Modeling Legal Argument: Reasoning with Cases and Hypotheticals
KEVIN D. ASHLEY, 1990
Cambridge, MIT Press
329 pp ISBN 0-262-0111Y-X $39.95

A busy attorney interviews a client. After an exchange of pleasantries, the attorney gathers facts by asking a series of questions that appear on the monitor of a computer, into which the client's responses are entered. The client's story is stored and analyzed by the machine itself. At some later time, an attorney (not necessarily the same one) presses a key and receives a printed copy of an analysis of the client's legal situation, complete with the best arguments that can be made on the client's behalf by citing cases most favorable to him and neutralizing the cases most favorable to his opponent. Thus armed, the attorney strides confidently into the courtroom to do battle.

In an age awed and even besotted by the possibilities of the computer, this Tomorrowland scenario may seem tame, even trite. But lawyers—especially legal academicians—of a century ago may have been readier to accept this story as stating a possibility than their present-day counterparts are. A century ago, law was viewed as a science, and as capable of rigor, precision, and predictive power as chemistry and physics. Positive law expressed an immanent legal reality and, as such, was as capable of formalization as any branch of natural science. In the twenties and thirties of this century, however, the academic lawyers known as Legal Realists mounted an attack upon what they termed this "mechanical jurisprudence". The Legal Realists attacked the very idea that judges found and followed legal rules, and emphasized the vast reaches of empty space within any apparently solid legal object, and the concomitantly vast degree of discretion exercised (knowingly or not) by any judge purporting to follow rules already laid down. The attack was so powerful that even today the Legal Realists' insights are components of any sophisticated conception of law. And thus, ironically, even as new worlds of computational possibility open out, legal theory insists that they can't be made much use of.

Kevin Ashley, who holds a joint appointment in the Learning Research and Development Center and the School of Law at the University of Pittsburgh, has joined a small but growing band of researchers who have entered where most modern legal theorists disdain to tread. Ashley has designed and implemented ("partially", anyway) a computer program called Hypo, whose goal is—if not to mimic legal reasoning—to produce an output "just as intelligent" (p. 1). Ashley makes two striking and sure-to-be-controversial claims based upon this effort. The first, practical, claim is that Hypo forms the basis of an approach that will lead to systems superior to key-word searches of full-text databases of legal authorities, which constitute the principal use of computers in legal work currently. The second, programmatic, claim is that modelling legal reasoning is the key to the study of informal reasoning generally. For Ashley, law is a midpoint between domains such as mathematics, in which deductive models may be achievable, and common-sense reasoning, in which they are not. As an "organized chaos" (p. 2), not susceptible to a deductive model, law should appeal to Artificial Intelligence ("AI") researchers as an entree to the realm of practical reasoning. (p. 6).

The Hypo program focusses upon a species of legal reasoning that Ashley terms "adversarial case-based reasoning" (p. 3). Adversarial case-based reasoning occurs outside legal contexts, but it is particularly prevalent in law. Legal reasoning is "adversarial" because of the adversarial nature of legal procedure. It is "case-based" because of the rule of stare decisis ("let the decision stand"), which requires a court to give controlling weight to prior cases sufficiently like the case it must decide. One must say "sufficiently like"—rather than "falling under the rule of"—for two reasons. One reason is that authoritative texts have to be interpreted before they can be said to stand for some rule. The second is that a precedent case cannot unambiguously lay down any single rule.

A precedent case cannot do this, Ashley argues, for two further reasons. The first is that under the common law "system" a court has no authority to decide any but the case before it and therefore has no proper power to state a major premise binding on later courts. The second is that even where later courts interpret precedents in order to formulate major premises for their own deductive efforts, they are confronted with a choice between many alternative and conflicting formulations, and must choose without further guidance (p. 217). Accordingly, Ashley believes he must represent a precedent case as its facts (or a selection thereof) plus its outcome, and reasoning with cases as essentially analogical. Thus, Ashley hopes to avoid the pitfalls of "mechanical jurisprudence" in the deductive sense while, at the same time, bringing computational methods to bear in a way that will illuminate the nature of analogical reasoning.

