Automatic Extraction of Legal Norms: Evaluation of Natural Language Processing Tools

Gabriela Ferraro[1,2], Ho-Pun Lam[1], Silvano Colombo Tosatto[1], Francesco Olivieri[1], Mohammad Badiul Islam[1], Nick van Beest[1], and Guido Governatori[1]

[1] Data61, CSIRO, Australia
firstname.lastname@data61.csiro.au
[2] Australian National University, Canberra, Australia

Abstract. Extracting and formalising legal norms from legal documents is a time-consuming and complex procedure. Therefore, the automatic methods that can accelerate this process are in high demand. In this paper, we address two major questions related to this problem: (i) what are the challenges in formalising legal documents into a machine understandable formalism? (ii) to what extent can the data-driven state-of-the-art approaches developed in the Natural Language Processing (NLP) community be used to automate the normative mining process. The results of our experiments indicate that NLP technologies such as relation extraction and semantic parsing are promising research avenues to advance research in this area.

Keywords: Natural Language Processing · Automatic Rule Extraction · Legal Norms · Evaluation

1 Introduction

Legal documents contain lots of information that needs to be interpreted and processed in order to decide whether an organisation's business processes are compliant with the regulations defined in such documents. Dealing with these regulations can be both costly and time consuming as it requires field experts to understand the legal aspects of the documentation concerning the required setting with respect to its interpretation and intent.

Having legal documents written into a format that could be automatically interpreted by machines would drastically reduce the cost associated with such process. However, in order to achieve that, it is first necessary to extract the set of legal norms from the legal documents and translate them into a machine understandable meaning representation, such as First-Order Logics (FOLs) or lambda calculus formulas, among others.

Language technologies developed by the Natural Language Processing (NLP) community have been shown capable of dealing with some of the challenges of mining normative information from legal documents [9,39,7,33]. However, legal documents differ considerably from other types of documents with respect to their structure, sentence length, and other characteristics.

This paper aims to explain the problem of generating legal norms from legal documents, and to provide an investigation into the challenges of formalising legal norms from a NLP perspective. We also envision a holistic NLP system for legal norms generation from text, and outline some promising research avenues, connecting the requirements of legal norms to existing NLP research streams. Furthermore, we report some preliminary experiments on *neural semantic parsing* and *open relation extraction*, as they can be considered as an intermediate step towards automated generation of legal norms from legal documents.

2 Problems with Legal Documents

The process of acquiring knowledge from legal documents is usually considered as a process of writing norms or rules [29] that, to a certain extent, have basically a conditional structure like [31,17]:

$$r : \textsf{IF} \quad a_1, \ldots, a_n \quad \textsf{THEN} \quad c \tag{1}$$

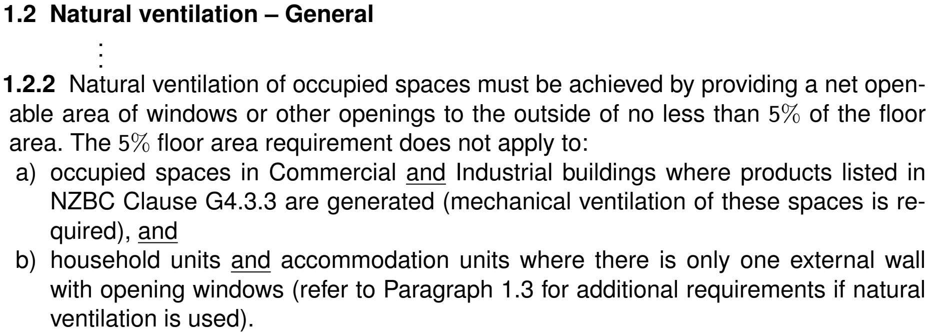
where r is the unique identifier of the rule, $a_1, \ldots, a_n$ denote the *antecedent* representing the conditions (including the context(s) under which it is created) of applicability of the legal norm, and c denotes the *conclusion* representing the *desired consequence* (or the *normative effect*) of the legal norm. Notice here that legal norms can be unconditional, meaning that the antecedent of the rule can be *empty*. However, such legal norms may be deceptively ambiguous and limit the case of conditional legal norms in some situations [23].

Generating legal norms from legal documents is far from being trivial or intuitive. As discussed in [37], legal documents are technically so complex that even human lawyers are having difficulties understanding and applying it. Therefore, it is important to discuss some of the characteristics of legal documents that needs to be taken into account towards the automation of legal norms extraction. In this section, we present some of those characteristics and elucidate important challenges of generating legal norms.

2.1 Cross-referencing

Technically, a legal document is structured into different chapters, articles, sections and subsections, where each of these sections and/or subsections contain several sentences each has their own specific *goal(s)*, *objective(s)*, and *context(s)* (or *scope(s)*) in which the clauses are applicable. This modular nature of legislation allows legal drafters to focus on a particular aspect of legislation when drafting the document. Hence, referencing information from one section to another, or to other regulations, is not uncommon in legal documents. This is referred to as cross-referencing.

Cross-referencing can help to avoid ambiguity that may occur across different sections of the documents and can help to indicate whether a sentence is an elaboration, subordinate, or prevailing with respect to other sentences or definitions. They can also be used to confer a priority to reconcile potential conflicts by discarding existing goals or substituting alternative top-level goals [24]. In other words, cross-referencing defines the context of lexical units which should be taken into account when generating legal



1.2 Natural ventilation – General

⋮

1.2.2 Natural ventilation of occupied spaces must be achieved by providing a net openable area of windows or other openings to the outside of no less than 5% of the floor area. The 5% floor area requirement does not apply to:

a) occupied spaces in Commercial and Industrial buildings where products listed in NZBC Clause G4.3.3 are generated (mechanical ventilation of these spaces is required), and

b) household units and accommodation units where there is only one external wall with opening windows (refer to Paragraph 1.3 for additional requirements if natural ventilation is used).

