Problems and Prospects in the Automatic Semantic Analysis of Legal Texts
A Position Paper

Adam Wyner

Department of Computer Science, University of Liverpool
Ashton Building, Asthon Street, Liverpool, L69 3BX, United Kingdom
adam@wyner.info

Abstract

Legislation and regulations are expressed in natural language. Machine-readable forms of the texts may be represented as linked documents, semantically tagged text, or translation to a logic. The paper considers the latter form, which is key to testing consistency of laws, drawing inferences, and providing explanations relative to input. To translate laws to a machine-readable logic, sentences must be parsed and semantically translated. Manual translation is time and labour intensive, usually involving narrowly scoping the rules. While automated translation systems have made significant progress, problems remain. The paper outlines systems to automatically translate legislative clauses to a semantic representation, highlighting key problems and proposing some tasks to address them.

Keywords: Translation, Semantics, Syntax, Text Analysis

1. Introduction

Laws as found in legislation and regulations are expressed in natural language. To make laws automatically processable, they must be made machine-readable since the language of the law is, from the point of view of a computer, an unstructured sequence of characters. There are several approaches to making legal texts machine-readable, depending on the goals and purposes to be served by the processed text. Among the approaches, legal texts may be processed to link documents, to annotate for information extraction, and to parse and translate them to a logic. Each of the approaches has its own use. For linked documents, the objective is to identify components in the text that may be associated with some other web-based document. For example, references to a law in a text may be associated with a web-accessible link to the particular law or other further information, e.g. The British Nationality Act 1981. Google's Advanced Scholar Search facility allows searches restricted to terms in legal opinions and returns decisions that have links to cases cited in the decision, e.g. Advanced Micro Devices, Inc. v. Intel Corp.. Linking documents not only helps to identify related documents, but also to highlight *relationships* between texts; in legal decisions, case citations can be used to indicate the relevance of precedents. In another approach, legal documents can be automatically tagged with a variety of sorts of annotations to enable information extraction and fine-grained search in the body of legal documents (Maynard and Greenwood, 2012; Wyner and Peters, 2011). In the final approach, legal texts are processed and rendered into a machine-processable logic that can be used for testing consistency of laws, drawing inferences, and giving users meaningful explanations following a consultation. While the first two approaches have seen very rapid, widespread, and continuing development, the third has not, despite being one of the early achievements in AI and Law, its current commercial success, and well-developed NLP tools.

This position paper is a pointer to problems and prospects bearing on automatic translation of legal text from natural language to a machine-processable logic. We begin with some background, then turn to some aspects of state-of-the-art systems on semi-automated systems, and finally consider a fully automated system. The discussion is illustrated with a well-known working example.

2. Background - Manual Translation

One of the early ambitions and achievements of artificial intelligence and law was to formalise legislation as a logic program. Several large scale projects were carried out (Sergot et al., 1986; Bench-Capon et al., 1987; Sergot, 1988). The method, carried out manually, was to take the source legal text, identify the relevant textual portions, decompose and paraphrase them as necessary, and then formalise the language in an executable logic such as Prolog, creating an *expert system*. From this formalisation, ground facts may be provided to the system which are then used to draw inferences and the rule system could be tested for consistency. Translation of the *British Nationality Act 1981* was one such exercise. The first clause is as follows. It is stated in the act that "after commencement" means on or after the date when the act comes into force.

> l.-(l) A person born in the United Kingdom after commencement shall be a British citizen if at the time of birth his father or mother is (a) a British citizen; or (b) settled in the United Kingdom.

The clause is translated into Prolog as:

```
is_a_British_citizen(X) :-
    was_born_in_the_U.K.(X), was_born_on_date(X,Y),
    is_after_or_on_commencement(Y),
    has_a_parent_who_qualifies_under_1.1_on_date(X,Y).
```

This is a "first" draft translation, and the literature discusses a range of issues that must be addressed such as dependencies between subsequent portions of text, the introduction of negation, drawing out implicit information, the complex structure of the clauses. An overall point is that the axioms of the legislation are formulated from the source by a

methodology of trial and error; that is, there is no systematic or automated analysis of the natural language text. Not only does this make the analysis expensive to produce and maintain, but does not facilitate reuse as each predicate is *sui generis* rather than composed from linguistic modules. Nonetheless, the translation provides a *gold standard*: in an interactive environment, a user queries the system, answers questions, and receives determinations and explanations.