Schematically, using Hypo to analyze a case involves no more than having a (human) "case enterer" specify on a menu the values that certain "factual predicates" take in the case at hand. Hypo generates its own internal representation of the problem case and then compares this to each of the similarly schematized precedent cases that comprise its "Case Knowledge Base", which consists of 30 trade secrets cases, most (but not all) of which are derived from actual, reported cases. The "made-up" cases were presumably created to exemplify legal doctrine that otherwise would have to be stated as a rule. In effect, the program matches the problem case with precedents that present identical "factors", which are the computational expression of the idea of "relevant similarity". The program collects and ranks the precedent cases according to the number of "factors" they share with the problem case.

Finally the program spits out (1) a summary of the client's most helpful precedent cases, (2) "three-ply" arguments citing these cases in the first instance or in rebuttal to anticipated arguments for the other side, and (3) hypothetical modifications of the facts that would strengthen the client's position (p. 20).

Ashley claims that, unlike "key word" searches devised *ad hoc* by an attorney, "with Hypo, the program performs the analysis" (p. 22), and as such "is a promising prototype of an automated "brief writer's assistant" (p. 193). Upon examination, however, Hypo appears to accomplish much less than Ashley suggests, either practically or programmatically.

Where do the "factors" crucial to the program come from? Ashley tells us that these "have been gleaned from legal treatises and law review articles" (p. 24). This might seem perfectly natural to non-lawyers, but in fact this approach inverts the way lawyers look at precedent. Treatises and articles are not binding authority in courts, and whatever persuasive authority they are accorded is largely derivative from the precedents they discuss. The fact that the program's "factors" are drawn from secondary sources and not from the cases directly represents a drastic, if unavoidable, weakness, as Ashley himself admits.

The program itself cannot recognize the presence or absence of "factors" in a natural-language text—no existing program can. Input must be therefore be cast in a "specially designed language for representing legal disputes" (p. 25). As it happens, though, *casting* what a client says into legal language is 90% of a lawyers' job. In trade secrets law, for example, a plaintiff will want to show that it guarded the secret it claims the defendant has violated. Thus, measures taken to secure the secret count, *ceteris paribus*, for the plaintiff, and disclosures against. Accordingly, the "factors" programmed into "Hypo" include "security measure" and "disclosure." But who is to decide whether an alarm system a "security measure"? Or whether a sale is a "disclosure"? Although Ashley claims that each factual predicate has "an attached retrieval method" to test whether the predicate applies" (p. 99), in fact, these decisions fall to the "case entry person", who seems to be pulling quite a bit of the weight. If "context" is being accounted for, as Ashley claims (p. 6), and if peculiarly analogical reasoning is going on, it seems to fall to the "case entry person" to do it.

It is important to Ashley that "[a] legal precedent [be] treated as a collection of factors ..." One reason has to do with the jurisprudential scruples about judicial lawmaking, but in Ashley's case there is the additional reason that "Hypo's Case Representation language does not record the court's rationale for deciding a case ... [r]ecording this kind of information would require a considerably more expressive language, complicate the computational process of reasoning with cases and make the human enterer's task much more subjective ... [t]he enterer is required to determine from the opinion only the facts of the case. He or she is not required to make legal interpretations of the case" (p. 105). This is an outrageously silly and demonstrably false claim, but I will not dwell on that, because the more interesting issue is: However we characterize what the "case entry person" has done, what can we say the machine is doing with its input?

"Factors" feed into what Ashley unhappily terms "Dimensions". The two are related in the following way: "Factors are a conceptual link between various clusters of operative facts that may appear in legal disputes and the conclusions they support or undermine. Dimensions allow a computer program to reason with those links" (p. 38). The list of thirteen implemented "Dimensions" look very much like rules of inference, which warrant conclusions about the "strength" of one side or another's case. Although the program's

"Argument Evaluation Criteria" rank cases in order of strength for a side, Ashley claims they are deliberately too weak to say which side has the "better" argument. But it is hard to see how Hypo improves on rule-based models since the Dimensions obviously express at least defeasible rules of inference; and "Hypo" implements a simplistic "dominance" theory—party *p* "wins" (but let's not call it that!) if the factors antecedently identified as party-*p*-favoring predominate.

It may be that effective "expert systems" in law will someday be a reality. If and when they are, those who use them will not care whether they work by smuggling in rules and by relying crucially upon the human operator's judgment. Practical utility is the test of progress here, not the degree of jurisprudential scrupulousness. Ashley's work, which is an important incremental step, tends to take the latter too seriously and to be too anecdotal about the former. Let's hope that a later version of Hypo ships soon.

WILLIAM A. EDMUNDSON, *Georgia State University*.