Fig. 1. Example of logical ambiguity, NZBC Clause G4 Ventilation (adopted from [25])

norms. Hence, identifying cross-references is an important task in mining legal norms as they define the context of linguistic utterance and can help to resolve referential and lexical ambiguities, which will be discussed below.

2.2 Ambiguity and Inconsistent Terminology

Drafters of legal documents try to avoid ambiguity and, ideally, produce a document that results in only one interpretation (e.g., avoid pronouns, avoid synonyms to refer to the same concept, add attributes to identify parties, use punctuation to define the scope of quantifiers, etc.). To avoid lexical ambiguity, legal documents usually include a glossary (sometimes referred to as Definitions) with the most important lexical items and their corresponding definitions.

However, as natural language is used to write the legislation, unintended ambiguities may arise. The most probable ambiguity is *referential ambiguity*, which occurs when a word or phrase has multiple meanings due to different restrictions/conditions. From a linguist point of view, the meaning of lexical units need to be *inferred from the context* in which they appear. However, in legal documents, in addition to the current (local) context, conditions related to the meaning of lexical units can be *inherited* from its parent statement(s), other document sections, and even from external documents via cross-referencing.

Logical ambiguity refers to the use of natural language that can be mapped to different logical interpretations [4]. Consider the fragment of legislation as shown in Figure 1. Syntactically, the terms "commercial building" and "industrial building" in the first sub-condition, "household units" and "accommodation units" in the second sub-condition, and the two sub-conditions are connected using the conjunction term "and". However, logically (or semantically), the statement is in fact representing conditions to the four different types of building and should be represented using disjunction, i.e., *or*, in the legal norm.

Apart from the ambiguity problems just mentioned, the *inconsistent use of terminology* across different documents may also impact the reasoning process, even though they do not directly affect the generation of legal norms at the sentence level.

2.3 Sentence Complexity

In legal documents, sentences tend to be extremely long in comparison with sentences from other domains. The average number of lexical units in a sentence written in the English Wikipedia is about nineteen [38], while sentences from legal documents can have more than fifty units, as can be seen in Figure 1.

Legal documents also tend to have a complex syntactic structure, in which coordinate and subordinate constructions are frequent. These pose a challenge to current NLP technologies, such as syntactic analysers and predicate-arguments extraction tools, which usually struggle to correctly capture the scope of coordinate conjunctions and the antecedent of subordinate phrases.

2.4 Normative effects and Deontic Modalities

Legal documents typically contain normative information. Legal concepts such as *right*, *no right*, *privilege*, *duty*, *power*, *disability*, *immunity*, and *liability*, are important as they significantly affect the way that we interpret legislation. In the field of Artificial Intelligence & Law, capturing deontic modalities or behaviours such as *obligations*, *permissions*, and *prohibitions* are of major concern. These modalities can be considered as a sub-type of normative effects and should be identified and attached to legal norms when appropriate. A more comprehensive and updated list of legal-related normative effects can be found in [30].

3 Related work

Algorithms developed by the NLP community have been used to model the process of extracting normative information from legal documents with certain success. [9] proposed an automated concept and norm extraction framework, by exploiting the use of a Juridical (Natural) Language Constructs (JLC) as an intermediate format between the legal texts and the formal model. In their approach, the JLC is essentially a set of patterns that can appear in the legal documents. Legal knowledge is identified and constructed using noun and verb phrase patterns that will later be transformed into formal rules. However, no evaluation of their automated approach has been reported in the paper.

[26,19] proposed to convert sentences in Japanese into a logical representation using a rule-based approach over a morphological and dependency analysis. [39] presented a linguistic oriented rule-based approach to extract deontic rules from regulations and found that serious issues may appear when mapping thematic roles to syntactic positions. [3] described a technique to automatically extract semantic knowledge from legal texts. Instead of using pattern matching methods based on lexico-syntactic patterns, they proposed to adopt syntactic dependencies between terms extracted with a syntactic parser, which is also the approach used in [8], but with different kinds of information extracted. [12] developed a translation systems to learn the semantics of unknown words from syntactically similar words with known meanings, which is able to translate into a variety of formal language representations. As mentioned by the authors, their method needs to be tested with a bigger corpus.

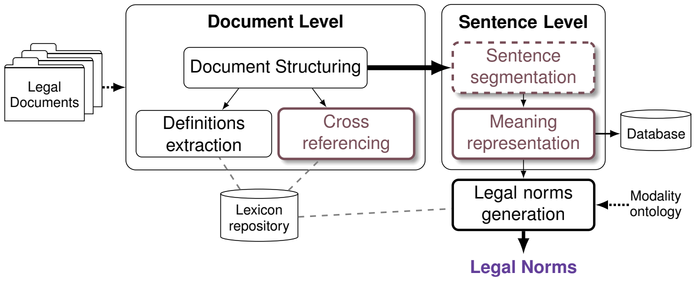


Fig. 2. Normative Rules Generation Framework

Recently, [33] used the syntactic structures and hand-written rules to identify entities (*agent*, *action*, *condition*, *exception*, among others), which are relevant in legal knowledge representation, and can be useful for the automatic extraction of legal norms.

