



TP18
G 226

8860848

**An Artificial Intelligence
Approach to Legal
Reasoning**

Anne von der Lieth Gardner



E8860848

A Bradford Book
The MIT Press
Cambridge, Massachusetts
London, England

Excerpts that appear in chapter 5 and figures 7.8 and 7.9 were taken from *Gilbert Law Summaries: Contracts*, 10th Edition, edited by Melvin A. Eisenberg, copyright © 1982 by Harcourt Brace Jovanovich Legal and Professional Publications, Inc. Reprinted by permission of the publisher.

© 1987 by The Massachusetts Institute of Technology

All rights reserved. No part of this book may be reproduced in any form by any electronic or mechanical means (including photocopying, recording, or information storage and retrieval) without permission in writing from the publisher.

This book was set in Palatino by DEKR Corporation and printed and bound by Halliday Lithograph in the United States of America.

Library of Congress Cataloging-in-Publication Data

Gardner, Anne von der Lieth

An artificial intelligence approach to legal reasoning.

(Artificial intelligence and the law: processes and models of legal reasoning series)
"A Bradford book."

Bibliography: p.

Includes index.

1. Information storage and retrieval systems—Law—United States. 2. Law—United States—Methodology. 3. Artificial intelligence. I. Title. II. Series: Artificial intelligence and the law.

KF242.A1V36 1987

025'.0634

86-10469

ISBN 0-262-07104-5

**An Artificial Intelligence
Approach to Legal
Reasoning**

Artificial Intelligence and Legal Reasoning

L. Thorne McCarty and Edwina L. Rissland, editors

An Artificial Intelligence Approach to Legal Reasoning
Anne von der Lieth Gardner, 1987

For my father, who has always
encouraged and never pushed



Series Foreword

The law, with its diverse modes of reasoning, provides a rich area for the study of both human and artificial intelligence. The reasoning can involve cases, rules, or text. Cases can be real or hypothetical; rules, well- or ill-formulated; text, free or highly structured. And the reasoning can be deductive, inductive, or analogical. It is a perfect arena in which cognitive scientists and artificial intelligence researchers can merge interests.

Although jurisprudence has traditionally grappled with problems of legal reasoning, it has not been able to describe them at the level of specificity that artificial intelligence and cognitive science strive for. Armed with new techniques of knowledge representation, procedural specification, and the like, artificial intelligence researchers now believe that a more detailed analysis of legal reasoning is possible. But artificial intelligence, which has had some practical successes in areas like medicine and geology, has not yet had much empirical experience in a domain with such hybrid modes of thought as the law. The law challenges AI on all fronts, from natural language processing to computational architectures combining rule- and case-based paradigms.

A joint endeavor in artificial intelligence and law will strengthen and inform both disciplines.

L. Thorne McCarty
Edwina L. Rissland

Acknowledgments

This book is a revision of a dissertation submitted to the Department of Computer Science of Stanford University in June 1984. For the opportunity to pursue the dissertation, I owe thanks to many people. I am grateful to the members of my reading committee, Terry Winograd, Bruce Buchanan, and Doug Lenat. As my principal adviser, Terry provided good advice, encouragement, and freedom to explore in the directions that seemed to me most important.

Helpful advice and a supportive working atmosphere were available both within and without Stanford's Heuristic Programming Project (now the Knowledge Systems Laboratory). Besides the members of my committee, I wish to thank Ed Feigenbaum, Mike Genesereth, John McCarthy, and Richard Weyhrauch. For guidance through undocumented mysteries of Maclisp and MRS, special thanks to Dick Gabriel and Andy Freeman. Outside Stanford, I have received valuable comments from Jerry Hobbs, Thorne McCarty, and Edwina Rissland.

Members of the Stanford Law School faculty have also been of great help in the course of this work. The late Joseph Leininger was one of the first to see the potential for law in AI. J. Keith Mann, who knew me as a law student, has helped pave the way for many things. Currently Paul Brest, Tom Heller, and Bob Mnookin are making Stanford an exciting place to work on computers and law.

Financial support for this work was provided, during 1979–81, by a fellowship from IBM's Thomas J. Watson Research Center in Yorktown Heights, New York. Support during three summers was provided through a grant from IBM to the Stanford Law School. The revised version of the dissertation was prepared at Stanford's Center for the Study of Language and Information, through the courtesy of John Perry and Terry Winograd.

