An Advisory System for Open Textured Law.

Uri J. Schild

Department of Computing, Imperial College, London, U.K.

Subject Categories:
Legal Systems, Decision-Aids, Rule-Based Systems, Reasoning by Example.

Background:
Many phrases in legal language, like e.g., "ordinary use" identify a variable standard. The choice of such words by the legal expert is an acknowledgement that there is usually room for discretionary judgment in deciding whether the standard has been met. The meaning is so dependent on individual circumstances, that no legislator, for example, could hope to foresee every possibility and make provision for it. Such concepts are called: open-textured.

In the Anglo-American Legal System, when such an open-textured concept appears in a particular instance, the applicability of the concept is determined by the appropriate court, based on previous decisions in relevant precedents (case-law). Thus open-texturedness is vagueness plus decision-scheme.

Closely related to the concept of open texture is the concept of easy and hard cases. Using an over-simplification we shall say that a case is easy, if all legal experts agree on its outcome. Otherwise a case will be called hard. It is not always true that a case involving open-textured concepts is hard.

Systems Requirements:
Our interest lies in developing an expert system based on and taking into account the fundamental open-texturedness of law, specifically case-law. Such a system may generate conflicting answers in many cases. Its role is to point out the different possible approaches to the legal problem at hand and the contradictory conclusions that may be reached. It must be able to supply explanations for its conclusions, i.e. supply the chain of legal argumentation and reasoning behind its conflicting answers. An advisory system of this kind could be a decision-making aid for a lawyer or a judge.

The central problems of developing such a system may be specified as follows:
(1) When considering a concrete case, how does the system know whether it is 'easy' or 'hard'?
(2) If indeed a case is 'hard' what advice should the system give its user?

Other Work:
Expert Systems based on case-law have been described in [1,2,3]. [4] deals with legal reasoning and open texture from a theoretical standpoint. Our present work is mainly based on the ideas in [5]. [6] gives a comprehensive jurisprudential background for legal expert systems.
Implementation:
The area of law we have chosen deals with common law, private nuisance which is a kind of tort. This particular area is all about balancing one person's interests against his neighbour's and is full of open-textured concepts like 'ordinary use' and 'reasonable excuse'. A problem which previous researchers have come up against is the large amount of common-sense knowledge needed in many areas of law. In the area of nuisance this problem appears to be of moderate extent only.

Our present system uses a rule-based representation of the case-law of nuisance, derived from the authoritative text-book by Salmond [7]. This book has not only been through eighteen editions, but it is cited by both counsel and judges in numerous cases in England and the Commonwealth.

The Prolog program consists of rules like the following:

nuisance if    appropriate-length-of-time
and         sensible-damage
and         appropriate-situation-of-plaintiffs-premises
and ......

Each of the antecedent predicates is itself defined by similar rules. In general the system is very shallow, not exceeding four levels. The reason for this is that the open-textured concepts are represented here as 'hyphenated predicates'. Rules for defining open-textured concepts may in principle be given ad infinitum. Our choice of cut-off was guided by the level of refinement actually used in court.

The system elicits the facts by querying the user, who is assumed to have a legal background. At each level he answers yes/no/maybe. Before answering the user may request to see relevant argumentation and court-decisions. He may also ask for full retrieval of case-reports. If the user answers 'maybe' the system automatically descends to a deeper level of inquiry (if such one exists).

For easy cases, i.e cases where all the user's answers at the final levels were yes or no, the system will give a definite answer: nuisance or not-nuisance. For hard cases, i.e. cases where final answers of maybe were given, the output of the system at present is a summary of which questions were not resolved, and the (contradictory) argumentations and court-decisions relevant to those questions. The system may therefore be used as an issue-spotter and as an aid in preparing lawyers' briefs. Our present work looks into the possibility of automatical weighing of such arguments against each other.

Acknowledgement: My thanks are extended to M.J. Sergot for his helpful advice and constructive criticism.
References:
    LRP-TR-13, Rutgers University, 1982.