Although it would be valuable to compare the mentioned approaches to our proposal, an experimental comparison is not possible because of the lack of evaluation material. For example, none of the works mentioned above have released their evaluation data sets.

4 Normative Rules Generation Framework

Figure 2 depicts an overview of the process of generating normative rules as we have envisioned.[3] We have taken into account the characteristics of legal documents as well as normative rule requirements. The NLP related modules are highlighted so that it is easy to connect with the research avenues proposed in the next section.

As illustrated, the process is divided into Document Level and Sentence Level modules. The Document Level module reads a legal document (usually a pdf or XML file) as input and passes it to the Document Structuring module, which identifies the hierarchical structure of the document, such as sections, sub-sections, articles, and sentences.

The Document Structuring module also takes care of lexical aspects that need to be solved at the document level. As mentioned before, legal documents have a section in which the most important terms in the document are defined (i.e., the glossary). Thus, we envisioned the Definitions Extraction sub-module here will identify the terms and their corresponding definitions and store them in a lexical repository, which can be used at the later stage of the process, for disambiguation purposes.

The Sentence Level module takes the sentences and information identified at the Document Level module as input, and processes each sentence at a time. The core of this module is the Meaning Representation sub-module, which is used to identify the *predicate units* and their *arguments* in the sentence for the generation of the meaning representation. This module is method agnostic, which means it can be modelled in different ways, e.g., distilling predicates from syntactic trees or generating a logic formula using

[3] Here, the term *generation* is used in a loose way (normative rules can be generated or extracted or distilled or built, etc.), and is not used to refer exclusively to the text generation problem study by the Natural Language Generation community.

semantic parsing, among other possibilities (see Section 5). The output of this module should reflect the predicate units in a given sentence and is formalism agnostic. This allows choice between different formalisms, such as Abstract Meaning Representation (AMR) [2], LegalRuleML [28], Process Compliance Language (PCL) [14], the "neo-Davidson" approaches [13], among others, as an intermediate formalism that capture all required information from the legal text, to ease the construction of legal norms at the later stage of the process. In addition, Figure 2 depicts an optional sub-module called Sentence Segmentation that can be used to determine how a legal sentence should be divided into sub-sentences or clauses, so as to reduce its complexity for further process while preserving its meaning and context.

The output of the Meaning Representation module is then passed to the last module, i.e., Legal Norms Generation module, which generates the normative rules, with the associated modalities such as *obligation* and *permission*, in accordance with the requirements specified by the applications. It is also expected that the lexical terms used in the rules should be linked to the terms in the lexical repository (when possible) so that consistency across different rules can be maintained.

5 Research Avenues

This section outlines some promising avenues to advance research in normative rules generation. One of our focuses is on proposing research streams related to the meaning representation module depicted in Figure 2, as it is the most challenging part in the whole normative rules generation process.

With new applications like this, evaluation can be a challenge in itself due to the lack of annotated data sets (i.e., sentences and their corresponding normative rules). Besides, this also impedes the possibility of using supervised machine learning, which is usually the preferred method for NLP. Fortunately, there are promising related techniques that can advance research in this area, which we are going to explore in this section.

Sentence Complexity Reduction: As discussed in Section 2, sentences in legal domain tend to be long and considerably complex in structure, which leads to the question of whether it makes sense to reduce the sentence complexity as a pre-processing step to provide simpler sentences as input to normative rule generation.

Sentence complexity can be reduced in at least two ways: by applying some sort of intra-sentential segmentation and by applying simplification techniques. However, simplification techniques are not appropriate for parsing legal text since it involves lexical and grammatical modifications and there is a risk of changing meaning.

Sentences can be segmented, for example, into clauses [35]. This task usually involves two sub-tasks, clause boundary detection and clause type identification, and both can be modelled as classification problems.

Sentence segmentation should be applied in a conservative way, for instance by only segmenting sentences that are likely to struggle with later processing. The state-of-the-art methods for measuring sentence complexity are developed in the context of readability assessment — using lexical and sentence level features to build binary classifiers or regression algorithms [32,36]. In the context of normative rule generation,

Table 1. Example of semantic parsing as an intermediate steps towards the generation legal norms

	Example
Sentence	*A large building is any building with a net lettable area greater than* $300\,\mathrm{m}^2$.
Logic formula	`lambda $0 (if (A large building: $0) then (is any building with a lettable area greater than ($0 300m2)))`
Legal norm	$building,\ greaterThan(netLettableArea, 300) \rightarrow largeBuilding$
Sentence	*For the purposes of subclause 2.4, a person is responsible for an individual if the person is a parent of the individual.*
Logic formula	`lambda $0 $1 (if (and (person:$0) (individual: $1) (parent of ($0 $1))) then (responsible for (for purpose of subclause 2.4 ($0 $1))))`
Legal norm	$subclause(2.4),\ parentOf(A, B) \Rightarrow_O responsible(A, B)$

this is going to be a challenge, as measurement should be calculated based only on the grammatical complexity of sentences, disregarding the lexical aspects.

Semantic Parsing: Semantic parsing is the task of mapping sentences in natural language to a meaning representation, such as a logic formula. It can be seen as an intermediate step towards the generation of normative rules due to its predicate-argument structure, which can be used as building blocks for normative rule generation. Table 1 shows some example sentences written in natural language, their corresponding logic formula in lambda calculus (we follow the notation from [21]), and their corresponding normative rule represented using PCL [14].