Contents

List of Figures	ix
Series Foreword	xi
Acknowledgments	xiii

1	Introduction	1
1.1	Some Aspects of Legal Reasoning	1
1.2	Choosing a Legal Reasoning Task	4
1.3	A Preview	11
2	Design I: The Place of Rules	17
2.1	Introduction	17
2.2	Legal Theories and AI Paradigms	18
2.3	Rules for Contract Law	24
3	Design II: When the Rules Run Out	33
3.1	The Problem of Applying a Rule	34
3.2	The Problem of Hard and Easy Questions	38
3.3	Heuristics for Resolving Easy Questions	42
3.4	Representing and Using Cases	47
3.5	Approaches to Hard Questions	59
4	Related Work	67
4.1	Legal Theory and Computer Programs	67
4.2	Rule-based Programs	70
4.3	Example-based Programs	74
4.4	More General Programs	76
4.5	A Misuse of Programs	81

5	Representing Problems	85
5.1	Representation Issues	85
5.2	Basic Elements of the Representation	88
5.3	Knowledge about Domain Entities	98
5.4	Representing Reported Speech	103
6	Representing and Using Legal Knowledge	119
6.1	Scope of the Knowledge Base	121
6.2	A Transition Network for the Sequence of Legal Events	123
6.3	Legal Rules: Organization and Structure	132
6.4	Legal Rules: Antecedents	138
6.5	Examples	155
7	Program Performance	163
7.1	Analyzing an Event and Summarizing the Analysis	163
7.2	Full Analysis of the Salt Problem	175
7.3	Analyses of Other Problems	183
8	Conclusion	189
	Notes	195
	References	201
	Index	215

List of Figures

- Figure 1.1 Upper level of an output graph 10
- Figure 1.2 Simplified transition network 13
- Figure 5.1 Elements of the abstraction hierarchy: part 1 90
- Figure 5.2 Elements of the abstraction hierarchy: part 2 91
- Figure 5.3 Representing a particular sale of a carload of salt 99
- Figure 5.4 Initial paraphrases 111
- Figure 5.5 Representing the propositional content of a sentence proposing a sale of a carload of salt 112
- Figure 5.6 Representing "Accept your offer carload of salt" 114
- Figure 5.7 Representing "immediate shipment, terms cash on delivery" 115
- Figure 5.8 Representing "B mailed a rejection" 116
- Figure 5.9 Representing "This offer will remain in effect for only 5 days" 117
- Figure 6.1 Transition network for offer and acceptance problems 124
- Figure 6.2 Arcs of the transition network 125
- Figure 6.3 Data structures initialized to test what new state transition best characterizes the current event 131
- Figure 6.4 Definitions of an offer 142
- Figure 6.5 Branching possibilities for an antecedent of the form (or $(P A)$ (not $(Q B)$)) 149
- Figure 6.6 Structure of MRS theories showing alternate interpretations of events that have been fully processed 151
- Figure 6.7 Relation of upper-level output graph to MRS theory structure 151
- Figure 6.8 Examples of indicating willingness to enter an exchange 157
- Figure 7.1 Detailed analysis tree for event 1 165
- Figure 7.2 Context interpreting event 1 as an offer 169
- Figure 7.3 Context interpreting event 1 as not an offer 170
- Figure 7.4 Detailed analysis of event 5 as a response to a counteroffer 173
- Figure 7.5 Condensed version of summary-level decision tree 176
- Figure 7.6 Analyses 1 through 4 of test problem 178
- Figure 7.7 Analyses 5 through 9 of test problem 179
- Figure 7.8 Additional problems 183
- Figure 7.9 Published answers to the additional problems 184

Legal reasoning is a major topic in the philosophy of law, but it is only beginning to receive widespread attention in artificial intelligence. The challenge, for AI, is to produce a description of legal reasoning that reflects its characteristic features and is, at least for appropriate subproblems, computationally well defined.

Jurists have long given noncomputational descriptions of legal reasoning. How far the accounts could be made computational has been the subject of deep, if inexplicit, debate. Some computer-oriented work, on the other hand, has produced models made computational by suppressing just those aspects of legal reasoning that jurists think important.

The objective of this study is to create a model of the legal reasoning process that makes sense from both jurisprudential and AI perspectives.