3. Manually Paraphrased and Automatically Translated

Since this early work, some commerical products have become available which support aspects of this process and serve the resultant expert systems to users on the web (Johnson and Mead, 1991), (Dayal et al., 1993), (Dayal and Johnson, 2000). In particular, Oracle Policy Management can take rules from legislation in natural language and automatically translate them into a logic; an inference engine is applied to grounded statements, providing determinations; there is a web-interface to serve the system statements to users. Explanatory notes, document access, and alternative evaluations are auxiliary capabilities. It has been applied to the examples discussed in (Sergot et al., 1986) and many other acts; it is in widespread use by government agencies in the United Kingdom and United States, e.g. for tax calculation and citizen benefits.

While this is a very significant development, its overall contribution is limited in two respects. First, the methodology of analysis, though industrialised, remains largely that of trial and error: the source text is analysed manually, scoped, and paraphrased *in a controlled natural language (fixed grammatical constructions) to meet the constraints of the parser and semantic interpreter*; the system uses "just enough" natural language processing to satisfy the clients' requirements. Second, as a proprietary product, the system is restricted in exposure, use, and development.

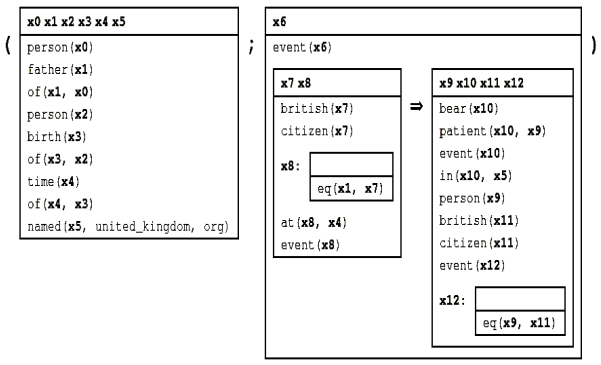
As an alternative, *Attempto Controlled English* (ACE) is a controlled language (Fuchs et al., 2008; Wyner et al., 2009) that has been applied in some small, non-legal domains (Shiffman et al., 2010; Wyner et al., 2010). As a controlled language, any source text must be paraphrased to fulfil the specifications of the language. Given the sentence, the system automatically parses and semantically represents an unambiguous semantic formula in *Segmented Discourse Representation Theory* (SDRT), which can be input to an inference engine. As discussed in (Wyner et al., 2010), matters are not straightforward for one must carefully evaluate whether the output semantic formula accurately represents the intended meaning of the input sentence, adjusting the input sentence accordingly. It also remains to be seen whether the parser and semantic interpreter can process the sorts of sentences we find in legal texts. ACE is not yet associated with an expert system interface. Despite these issues, the system has several advantages: the input is in natural language, the result is a single, unambiguous semantic translation, the system is open source and extensible, it has a web interface, and it is already highly flexible.

4. A Fully Automatic System

A more expressive, flexible, and powerful natural language processing system is C&C/Boxer (Bos et al., 2004); it has extensive, efficient parsing using categorial grammar and a translation to SDRT; while it may be controlled, it is not constrained to be so. It generates the most *likely* parse and semantic representation, requiring analysis of the results, selection among the alternative analyses, or modification of the input till one gets the intended representation.

We illustrate the output of the tool with a paraphrase of our example legislative clause to show the results and the issues. For example, *A person born in the United Kingdom shall be a British Citizen if at the time of his birth his father is a British citizen.* has the output representation in Figure 1. We found we had to make explicit the implicit pronominal relationships and also the gender. The semantics does not specify the relation between the man and time of birth or between the man and the father. And we note that we have only represented a small fragment.