Currently, the semantic parsing community is shifting from domain and meaning representation dependent approaches to more universal models using techniques such as transfer learning [18,10], multi-task learning [34,15], and data augmentation [16,20]. However, the data sets for semantic parsing are restricted at the moment to very specific domains such as geographical locations, fly booking system, and jobs, to just name a few, which contain only a small set of vocabularies (usually less than 100 words). As concluded by [18], the main limitation of applying transfer learning to semantic parsing is the domains of the data sets are considerable different. More importantly, the structures of the meaning representation proposed are simple and repetitive [11], which may not be sufficient to represent normative information in the legal domain. Hence, in addition to improving the efficiency and accuracy of semantic parsing, the challenge here is to develop more complex data sets with bigger sets of vocabularies, and a more diverse set of meaning representation structures that can cater the needs of representing legal information.

Open Relation Extraction: Relation extraction can be seen as a shallow semantic parsing analysis. In this context, *open* means that relations are not restricted to a set of relations defined beforehand, but to all possible relations. More concretely, open relation extraction focuses on capturing the verbal predicates and their arguments, and format them as tuples. Hence, it can be seen as a (shallow) meaning representation. Note that open relation extraction usually focuses on verbal predicates (cf. the first sentence in

Table 2. Example of relation extraction as an intermediate steps towards normative rules using OpenIE.

	Example
Sentence	*A large building is any building with a net lettable area greater than* $300\,\mathrm{m}^2$.
RE Tuples	‘A large building’, ‘is building with’, ‘net lettable area greater than 300 m2’
Normative rule	$building, greaterThan(netLettableArea, 300) \rightarrow largeBuilding$
Sentence	*For the purposes of subclause 2.4, a person is responsible for an individual if the person is a parent of the individual.*
RE Tuples	‘For the purposes of subclause 2.4’, ‘a person’, ‘is responsible for an individual’, ‘if the person is a parent of the individual’.
Normative rule	$subclause(2.4), parentOf(A, B) \Rightarrow_O responsible(A, B)$

Table 2, in which the predicative adjective *greater than* and its arguments is not captured by the relation extraction tool OpenIE [1]) and most methods are heavily depend on syntactic analysis. Thus, one of the main challenges of using relation extraction as an intermediate step towards normative rules generation is to have high quality syntactic parser outputs for complex sentences. Another possibility would be to explore approaches that combine sentence segmentation, syntactic analysis and relation extraction for complex sentences. Another research avenue would be to extend relation extraction to predicate nouns and adjectives. This is an interesting research direction, since not all nouns and adjectives are predicates.

6 Experiments

In this section, we present empirical results based on experiments in neural semantic parsing and relation extraction in the legal domain.

6.1 Neural Semantic Parsing Experiments

As discussed in Section 5, semantic parsing technologies can be used to generate meaning representations from which it is possible to distil normative rules. We have chosen to evaluate two models, namely: the sequence-to-sequence model proposed by [6], and the coarse-to-fine model proposed by [22], as described in more detail below:

Sequence-to-Sequence: This model [6] consists of an encoder and decoder with two different L-layer recurrent neural networks with long-short-term-memory (LSTM) units, which recursively process tokens one by one. A sentence in natural language x is encoded into a vector representation, and decoded into a sequence $y_1, \ldots, y_{|y|}$ that is learned conditioned on the encoded vector $p(y|x)$. Additionally, this approach can incorporate with an attention mechanism to integrate encoder-side information (also referred to as context version) for the current hidden state.

Coarse-to-Fine: In this model [22], the decoder process has two steps. In the first step, a decoder generates a sketch of the meaning representation, omitting arguments and variable names. Then, in the second step, a second decoder fills the missing

Table 3. Data sets splits and average length of input and output sequences (dash signs '-' indicates 'no data for those cells')

	ATIS	GEO	RegTech
Training set	4434	600	140
Development set	491	-	-
Test set	448	280	79
Total	5373	880	79
Avg. input length	10.6	7.3	26.75

details conditioned on the input and the sketch. As such, the sketches constrain the generation process. Thus, $p(y|x)$ is decomposed into a two stage generation which are realized as: $p(y|x) = p(y|x, a)\, p(a|x)$, where $a = a_1, \ldots, a_{|a|}$ is an abstract sketch representation of y.

The sequence-to-sequence approach can be considered as a vanilla model. Meanwhile, the coarse-to-fine model restricts the decoder by modelling the structure of the output meaning representations.

Data Sets for Training Semantic Parsers

GEO This is a standard semantic parsing benchmark which consists of a set of queries to a database of U.S. geography. The meaning representation of this data set is a lambda-calculus like formula (as in [6]) and values for variables such as *city*, *state*, *country*, *river* and *number* are identified beforehand. We used the splits provided by [6] in our experiments.

ATIS This is another standard benchmark in semantic parsing which consists of queries to a flight booking system. Sentences are paired with lambda-calculus expressions and values for variables such as *date*, *time*, *city*, *aircraft code*, *airport*, *airline* and *number* are identified beforehand. We used the standard splits provided in the data set in our experiments.

RegTech[4] We have developed this data set for evaluating the performance of semantic parsing in the legal domain. At the moment, the data set consists of 140 sentences extracted from regulations from New Zealand and Australia. Following the annotation schema of [6], sentences in the data set are paired with logical expressions that are used to indicate the scope of the predicates and their arguments. The annotations were carried out by annotators with a background in logic. Each annotator annotated a set of 10 sentences (without overlap). Next, two annotators reviewed the logical expressions, agreed on the best practises and produced a consistent final version of the data set.

Information and examples of the data sets are shown in Tables 3 and 4, respectively.

