carry\_property(?x, ?p) \wedge Property(?p) \rightarrow in\_possession\_of(?x, ?p)$
- (b)  $has\_committed(?x, ?r) \wedge negate\_required(?m, ?i) \wedge Intent(?i) \rightarrow did\_not\_intend(?x, ?r)$
- (c)  $force\_person(?x, ?y) \wedge take\_property(?x, ?p) \wedge in\_possession\_of(?y, ?p) \wedge charge\_with(?x, ?r) \wedge differentFrom(?x, ?y) \rightarrow has\_committed(?x, ?r)$
- (d)  $has\_committed(?x, ?r) \wedge did\_not\_intend(?x, ?r) \wedge Crime(?r) \rightarrow should\_be\_acquitted\_of(?x, ?r)$

The importance of using this approach is that legal rules defined are reusable and the whole process could lead to generalization of the rules. Some rules could be applicable to other legal subdomains. In addition, having a clear rule set will be helpful to automate legal rule development process in future.

## Application of the Ontology

To understand the dependencies between the rules, we tested each of the rules individually with a populated ontology. Our queries are formulated in the Semantic Query-Enhanced Web Rule Language (SQWRL), which is based on SWRL and provides SQL-like operators for querying information from OWL ontologies. We assumed the following ABox assertions.

*carryproperty(passenger, briefcase), take\\_property(Mel, briefcase), knock\\_person(Mel, passenger), Crime(robbery), perform\\_bymistake(Mel, robbery), Intention(intent), differentFrom(Mel, passenger)*

**Table 3** Evaluation results

Index	Previous work	CLOR
True positive	15.0	16.0
False positive	16.0	16.0
True negative	32.0	32.0
False negative	1.0	0.0
Precision	0.48	0.5
Recall	0.93	1.0
Accuracy	0.73	0.75
F-measure	0.63	0.66

from the example question in Table 1. In effect, the SQWRL queries enable assessment of the ontology relative to the competency questions as well as the relevant to rule firing.

We have the following queries for the ontology:

- The query *in possession of*( $?x, ?r$ )  $\rightarrow$  sqwrl : *select*( $?x, ?r$ ) is used in querying the possession rule (a) and the output is ( $?x=passenger, ?r=briefcase$ ).
- The query *has committed*( $?x, ?r$ )  $\rightarrow$  sqwrl : *select*( $?x, ?r$ ) is used for querying the robbery rule (c) and the output is ( $?x=Mel, ?r=robbery$ );
- The query *did not intend*( $?x, ?r$ )  $\rightarrow$  sqwrl : *select*( $?x, ?r$ ) for querying the did not intend to commit rule (b) and the output is ( $?x=Mel, ?r=robbery$ );
- The query *should be acquitted*( $?x, ?r$ )  $\rightarrow$  sqwrl : *select*( $?x, ?r$ ) for the acquit rule in (d) and the output is ( $?x=Mel, ?r=robbery$ ).

However, to be sure that the rules satisfy the dependencies in sequence to arrive at the final conclusion, we altered the ABox fact *carry property*(*passenger*, *briefcase*) in the ontology. Then, we executed the same queries and examine the output, which did not generate any results. In addition, we altered the fact *knock person*(*Mel*, *passenger*), leaving all others intact. As a result, the query *in possession of*( $?x, ?r$ )  $\rightarrow$  sqwrl : *select*( $?x, ?r$ ) returned ( $?x=passenger, ?r=briefcase$ ), while the rest did not generate any results. Also, we kept all facts intact and altered *perform bymistake*(*Mel*, *robbery*). In executing the queries, we observed that the last two queries did not generate any result. Finally, we also tested the situation where we had all facts intact and altered *Crime*(*robbery*) fact. We observed that all rules work as usual, due to the fact that the *Crime* class is the range of *has committed*. Thus, even if we do not have *Crime*(*robbery*) explicitly stated, it is entailed by the ontology. This shows that the dependencies amongst the rules were executed in the right order.

