

Issues in Natural Language Engineering for Case-based Legal Argument Instruction

A Position Statement for the Jurix 2008

Workshop on the Natural Language Engineering of Legal Argumentation:
Language, Logic, and Computation

Kevin D. Ashley

Professor of Law and Intelligent Systems

Senior Scientist, Learning Research and Development Center

University of Pittsburgh

ashley@pitt.edu

Argument

- Viewpoint: designers of instructional systems for teaching law students skills of legal argumentation
 - Empirical analysis, formalization, and implementation of legal argumentation in natural language in an educational context.
- Our approach to argument instruction:
 - Implement computational model of case-based legal argumentation
 - Use to engage law students in making and responding to arguments.
 - Typically, arguments of the form:
 - assertion that problem should be decided in accord with a retrieved case
 - justification based on drawing a factor-based analogy between problem and case
 - supplement with reasons why shared factors matter and
 - unshared factors either support, or do not seriously detract from, the assertion.
 - responses = distinguishing the problem from the case; citing other cases as counterexamples.
- Pragmatic approach to natural language generation:
 - associating scripts or “canned” text fragments with factors and reasons
 - flexibly to support some composition as naturally as possible without repetition, etc.
 - Forgo natural language inputs;
 - lists of user responses prepared dynamically and presented in menus.
 - flexible enough to facilitate presenting the same arguments in different contexts, for instance, a court room setting (Ashley, et al., 2002).

Three Issues for CBLA Instruction

1. *Synthesis of multiple cases into arguments:*

- CATO program generated multi-case arguments organized by issues
- Not to the standard of legal writing instructors
- Arguments did not effectively synthesize the cases into reasons supporting a proposition
- Need to characterize cases at appropriate level of abstraction, highlighting shared aspects of their often divergent fact situations in a manner salient to the proposition, and explaining away certain differences as insignificant

2. *Deciding how abstractly to characterize similarities and differences:*

- CATO used algorithmic approach to deciding how abstractly to characterize factual differences among cases in terms of the Factor Hierarchy's abstract factors and issues depending on the argument context
- Factor Hierarchy was a graph; this often entailed making choices of which abstraction path to follow.
- How to adapt when hierarchies extended to include principles and policies underling the legal domain that inform reasons why similarities and distinctions matter?

3. *Integrating proposed tests and underlying principles/policies into arguments:*

- Legal arguers propose rules or tests for deciding a case that must accomplish an advocate's desired result, be consistent with prior cases, and embody an acceptable trade off among conflicting underlying principles/policies.
- Typically by evaluating proposed tests by posing hypothetical cases to explore the test's meaning or critique it as too broad or narrow.
- Suggests new challenges for flexibly supporting case comparisons, reasons why they matter, and modifying tests.

Best way to address these issues?

- systematic, empirical study of real legal arguments
- distill general methods for synthesizing multiple cases, determining how and how abstractly to characterize cases, similarities, and differences, and integrating proposed tests, principles, and policies into arguments.