[4] Available at: http://bitbucket.csiro.au/users/fer201/repos/regtech-dataset

Table 4. Examples of sentences written in natural language and their corresponding meaning representation for the three data sets

GEO	*what is the capital of the state with the largest population density?* `(capital:c (argmax $1 (state:t $1) (density:i $1)))`
ATIS	*is there ground transport available at the airport?* `(lambda $0 e (and (ground-transport $0) (from-airport $0 ap0)))`
RegTech	*a large building is any building with a net lettable area greater than 300 m2.* `lambda $0 (if (a large building:$0) then (is any building with a lettable area greater than ($0 300m2)))`

Neural Semantic Parsing Settings Semantic parsing is evaluated on accuracy, which is defined as the proportion of the input sentences that have an exact match with their gold standard logical form. Both models are trained on GPU with their default hyperparameters.

6.2 Relation Extraction Experiments

We have chosen to evaluate the approach from OpenIE [1] due to its two promising features: (i) it is open to any relation (predicates), thus relations do not have to be defined in advance; (ii) it models long-range dependencies, thus is able to extract relations from long sentences which frequently appear in legal texts. The approach consists of two major steps. It first learns a classifier for splitting sentences into shorter utterances by traversing dependency parsed trees recursively. Subsequently, through the use of some natural logic, it tries to shorten these utterances into small clauses or compact sentences while maintaining the necessary context. Finally, it identifies *subject-verb-object* tuples with a traditional relation extraction approach.

Relation Extraction Settings We run OpenIE off-the-shelf using Stanford CoreNLP with its default parameters. We have selected 78 sentences from the RegTech data set and obtained sets of tuples for each sentence. Since sentences in the RegTech data sets are pairs with logic representations rather than triplets, we cannot automatically compute the quality of the extracted tuples. Instead, we manually inspect the extracted triplets adopting the following criteria:

- Number of analysed sentences
- Number of tuples outputs
- Number of correct tuples: a tuple is considered correct if it is a predicate in the sentence and if its arguments (which explicitly appear in the sentence) are extracted as well
- All predicates are extracted (complete analysis): YES or NO
- Contained coordinated conjunctions: YES or NO

6.3 Semantic Parsing Evaluation Results

Table 5 shows the results of the semantic parsing experiments. We first verify the ability of the methods in analysing sentences of different complexity, assuming the sentence

Table 5. Semantic parsing evaluation on test sets (short sentences contained less than 10 tokens ($<$ 10 tokens) and long sentences contained 10 or more tokens ($\geq$ 10 tokens))

	sequence-to-sequence	coarse-to-fine
ATIS (all)	83.03	86.83
ATIS ($<$ 10 tokens)	85.05	88.28
ATIS ($\geq$ 10 tokens)	62.99	81.10
GEO (all)	83.57	88.93
GEO ($<$ 10 tokens)	92.09	89.47
GEO ($\geq$ 10 tokens)	68.93	84.85
GEO (140)	29.28	29.86
RegTech (140)	18.28	18.42

Table 6. Examples of gold-standard formulas (GS) from RegTech and their corresponding predicted (P) formulas generated by the coarse-to-fine semantic parsing model

GS	`lambda $0 $1 $2 (and ((endorsing body or)`
	`(supplier of:$1))) then (must (be replaced by ($0 $1)))`

P	`lambda $0 $1 $2 (and ((endorsing body:$0)`
	`(supplier of:$1 (food:$2)) (must (<U> ($0 $1)))))`

GS	`lambda $0 (and ((claim:$0)) (must (not (refer to ($0`
	`prevention of (or ((disease) (disorder) (condition))))))))`

P	`lambda $0 (and ((claim:$0)) (must (not (refer to ($0`
	`diagnosis of (or ((disease) (disorder) (condition))))))))`

length is an indicator of sentence complexity (the longer the sentence, the more likely it is to contain complicated semantic structures). As already shown in Table 4, sentences in the legal domain (for example, in RegTech) have an average input length that is significantly higher than sentences in standard semantic parsing data sets such as ATIS and GEO. Since the size of the RegTech data set is small and potentially not sufficient to properly train a semantic parser, we evaluate the parsers performance on long sentences by splitting the ATIS and GEO test sets in two subsets: sentences containing less than 10 tokens and sentences with 10 or more tokens, respectively. For comparison, we also report the evaluation results with the full test sets: ATIS (all) and GEO (all). As expected, results shows considerable drops when parsing long sentences, The drop is more dramatic for sequence-to-sequence, which dropped about 20 points when parsing long sentences with ATIS (from 83.03 to 62.99), and about 14 points with GEO (from 83.57 to 68.99). The coarse-to-fine model is more resilient to long sentences, thus its performance is less hurt, dropping about 6 points with ATIS (from 86.83 to 81.1) and about 4 points with GEO (from 88.93 to 84.85). The experiments indicate that having a model that constrained its decoder with some kind of structure is preferable for the legal domain.

As mentioned, the size of RegTech is potentially too small to train a semantic parser. Results shows an accuracy of 18.28 for sequence-to-sequence, and a slightly better performance for coarse-to-fine with an accuracy of 18.42. For comparison, we report

Table 7. Relation extraction evaluation results using the RegTech test set

	Counts	Percentage (%)
Total sentences	78	-
Analysed sentences	46	58.97
Triplets outputs	246	-
Correct triplets	166	67.48
Complete analysis	19	24.36
Contained coord. conjunctions	17	21.79

results on GEO trained only with 140 sentences that were randomly chosen. Results indicate that models trained with a limited amount of data are not able to generalise well. A qualitative error analysis performed on the output of the semantic parsers trained with RegTech indicates that the models are able to correctly output the structure of the logic formulas, but failed to instantiate the appropriate vocabulary (cf. Table 6). We attribute this limitation to the vocabulary mismatch between the training and testing sets. The main take away from these experiments is that current technologies for semantic parsing are data hungry, and creating data sets for semantic parsing is not a trivial task.