## Ontology Evaluation

While the criminal law and procedure ontology and rule sets are still under development, we evaluated them in three ways: task based, competency questions, and ontology evaluation tools. We note that while the results in Table 3 is incrementally better than previously reported, this has been done in the context of a systematic and transparent methodology. The advantage now is that in error analysis, we can trace the problem to a particular part of the methodology and revised that component, then rerun and test. We should emphasise that CLOR was developed on 12 multiple choice questions out of 16, which constitute the training data (results below), then applied to 4 new questions (for a 30% increase of data), which constitute the testing data, as they had not been included amongst the questions used to develop the ontology. Of the 4 testing data, CLOR accounted for three, while CLOR required slight modifications to take the fourth question into account. This demonstrates that our iterative approach to the development of CLOR is feasible.

Firstly, we took a task-based approach, assessing the performance of CLOR with respect to benchmark answers to the bar examination questions. A semantic interpretation is said to be accurate if it produces the correct answer based on the question with respect to the application of the law. We present a preliminary experimental results from 16 MBE questions, each with four possible answers, constituting a total of 64 question–answer pairs. CLOR was evaluated against our previous work [41]. See evaluation result in Table 3. Secondly, we evaluated

the system against our competency questions in the development stage. The ontology is evaluated with respect to how its concepts match with the respective terms in the competency questions. Here, we want to ascertain the completeness of the ontology in relation to the competency questions and whether the ontology answers the list of previously formed competency questions or not.

Finally, we used several ontology evaluation tools. To ensure the ontology is consistent and its general qualities are sustained, we applied the Pellet reasoner and the Ontology Pitfall Scanner (OOPS) [42, 43]. The ontology is consistent. The OOPS is a web-based evaluation tool for evaluating OWL ontologies. Its evaluation is mainly based on structural and lexical patterns that recognize pitfalls in ontologies. Currently, the tool contains 41 pitfalls in its catalogue, which are applied worldwide in different domains. OOPS evaluates an OWL ontology against its catalogue of common mistakes in ontology design and creates a single issue in Github with the respective summary of the detected pitfalls with an extended explanation for more information. Each of the OOPS pitfalls is evaluated into three categories based on its impact on the ontology:

- (a) Critical means that the pitfall needs to be corrected else it may affect the consistency and applicability of the ontology, amongst others.
- (b) Important means that it is not critical in terms of functionality of the ontology but it is important that the pitfall is corrected.
- (c) Minor means that it does not impose any problem. However, for better organization and user friendliness, it is important to make correction.

**OOPS! Ontology Pitfall Scanner!**

OOPS! (Ontology Pitfall Scanner!) helps you to detect some of the most common pitfalls appearing when developing ontologies. To try it, enter a URI or paste an OWL document into the text field above. A list of pitfalls and the elements of your ontology where they appear will be displayed.

Scanner by URI:  [Scanner by URI](#)

Example: [http://data.semanticweb.org/ns/swc/swc\\_2009-05-09.rdf](http://data.semanticweb.org/ns/swc/swc_2009-05-09.rdf)

Scanner by direct input:  [Scanner by RDF](#)

```

ontology-429"
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:owl="http://www.w3.org/2002/07/owl#"
xmlns:xml="http://www.w3.org/XML/1998/namespace"
xmlns:swrlb="http://www.w3.org/2003/11/swrlb#"
xmlns:untitled-ontology-
429="http://www.semanticweb.org/r01bjf15/ontologies/2018/7/untitled-ontology-
429#"
xmlns:swrl="http://www.w3.org/2003/11/swrl#"
xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
    
```

Uncheck this checkbox if you don't want us to keep a copy of your ontology. [Go to advanced](#)

### Evaluation results

It is obvious that not all the pitfalls are equally important; their impact in the ontology will depend on multiple factors. For this reason, each pitfall has an importance level attached indicating how important it is. We have identified three levels:

- Critical** 🚨: It is crucial to correct the pitfall. Otherwise, it could affect the ontology consistency, reasoning, applicability, etc.
- Important** ⚠️: Though not critical for ontology function, it is important to correct this type of pitfall.
- Minor** 🟡: It is not really a problem, but by correcting it we will make the ontology nicer.

[Expand All] | [Collapse All]

Results for P04: Creating unconnected ontology elements.	10 cases   Minor 🟡
Results for P08: Missing annotations.	127 cases   Minor 🟡
Results for P13: Inverse relationships not explicitly declared.	53 cases   Minor 🟡
SUGGESTION: symmetric or transitive object properties.	3 cases

According to the highest importance level of pitfall found in your ontology the conformance badge suggested is "Minor pitfalls" (see below). You can use the following HTML code to insert the badge within your ontology documentation:

Want to help?