Figure 1: SDRT of Sample Statement



Even from this small sample, we can see that the problems of manual translation have shifted from initial analysis to evaluation of output - the discourse elements and predicates. The first problem is that x0 (the person with the father) and x2 (the person with the birthday) are possibly distinct. In the antecedent of the conditional, x1 (the father) is a British Citizen at the time of the birth of x2. But, x2 need not be identical to the person who is the son of that citizen. Finally, in the conclusion, x9 gets British Citizenship, but is not identified with either x0 or x2.

Despite these problems, automated systems are systematic, grounded, and open to refinement. Besides continued evaluation of output, automated systems need testing suites to support evaluation, especially where large, complex expressions and documents are concerned. Furthermore, the parser and semantic interpreter must developed and refined to meet the requirements found in legal textual language.

5. Conclusion

In this position paper, we have briefly presented three main approaches to semantic representation of legislative documents that can be used for automated inference. Some of their problems and prospects have been outlined with the intention that the observations can be used to develop more sophisticated systems.

6. Acknowledgements

The author was supported by the FP7-ICT-2009-4 Programme, IMPACT Project, Grant Agreement Number 247228. The views expressed are the author.

7. References

Trevor Bench-Capon, George Robinson, Tom Routen, and Marek Sergot. 1987. Logic programming for large scale applications in law: A formalisation of supplementary benefit legislation. In *Proceedings of ICAIL '87*, pages 190–198. ACM.

Johan Bos, Stephen Clark, Mark Steedman, James R. Curran, and Julia Hockenmaier. 2004. Wide-coverage semantic representations from a CCG parser. In *Proceedings COLING '04*, pages 1240–1246, Morristown, NJ, USA. Association for Computational Linguistics.

Surend Dayal and Peter Johnson. 2000. A web-based revolution in australian public administration. *Journal of Information, Law, and Technology*, 1. Online.

Surendra Dayal, Michael Harmer, Peter Johnson, and David Mead. 1993. Beyond knowledge representation: commercial uses for legal knowledge bases. In *Proceedings of ICAIL '93*, pages 167–174. ACM.

Norbert E. Fuchs, Kaarel Kaljurand, and Tobias Kuhn. 2008. Attempto controlled english for knowledge representation. In Cristina Baroglio, et al., editors, *Reasoning Web*, pages 104–124. Springer.

Peter Johnson and David Mead. 1991. Legislative knowledge base systems for public administration: Some practical issues. In *Proceedings of ICAIL '91*, pages 108–117. ACM.

Diana Maynard and Mark Greenwood. 2012. Large scale semantic annotation, indexing and search at the national archives. In *Proceedings of LREC '12*, Istanbul, Turkey, May.

Marek Sergot, Fariba Sadri, Robert Kowalski, Frank Kriwaczek, Peter Hammond, and Therese Cory. 1986. The British Nationality Act as a logic program. *Communications of the ACM*, 29(5):370–386.

Marek Sergot. 1988. Representing legislation as logic programs. *Machine Intelligence*, pages 209–260.

Richard N. Shiffman, George Michel, Michael Krauthammer, Norbert E. Fuchs, Kaarel Kaljurand, and Tobias Kuhn. 2010. Writing clinical practice guidelines in controlled natural language. In Norbert E. Fuchs, editor, *Proceedings of the 2009 conference on Controlled Natural Language*, pages 265–280. Springer.

Adam Wyner and Wim Peters. 2011. On rule extraction from regulations. In Katie Atkinson, editor, *Proceedings of JURIX '11*, pages 113–122. IOS Press.

Adam Wyner, Krasimir Angelov, Guntis Barzdins, Danica Damljanovic, Brian Davis, Norbert E. Fuchs, Stefan Höfler, Ken Jones, Kaarel Kaljurand, and Tobias Kuhn. 2010. On controlled natural languages: properties and prospects. In Norbert E. Fuchs, editor, *Proceedings of the 2009 conference on Controlled Natural Language*, volume 5972 of *Lecture Notes in Computer Science*, pages 281–289. Springer.

Adam Wyner, Tom van Engers, and Kiavash Bahreini. 2010. From policy-making statements to first-order logic. In Kim Normann Andersen, Enrico Francesconi, Åke Grönlund, and Tom M. van Engers, editors, *Proceedings of EGOVIS '10*, pages 47–61. Springer.