In the ATIS and GEO data sets, values for in-domain variables e.g., city, airport, etc. are anonymised before training, thus the vocabulary size is reduced, making encoding and decoding simpler. Note that variables in RegTech are not anonymised. Consequently, the vocabulary size is bigger, which impacts the generalisation power of the model. Nevertheless, we argue that it is less costly to increase the size of RegTech and train a semantic parser, than investing in a syntax-based approach, which requires to manually annotate in-domain syntactic structures to re-train a syntax analyser and to write grammars to distil the rules from the trees.

6.4 Relation Extraction Evaluation Results

Overall, relation extraction is able to correctly identify many predicates from sentences in the RegTech data set. It also struggles to analyse some sentences, thus producing no tuples. Table 7 shows the experiments results for relation extraction when applied to RegTech. The OpenIE tool was able to extract predicates from 58.97% of the sentences. The results also show that OpenIE manages to analyse 21.79% of the sentences with coordinate phrases. This is not surprising, since OpenIE relies on automatic syntactic parsing, which is known to perform poorly with complex linguistics structures [5]. The tool extracted 246 predicates from which 67.48% are correct. Further assessment was done to evaluate whether all the predicates were extracted from a given sentence. The results show that for only 24.36 sentences (24.36%) all the predicates were extracted (cf., Table 7 *'Complete analysis'*).

Error Analysis In the following, we summarised the OpenIE most salient errors when applied to the RegTech data set.

- **No output tuples**: the analysis of long and complicated sentences usually produce no output tuples. Sentences that were not parsed are usually long and contained

Table 8. Examples of natural language sentences from the RegTech data set and their tuples extracted using OpenIE

	Complete analysis
Input	*Toilet facilities for males must contain WC pans and basins and may contain urinals.*
Tuples	toilet facility; contain; urinal toilet facility; must contain; WC pan toilet facility; must contain; basin
Input	*A urinal flushing system shall have the cistern outlet at least 450 mm above the sparge pipe and comply with Table 5.*
Tuples	flushing system; shall have; cistern outlet at least 450 mm flushing system; comply with; Table 5
	Incomplete analysis
Input	*Flushing systems for sanitary fixtures shall use either cisterns or flushing valves.*
Tuples	flush system; shall use either; cistern
Input	*This Verification Method can be used for housing, communal residential, communal non-residential and commercial buildings.*
Tuples	verification method; can; can used for housing
	No analysis
Input	*WC pans and basins are required in any building where people: a) live or are accommodated or b) work or c) eat food or drink on the premises or d) assemble.*
Tuples	-
Input	*For determining the insulation requirements of the building envelope, buildings other than housing are classified as being either small or large.*
Tuples	-

itemised coordinate constructions, cf., Table 8 ‘*No analysis*’. We also found that the analysis of simple sentences in which a negation proceeds a verbal predicate produce no tuples, e.g., *This Standard does not apply to food that is intended for labelling*.

- **Prepositional phrases and noun post modifiers**: prepositional phrases and other noun post modifiers are not consistently analysed by OpenIE, cf., Table 8 *‘Toilets facilities for males …’*, in which ‘*for males*’ is not output as an attribute of ‘*Toilets facilities*’. In contrast, from the text spanning ‘*cistern outlet at least 450mm above the sparge pipe …*’, it manages to capture the first prepositional phrase (‘*at least 450mm*’) that modifies the noun ‘*cistern outlet*’, and fails to capture the second prepositional phrase ‘*above the sparge pipe*’ as a modifier.
- **Modal verbs**: predicates with auxiliary modal verbs such as *must*, and *shall* are inconsistently analysed by OpenIE, e.g., for the sentence: *Flushing systems for sanitary fixtures shall use either cisterns or flushing valves*, one of the extracted tuple is: *‘systems’*; *‘use’*; *‘flushing valves’*.

OpenIE relies on syntactic analysis. Thus, it is not surprising that it struggles to find the correct scope of prepositional phrases. Prepositions describe relations between terms, primarily about location, direction and time, and capture important content that should be available in a legal norm. This is a big challenge to syntax based approaches because of the well-known problem of attachment ambiguity, which is most commonly produced by prepositional attachment ambiguity. This happens when a prepositional phrase can be attached to a syntactic tree in more than one place. Similarly, syntactically analysing long and complex sentences is currently an open problem. This is mainly due to the treebanks used to train syntactic parsers, which mostly contained short sentences from news articles. This problem is likely to be alleviated by re-training a syntactic parser with in-domain sentences. The performance of OpenIE can also be boosted by reducing the complexity of the input sentences beforehand, for instance by applying a pre-process segmentation strategy, as proposed by [27].

In this paper, we did not address the problem of attaching modalities to the rules since that process can be considered as a separate problem. Nevertheless, we are aware that rules needs to be legally characterised with modalities to be useful for reasoning.

7 Conclusions and Future Work

This paper presents the task of generating normative rules from legal documents. Normative rules are executable statements generated from sentences written in natural language, which are used for reasoning. To attract research in this area, we have outlined some promising research avenues, connecting the requirements of normative rules to existing NLP research streams. One of the main challenges is how to produce an accurate meaning representation from complex sentences. We report some preliminary experiments on neural semantic parsing and open relation extraction, as they can be considered as an intermediate step towards the generation of normative rules. Results show that there is plenty of room for improvement. For future work, we plan to increase the RegTech data set, and to run experiments following the recommendations and new versions of semantic parsing benchmarks from [11].