- Suggest new
- Provide feedb

Documentatic

- Pitfall catalogo
- User guide
- Technical rep

Related paper

- ISWIS 2014
- FKAW 2012

Fig. 6 OOPS evaluation summary

Not all the pitfalls in [42] are relevant for evaluating our ontology. Moreover, some of these pitfalls depend on the domain being modelled while others on the specific requirements or use case of the ontology.

Our criminal law and procedure ontology were evaluated against the 41 pitfalls in OOPS (see evaluation result in Fig. 6). The evaluation is to ensure that our ontology is free from the critical and important pitfalls. On evaluating our ontology, we observed that critical pitfalls polysemous elements are not present in the ontology as well as synonymous classes. Other pitfalls like “is” relations, equivalent properties, specialization of too many hierarchies and primitive and defined classes are not misused. Also, the naming criteria are consistent and so on. However, it returned an evaluation report of 3 minor pitfalls as shown in Fig. 6) (P04, P08, and P13). P04 is about creating unconnected ontology elements, P08 is missing annotations while P13 is about inverse relations not explicitly declared. At this initial evaluation, these pitfalls appear to be irrelevant, since the construction of the ontology is still in progress.

Furthermore, we manually evaluated the ontology generated from the semiautomatic legal ontology generation tool against the manually constructed ontology in [5], which establishes our gold standard. Both the semi-automated and manual ontologies are constructed following the same ontology development steps in “Methodology”. To evaluate the semi-automatic analysis, an ontological analysis of each question was manually created, following the steps described in

**Table 4** Semi-automatic ontology generation tool evaluation result

	MAN	AUTO	MAN	AUTO	MAN	AUTO	MAN	AUTO
Questions	Q1	Q1	Q7	Q7	Q15	Q15	Q66	Q66
Class	18	16	18	16	9	9	24	24
Sub class	7	8	4	4	5	7	9	9
Subsubclass	5	0	1	2	0	1	2	2
Instance	8	13	8	4	5	8	21	21
Object property	16	14	13	9	17	13	15	21
Axioms	112	113	92	89	86	87	108	171

“**Methodology**”. Furthermore, an ontology was semi-automatically created for each question using the ontology learning tool, as described in “**Implementation of legal ontology generation tool**”. Table 4 compares the ontology created by the manual analysis (MAN) to the ontology created by the semi-automatic analysis (AUTO). Considering questions 1, 7, 15, and 66, the semi-automated tool shows reasonable performance (see Table 4).

Also, we evaluated the ontology based on the automation steps. Taking steps 3 and 10; and from our initial MBE example question above, using the Stanford NLP Parts of Speech recogniser (POS) successfully selected all the nouns (job, quart, vodka, bus, home, briefcase,...). The relevant nouns and identification of noun classes, instances and instance classes are also successful. In analysing the results from the handcrafted resource of keywords (nouns and relations), POS and NERS, the results from step 4, 5, and 6 as well as 11 appear to be good.

In the same way, enhancing the tool with the WordNet database for steps 8, 12, 13, and 14 yields a reasonable result. However, this seems to leave some important information out. For instance, some of the hypernyms relating to the legal domain are not in WordNet because WordNet was developed for open, common-sense domains. Hence, using it for a specialized domain like the law does not yield all the desired results. Using a legal database of concepts along with other knowledge bases like Yago, which is a combination of WordNet and DBpedia, may improve the extraction of hypernyms and superclasses covering both legal and non-legal domains. That is, information relating to the law will be extracted from the legal database while common-sense knowledge will be extracted from the Yago database.

For steps 15, 16 and 17, natlog and openie successfully extract the relevant facts. However, they were developed for open domain information extraction and as such are not very successful in extracting all the required facts in some questions. For example, from question 22 facts like” (local ordinance forbid sale alcohol, Person sell alcohol, Person sell to Student,...)”. Also, a range of other questions like 29, 35 and so on had issues relating to extraction of all the necessary facts useful for reasoning.