1.1 Some Aspects of Legal Reasoning

As a task for AI programs, legal reasoning brings together two areas usually treated as distinct. One is research on expert systems (Feigenbaum 1977; Buchanan 1982; Hayes-Roth, Waterman, and Lenat 1983); the other, natural-language understanding and commonsense reasoning (Schank and Abelson 1977; Winograd 1980a; McCarthy 1984; Hobbs and Moore 1985).

The expert-systems area is obviously relevant since a legal reasoning program requires substantial professional knowledge. But a good computer model of the legal domain will need several capabilities not provided by previous work on expert systems. These include (1) distinguishing between questions the program has enough information to resolve and questions that competent professionals could

argue either way, (2) using incompletely defined technical concepts, and (3) combining the use of knowledge expressed as rules and knowledge expressed as examples. All are important in other domains besides law, but past AI research has left them largely unexplored.

The natural-language aspect of the project is present partly because of the particular legal subdomain selected: the formation of contracts by offer and acceptance. In problems about offer and acceptance, the data to be interpreted consist mostly of reported dialogue.

But there is a deeper reason for the natural-language aspect of legal analysis. This reason is the incomplete definition, or open texture, of many legal predicates. More accurately, *open texture* refers to the inherent indeterminacy of meaning in the words by which fact situations are classified into instances and noninstances of legal concepts (see Hart 1961, pp. 121–132). It applies equally to legal subdomains such as assault and battery (Meldman 1975), corporate taxation (McCarty 1977; McCarty and Sridharan 1982), and manufacturers' product liability (Waterman and Peterson 1981, 1984), as well as contract law. Although the work described here deals only with open texture in law, the concept has much wider importance. The term was coined in philosophy and used originally of words like *dog* and *gold* in pointing out that most empirical concepts are not delimited in all possible directions (Waismann 1945).¹ Recent analyses of such natural-kind words, and other sorts of words too, have involved closely related observations (Putnam 1975; see generally Schwartz 1977). There is at least a kinship with Wittgenstein's idea of family resemblances (1958 ed.). In AI, Winograd's discussion of the word *bachelor* (1976) is an exploration of open texture.

In this book I will propose a framework within which to provide for open texture and for the other capabilities listed. I will also describe a program based on this framework. The design of the program is intended to reflect lawyers' own understanding of the nature and uses of legal materials—in other words, to accord with a legally plausible conceptualization of the domain. The result is a conceptual analysis of legal reasoning, not a psychology. In terms of a contrast drawn by Newell and Simon (1972, p. 55), the focus is on the structure of the legal task environment, as distinguished from the psychology of a problem-solving agent. The environment has the following distinctive features.

First, legal rules are used consciously by the expert to provide guidance in the analysis, argumentation, and decision of cases. In this respect the rules are unlike those used in most expert systems or the rules of a grammar, which seek to describe behavioral regularities of which the expert or native speaker may be unaware. Legal reasoning might thus be classified as a *rule-guided activity* rather than a rule-governed activity.

Second, and as a consequence, the experts can do more with the rules than just follow them. In a field like contract law, where the rules have been developed mainly through decisions in individual cases, lawyers can argue about the rules themselves and can propose refinements, reformulations, or even newly formulated rules to adapt the law to a particular case at hand. Sometimes, it is true, the rules may be taken as fixed. In a case law field, like contracts, they can become fixed by long acceptance; in a statutory field, like taxation, they are fixed by legislative enactment. Even with the simplification of fixed rules, lawyers are free to argue about what counts as following the rules in a particular case.

Third, lawyers are not merely free to disagree; on hard questions of law they are expected to do so. Unlike other domains, in which writers of expert systems hope for consensus among the experts, the legal system makes institutional provision for expert disagreement—for instance, in the institutions of opposing counsel, dissenting judicial opinions, and appellate review of lower court decisions.

Fourth, since lawyers are expected sometimes to disagree, the following question arises: Is there any class of cases as to which all competent lawyers would reach the same conclusion? In the legal literature, the problem is stated, but not solved, in terms of whether a dividing line between hard cases and clear cases can be found (Hart 1958; Fuller 1958; M. Moore 1981). Despite the lack of a theoretical solution, most cases are in fact treated as raising no hard questions of law. (Whether they raise hard questions of fact is another matter.)