References

1. Angeli, G., Johnson 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. pp. 344–354. ACL 2015, Beijing, China (2015)
2. Banarescu, L., Bonial, C., Cai, S., Georgescu, M., Griffitt, K., Hermjakob, U., Knight, K., Koehn, P., Palmer, M., Schneider, N.: Abstract Meaning Representation for Sembanking. In: Pareja-Lora, A., Liakata, M., Dipper, S. (eds.) Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse. pp. 178–186. Sofia, Bulgaria (Aug 2013)
3. Boella, G., Di Caro, L., Robaldo, L.: Semantic Relation Extraction from Legislative Text Using Generalized Syntactic Dependencies and Support Vector Machines. In: Morgenstern, L., Stefaneas, P., Lévy, F., Wyner, A., Paschke, A. (eds.) Proceedings of the 7th International Web Rule Symposium: Theory, Practice, and Applications of Rules on the Web. pp. 218–225. RuleML 2013, Springer Berlin Heidelberg, Seattle, WA, USA (2013)
4. Breaux, T.D., Antón, A.I.: A Systematic Method for Acquiring Regulatory Requirements: A Frame-Based Approach. In: Proceedings of the 6th International Workshop on Requirements for High Assurance Systems. Pittsburg, PA, USA (Sep 2007)
5. Burga, A., Codina, J., Ferraro, G., Saggion, H., Wanner, L.: The Challenge of Syntactic Dependency Parsing Adaptation for the Patent Domain. In: Proceedings of ESSLLI-13 Workshop on Extrinsic Parse Improvement. EPI, Düsseldorf, Germany (Aug 2013)

6. Dong, L., Lapata, M.: Language to Logical Form with Neural Attention. In: Erk, K., Smith, N.A. (eds.) Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics. vol. 1: Long Papers, pp. 33–43. Berlin, German (Aug 2016)
7. Dragoni, M., Villata, S., Rizzi, W., Governatori, G.: Combining Natural Language Processing Approaches for Rule Extraction from Legal Documents. In: Pagallo, U., Palmirani, M., Casanovas, P., Sartor, G., Villata, S. (eds.) AI Approaches to the Complexity of Legal Systems International Workshops. pp. 287–300. AICOL 2015 (2015)
8. Dragoni, M., Villata, S., Rizzi, W., Governatori, G.: Combining NLP Approaches for Rule Extraction from Legal Documents. In: 1st Workshop on MIning and REasoning with Legal texts (MIREL 2016). Sophia Antipolis, France (Dec 2016)
9. van Engers, T.M., van Gog, R., Sayah, K.: A Case Study on Automated Norm Extraction. In: Gordon, T. (ed.) The 17th International Conference on Legal Knowledge and Information Systems. pp. 49–58. JURIX 2004, IOS Press, Amsterdam, Netherlands (2004)
10. Fan, X., Monti, E., Mathias, L., Dreyer, M.: Transfer Learning for Neural Semantic Parsing. In: Blunsom, P., Bordes, A., Cho, K., Cohen, S., Dyer, C., Grefenstette, E., karl Moritz Hermann, Rimell, L., Weston, J., Yih, S. (eds.) Proceedings of the 2nd Workshop on Representation Learning for NLP. pp. 48–56. Rep4NLP 2017, Vancouver, Canada (Aug 2017)
11. Finegan-Dollak, C., Kummerfeld, J.K., Zhang, L., Ramanathan, K., Sadasivam, S., Zhang, R., Radev, D.: Improving Text-to-SQL Evaluation Methodology. In: Gurevych, I., Miyao, Y. (eds.) Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics. pp. 351–360. ACL 2018, Melbourne, Australia (Jul 2018)
12. Gaur, S., Vo, N.H., Kashihara, K., Baral, C.: Translating Simple Legal Text to Formal Representations. In: Murata, T., Mineshima, K., Bekki, D. (eds.) 8th International Workshop on Juris-Informatics. pp. 259–273. JURISIN 2015, Kanagawa, Japan (Oct 2015)
13. Gordon, A.S., Hobbs, J.R.: A Formal Theory of Commonsense Psychology: How People Think People Think. Cambridge University Press, New York, NY, USA, 1st edn. (2017)
14. Governatori, G., Rotolo, A.: A Conceptually Rich Model of Business Process Compliance. In: Proceedings of the 7th Asia-Pacific Conference on Conceptual Modelling. pp. 3–12. APCCM 2010, ACS, Brisbane, QLD, Australia (Jan 2010)
15. Herzig, J., Berant, J.: Neural Semantic Parsing over Multiple Knowledge-bases. In: Barzilay, R., Kan, M.Y. (eds.) Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. pp. 623–628. ACL 2017, Vancouver, Canada (Jul 2017)
16. Jia, R., Liang, P.: Data Recombination for Neural Semantic Parsing. In: Erka, K., Smith, N.A. (eds.) Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics. pp. 12–22. ACL 2016, Berlin, Germany (Aug 2016)
17. Kelsen, H.: General Theory of Norms. Oxford University Press, Inc. (1991)
18. Kennardi, A., Ferraro, G., Wang, Q.: Domain Adaptation for Low-Resource Neural Semantic Parsing. In: MacKinlay, A., Piccardi, M. (eds.) Proceedings of the 17th Workshop of the Australasian Language Technology Association. pp. 107–113. ALTA 2019, Sydney, Australia (2019)
19. Kimura, Y., Nakamura, M., Shimazu, A.: Treatment of Legal Sentences Including Itemized and Referential Expressions – Towards Translation into Logical Forms. In: Hattori, H., Kawamura, T., Idé, T., Yokoo, M., Murakami, Y. (eds.) 2nd International Workshop on Juris-Informatics. pp. 242–253. JURISIN 2009, Asahikawa, Japan (Jun 2009)
20. Kočiský, T., Melis, G., Grefenstette, E., Dyer, C., Ling, W., Blunsom, P., Hermann, K.M.: Semantic Parsing with Semi-Supervised Sequential Autoencoders. In: Su, J., Duh, K., Carreras, X. (eds.) Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. pp. 1078–1087. EMNLP 2016, Association for Computational Linguistics, Austin, TX, USA (Nov 2016)