## Conclusion

We have developed a methodology and semi-automated legal ontology generation tool following our step-by-step approach in OWL with legal rules in SWRL to infer conclusions. The ontologies generated for each question represent legal concepts and the relations among those concepts in criminal law and procedure. As far as we know, this is the first fine-grained methodology for constructing legal OWL ontologies with SWRL rules. We envision that such a methodology can be applied to other domains and applications of textual entailment, such as fake news detection [44]. However, it is important to emphasize that the system does not address a range of challenging issues such as defeasible reasoning complex compound nouns, polysemy, legal named entity recognition, and implicit information in legal text. Ontology learning techniques [45, 46] might be used to learn further OWL axioms, which can be used together with SWRL rules. Due to the uncertainties introduced by NLP and ontology learning techniques, we will consider some uncertainty/fuzzy extensions of OWL [47] and SWRL [48] in our future work as well as legal rule learning for the question answering task. We will also develop a Legal NER system to serve in identifying legal named entities such as Judge and Barrister. Issues related to scalability would also be considered.

**Open Access** This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

## References

1. National conference of bar examiners: the MBE multistate bar examination sample MBE III, <http://www.kaptest.com/bar-exam/courses/mbe/multistate-barexam-mbe-change>. Accessed 5 Sep 2015
2. Segura-Olivares, A., Garcia, A., Calvo, H.: Feature analysis for paraphrase recognition and textual entailment. *Res. Comput. Sci.* **70**, 119–144 (2013)
3. Magnini, B., Zanolini, R., Dagan, I., Eichler, K., Neumann, G., Noh, T., Pado, S., Stern, A., Levy, O.: The Excitement open platform for textual inferences. *ACL (system demonstrations)*, pp. 43–48 (2014)
4. Fawei, B.J., Wyner, A.Z., Pan, J.Z.: Passing a USA National bar exam: a FirstCorpus for experimentation. *LREC 2016, Tenth International Conference on Language Resources and Evaluation*, pp. 3373–3378 (2016)
5. Fawei, B., Wyner, A., Pan, J.Z., Kollingbaum, M.: A methodology for a criminal law and procedure ontology for legal question answering. In: *Proceedings of the Joint International Semantic Technology Conference*, Springer, New York, pp. 198–214 (2018)
6. Emmanuel, S.L.: *Strategies and tactics for the MBE (multistate bar exam)*, vol. 2. Wolters Kluwer, Maryland (2011)
7. Herring, J.: *Criminal law: text, cases, and materials*. Oxford University Press, Oxford (2014)
8. Martin, J., Storey, T.: *Unlocking criminal law*, 4th edn. Routledge, New York (2013)
9. Davis, E., Marcus, G.: Commonsense reasoning and commonsense knowledge inartificial intelligence. *Commun. ACM* **58**(9), 92–103 (2015)
10. Liu, H., Singh, P.: ConceptNet—a practical commonsense reasoning tool-kit. *BT Technol J* **22**(4), 211–226 (2004)

11. El Ghosh, M., Naja, H., Abdulrab, H., Khalil, M.: A ontology learning process as a bottom-up strategy for building domain-specific ontology from legal texts. *ICAART* (2), Springer, New York, pp. 473–480 (2017)
12. Hwang, R., Hsueh, Y., Chang, Y.: Building a Taiwan law ontology based on automatic legal definition extraction. *Appl. Syst. Innovat.* **1**(3), 22 (2018)
13. Deng, L., Wang, X.: Context-based semantic approach to ontology creation of maritime information in Chinese. In: 2010 IEEE international conference on granular computing, pp. 133–138 (2010)
14. Johnson, J.R., Miller, A., Khan, L.: Law enforcement ontology for identification of related information of interest across free text documents. In: Intelligence and security informatics conference (EISIC), pp. 19–27 (2011)
15. Osathitporn, P., Soonthornphisaj, N., Vatanawood, W.: A scheme of criminal lawknowledge acquisition using ontology. *Software engineering, artificial intelligence, networking and parallel/distributed computing (SNPD), 2017 18th IEEE/ACIS International Conference on IEEE*, pp. 29–34 (2017)
16. Bak, J., Cybulka, J., Jedrzejek, C.: Ontological modeling of a class of linked economic crimes. In: *Transactions on computational collective intelligence IX*, Springer, New York, pp. 98–123 (2013)
17. Ceci, M., Gangemi, A.: An OWL ontology library representing judicial interpretations. *Semantic Web* **7**(3), 229–253 (2016)
18. Gangemi, A., Sagri, M. T., Tiscornia, D.: A constructive framework for legal ontologies. *Law and the semantic web*. Springer, pp. 97–124 (2005)
19. Gangemi, A.: Introducing pattern-based design for legal ontologies. *Law, Ontologies and the Semantic Web*, pp. 53–71. (2009)
20. Suárez-Figueroa, M.C., Gómez-Pérez, A., Fernández-López, M.: The neon methodology for ontology engineering. *Ontology engineering in a networked world*. Springer, pp. 9–34 (2012)
21. De Nicola, A., Missikoff, M.: A lightweight methodology for rapid ontology engineering. *Commun. ACM* **59**(3), 79–86 (2016)
22. Gangemi, A.: Design patterns for legal ontology constructions. *LOAIT* **2007**, 65–85 (2007)
23. Golbreich, C., Horrocks, I.: The obo to owl mapping, go to owl 1.1. In: *Proceedings of the OWLED 2007 workshop on OWL: experiences and directions*, Citeseer (2007)
24. Hoekstra, R., Breuker, J., Di Bello, M., Boer, A.: The LKIF core ontology of basic legal concepts. *LOAIT* **321**, 43–63 (2007)
25. Athan, T., Boley, H., Governatori, G., Palmirani, M., Paschke, A., Wyner, A.: Oasislelegalruleml. In: *Proceedings of the fourteenth international conference on artificial intelligence and law*, ACM, pp. 3–12 (2013)
26. Gandon, F., Governatori, G., Villata, S.: Normative requirements as linked data. The 30th international conference on legal knowledge and information systems. *JURIX* (2017)
27. Moens, M. F., Spyns, P.: Norm modifications in defeasible logic. *legal knowledge and information systems: JURIX 2005: the Eighteenth Annual Conference*, 134(13) IOS Press (2005)
28. Navarro, P.E., Rodríguez, J.L.: *Deontic logic and legal systems*. Cambridge University Press, Cambridge (2014)
29. Clarkson, K.W., Miller, R.L., Cross, F.B.: *Business law: text and cases: legal, ethical, global, and corporate environment*. Cengage Learning, Canada (2010)
30. N. Y. State Board of Law Examiners. Course materials for the New York law course and New York law examination, <https://www.newyorklawcourse.org/CourseMaterials/NewYorkCourseMaterials.pdf>. Accessed 15 Jul 2018
31. Ren, Y., Parvizi, A., Mellish, C., Pan, J.Z., Van Deemter, K., Stevens, R.: Towards competency question-driven ontology authoring. *European Semantic Web Conference*. Springer, pp. 752–767 (2014)
32. Bezerra, C., Freitas, F., Santana, F.: Evaluating ontologies with competency questions. In: *WI-IAT*. pp. 284–285 (2013)
33. Y. Ren, A. Parvizi, C. Mellish, J.Z. Pan, K. van Deemter, Robert S.: Towards competency question-driven ontology authoring. In: *ESWC* (2014)
34. Dennis, M., van Deemter, K., Dell’Aglío, D., Pan, J.Z.: Computing authoring tests from competency questions: experimental validation. In: *ISWC* (2017)
35. Pan, J.Z., Vetere, G., Gomez-Perez, J.M., Wu, H.: *Exploiting linked data and knowledge graphs for large organisations*. Springer, ISBN:978-3-319-45652-2 (2016)
36. Pan, J.Z., Calvanese, D., Eiter, T.H., Horrocks, I., Kifer, M., Lin, F., Zhao, Y.: Reasoning web: logical foundation of knowledge graph construction and querying answering (2017)