21. Kwiatkowski, T., Zettlemoyer, L., Goldwater, S., Steedman, M.: Lexical Generalization in CCG Grammar Induction for Semantic Parsing. In: Barzilay, R., Johnsn, M. (eds.) Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing. pp. 1512–1523. EMNLP 2011, Edinburgh, Scotland, UK (Jul 2011)
22. Lapata, M., Dong, L.: Coarse-to-Fine Decoding for Neural Semantic Parsing. In: Gurevych, I., Miyao, Y. (eds.) Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics. pp. 731–742. ACL 2018, Melbourne, Australia (Jul 2018)
23. Makinson, D.: On a Fundamental Problem of Deontic Logic. In: McNamara, P., Prakken, H. (eds.) Norms, Logics and Information Systems. New Studies in Deontic Logic and Computer Science, vol. 49, pp. 29–53. IOS Press, Amsterdam, Netherlands (1999)
24. Miall, H.: Emergent Conflict and Peaceful Change. Palgrave MacMillan (2007)
25. Ministry of Business, Innovation and Employment, New Zealand: New Zealand Building Code Clause G4 Ventilation: Acceptable Solutions and Verification Methods, 4th edn. (2017)
26. Nakamura, M., Nobuoka, S., Shimazu, A.: Towards Translation of Legal Sentences into Logical Forms. In: Satoh, K., Inokuchi, A., Nagao, K., Kawamura, T. (eds.) 1st International Workshop on Juris-Informatics. pp. 349–362. JURISIN 2007, Miyazaki, Japan (Jun 2008)
27. Niklaus, C., Cetto, M., Freitas, A., Handschuh, S.: Transforming Complex Sentences into a Semantic Hierarchy. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. pp. 3415–3427. Florence, Italy (Jul 2019)
28. OASIS LegalRuleML TC: OASIS LegalRuleML (2013), http://www.oasis-open.org/committees/legalruleml, [accessed: 7 Dec 2019]
29. Popple, J.: A Pragmatic Legal Expert System. Dartmouth (1996)
30. Rubino, R., Rotolo, A., Sartor, G.: An OWL Ontology of Fundamental Legal Concepts. In: Proceedings of the 9th Annual Conference on Legal Knowledge and Information Systems. pp. 101–110. JURIX 2006, IOS Press, Amsterdam, Netherlands (2006)
31. Sartor, G.: Legal Reasoning: A Cognitive Approach to the Law, A Treatise of Legal Philosophy and General Jurisprudence, vol. 5. Springer (2005)
32. Schwarm, S.E., Ostendorf, M.: Reading Level Assessment Using Support Vector Machines and Statistical Language Models. In: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics. pp. 523–530. ACL '05, Stroudsburg, PA, USA (2005)
33. Sleimi, A., Sannier, N., Sabetzadeh, M., Briand, L., Dann, J.: Automated Extraction of Semantic Legal Metadata Using Natural Language Processing. In: The 26th IEEE International Requirements Engineering Conference. pp. 124–135. IEEE, Banff, AB, Canada (Aug 2018)
34. Susanto, R.H., Lu, W.: Neural Architectures for Multilingual Semantic Parsing. In: Barzilay, R., Kan, M.Y. (eds.) Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. pp. 38–44. ACL 2017, Vancouver, Canada (Jul 2017)
35. Tjong, E.F., Sang, K., Déjean, H.: Introduction to the CoNLL-2001 Shared Task: Clause Identification. In: Daelemans, W., Zajac, R. (eds.) Proceedings of the 2001 Workshop on Computational Natural Language Learning. ConLL '01, Association for Computational Linguistics, Toulouse, France (Jul 2001)
36. Vajjala, S., Meurers, D.: Readability-based Sentence Ranking for Evaluating Text Simplification. CoRR (2016), http://arxiv.org/abs/1603.06009
37. Wieringa, R.J., Meyer, J.J.C.: Applications of Deontic Logic in Computer Science: A Concise Overview. In: Meyer, J.J.C., Wieringa, R.J. (eds.) International Workshop on Deontic Logic in Computer Science. pp. 17–40. DEON 1993, Amsterdam, Netherlands (Dec 1993)
38. Woodsend, K., Lapata, M.: WikiSimple: Automatic Simplification of Wikipedia Articles. In: Proceedings of the 25th AAAI Conference on Artificial Intelligence. pp. 927–932. AAAI 2011, AAAI Press, San Francisco, CA, USA (Aug 2011)
39. Wyner, A.Z., Peters, W.: On Rule Extraction from Regulations. In: Atkinson, K.M. (ed.) Proceedings of the 24th International Conference on Legal Knowledge and Information. pp. 113–122. JURIX 2011, IOS Press, Vienna, Austria (Sep 2011)