37. Manning, C., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S., McClosky, D.: The Stanford Core NLP natural language processing toolkit. In: Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations, pp. 55–60 (2014)
38. Bowman, S.R., Potts, C., Manning, C.D.: Learning distributed word representations for natural logic reasoning. In: Proceedings of the Association for the Advancement of Artificial Intelligence Spring Symposium (AAAI), pp. 10–13 (2015)
39. Angeli, G., Premkumar, M.J., Manning, C.D.: Leveraging linguistic structure for open domain information extraction. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics (ACL 2015) (2015)
40. N. Y. State Board of Law Examiners. Course materials for the New York law course and New York law examination, <https://www.newyorklawcourse.org/CourseMaterials/NewYorkCourseMaterials.pdf>. Accessed 15 Jul 2018
41. Fawei, B., Wyner, A., Pan, J.Z., Kollingbaum, M.: Using legal ontologies with rules for legal textual entailment. In: Proceedings of ALCOL2017, Springer, New York, pp. 317–324 (2017)
42. Poveda-Villalón, M., and Gómez-Pérez, A., Suárez-Figueroa, M. C.: Oops! (ontology pitfall scanner!): an online tool for ontology evaluation. In: IJWSIS 10(2):7–34 (2014)
43. Poveda-Villalón, M., Suárez-Figueroa, M. C.: OOPS!—ontology pitfalls scanner!. Ontology Engineering Group. Universidad Politécnica de Madrid (2012)
44. Pan J.Z., Pavlova S., Li C., Li N., Li Y., Liu J.: Content based fake news detection using knowledge graphs. In: ISWC (2018)
45. Maedche, A., Staab, S.: Ontology learning for the semantic web. *IEEE Intell. Syst.* **16**(2), 72–79 (2001)
46. Zhu, M., Gao, Z., Pan, J.Z., Zhao, Y., Xu, Y., Quan, Z.: TBox learning from incomplete data by inference in BelNet+. *Knowl. Based Syst.* **75**, 30–40 (2015)
47. Stoilos, G., Stamou, G., Pan, J.Z., Tzouvaras, V., Horrocks, I.: Reasoning with very expressive fuzzy description logics. *JAIR* **30**, 273–320 (2007)
48. Pan, J.Z., Stoilos, G., Stamou, G., Tzouvaras, V., Horrocks, I.: f-SWRL: a fuzzy extension of SWRL. *J. Data Semantic.* **4090**(2006), 28–46 (2006)

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Biralatei Fawei** is currently a Ph.D. student at the University of Aberdeen and a Lecturer at the Department of Mathematics/Computer Science, Niger Delta University, Wilberforce Island, Bayelsa State, Nigeria. He received his BSc degree (2.1 division) in Computer Science from Niger Delta University, Nigeria in 2006 and an MSc degree in Information Technology from the University of Wolverhampton, UK in 2012. His research interests include question answering, knowledge representation, artificial intelligence and law, and textual document retrieval.

**Jeff Z. Pan** received his Ph.D. in computer science from the University of Manchester in 2004. He joined the faculty in the Department of Computing Science at the University of Aberdeen in 2005. He is the Director of the Aberdeen–Wuhan Joint Research Lab on Knowledge Engineering and Information Security. His research focuses primarily on knowledge representation, artificial intelligence, and data science, in particular on knowledge graph-based learning and reasoning, as well as their applications, such as those in healthcare and smart city. He is a key contributor of the W3C OWL (Web Ontology Language) standard. He leads the development of the award-winning TrOWL reasoner, the only ontology reasoner that Oracle Spatial and Graph (from v12) uses via the OWL–DBC database connection. He is an internationally leading expert on Knowledge Graph, being the Chief Editor of the first two books on Knowledge Graph, a new technology that is widely used by world leading IT companies, such as Google. As the Chief Scientist and Coordinator of the EU Marie-Curie K-Drive project, he led a team of 22 Marie-Curie Fellows on Knowledge Graph and Ontology research. He is an Associate Editor of the Journal of Web Semantics (JWS) and of the International Journal on Semantic Web and Information Systems (IJWSIS). He was/is a General Chair of 4th Joint International Semantic Technology Conference, a General Chair of 8th Chinese Semantic Web and Web Science Conference and was/is a Program Chair of RR2007, JIST2011, RW2016, and ISWC2020.

**Martin Kollingbaum** holds a Ph.D. in Computing Science (University of Aberdeen). He is a Lecturer at the University of Aberdeen, School of Natural and Computing Science. His research interests are Multiagent Systems, with a focus on normative reasoning, and applications in Transport and Logistics.

**Adam Wyner** has a Ph.D. in Linguistics (Cornell University, 1994) and a Ph.D. in Computer Science (King's College London, 2008). As of May 2018, he is an Associate Professor of Law and Computer Science at Swansea University, which is a joint position in the School of Law and Department of Computer Science. Previously, he was a Lecturer at the University of Aberdeen in the Department of Computer Science. He has taught various topics in Computer Science and legal informatics. He is an established researcher in Artificial Intelligence with numerous publications and presentations bearing on natural language processing (rule-based and machine learning), information extraction, ontologies, argumentation, controlled languages, case-based reasoning, policy consultations, and semantic web. He is on the OASIS Technical Committee for LegalRuleML, which is developing a machine-readable standard for legal rules.