Fifth, when hard legal questions do arise, their basis is quite different from the sources of uncertainty usually described in connection with expert systems. They do not generally involve insufficient data, for example, or incomplete understanding of the workings of some physical process. Instead, an especially important source of hard questions is the open texture of legal predicates.

Finally, there is the task of resolving legal questions, hard or easy. How does the judge carry out this task? How should the task be

done? Once it is done, how should the judge justify the results in a written opinion? These questions—often not distinguished from one another—are central in legal philosophy. Different writers, all intimately familiar with the judicial process, paint rather different pictures of it (Cardozo 1921; Levi 1949; Llewellyn 1960; Hart 1961; Dworkin 1977a). They agree on this much: in a well-developed, relatively stable field of law (like contracts), at least two distinct knowledge sources must be brought to bear. Legal rules are one; and rules exist even in a nonstatutory field (like contracts) where they lack official wording. (For an influential unofficial attempt to state the rules of contract law, see *Restatement of Contracts*, 1932, and *Restatement of Contracts*, Second, 1981.) Second, there are decisions in previous cases. There is no tidy consensus about how the rules and the precedents are used together.

These domain characteristics dictate the main features of the program. The overall objective is not a program that “solves” legal problems by producing a single “correct” analysis. Instead, the objective is to enable the program to recognize the issues a problem raises and to distinguish between those it has enough information to resolve and those on which competent human judgments might differ. Toward this end a heuristic distinction between hard and easy questions is proposed. The distinction in turn draws on ideas about how rules and examples interact and how their interaction allows for open texture.

1.2 Choosing a Legal Reasoning Task

1.2.1 The Input

To provide a definite context for studying legal reasoning, this study uses materials classically taught by the case method in law schools and classically tested by asking the student, given the facts of a new case, to analyze their legal consequences. The specific legal topic is a standard one for first-year law students, the formation of contracts by offer and acceptance. A typical examination question is the following:

On July 1 Buyer sent the following telegram to Seller: “Have customers for salt and need carload immediately. Will you supply carload at \$2.40 per cwt?” Seller received the telegram the same day.

On July 12 Seller sent Buyer the following telegram, which Buyer received the same day: "Accept your offer carload of salt, immediate shipment, terms cash on delivery."

On July 13 Buyer sent by Air Mail its standard form "Purchase Order" to Seller. On the face of the form Buyer had written that it accepted "Seller's offer of July 12" and had written "One carload" and "\$2.40 per cwt." in the appropriate spaces for quantity and price. Among numerous printed provisions on the reverse of the form was the following: "Unless otherwise stated on the face hereof, payment on all purchase orders shall not be due until 30 days following delivery." There was no statement on the face of the form regarding time of payment.

Later on July 13 another party offered to sell Buyer a carload of salt for \$2.30 per cwt. Buyer immediately wired Seller: "Ignore purchase order mailed earlier today; your offer of July 12 rejected." This telegram was received by Seller on the same day (July 13). Seller received Buyer's purchase order in the mail the following day (July 14).

Briefly analyze each of the items of correspondence in terms of its legal effect, and indicate what the result will be in Seller's action against Buyer for breach of contract.

To define an adequate internal representation for such questions is itself a substantial research task. The representation used will be discussed in chapter 5. The encoding of English into this representation is done by hand. Giving an algorithm to encode questions automatically would involve further difficulties, some of which will be mentioned in chapter 5.

The Choice of Offer and Acceptance Problems

As the legal area to be studied, offer and acceptance problems have several features in common with Meldman's (1975) choice of assault and battery: the law is reasonably well settled; it is taught early in the first semester of law school, so that not much other legal knowledge is prerequisite; and it is an area dominated by case law, not statutes, so that legal reasoning is used in its most traditional and most studied forms.

It might seem that a statutory field, like taxation, would be easier for an AI program to handle. In such a field, reasoning can begin from explicit rules enacted by a legislature. In contrast, fields based on case law, or *common law*, develop their rules gradually through decisions in individual cases, and whatever rules there are are not officially tied to any fixed form of words.

The task of statutory interpretation, however, raises many special problems of its own (see Dias 1979, chap. 7). Further, once there

have been decisions interpreting the statutes, one must then be concerned with interpreting the decisions too. It would be unrealistic to try to work with statutes to the exclusion of case law. Beginning from statutes therefore seems likelier to add a layer of complication than to remove one.

There are also some special features of offer and acceptance that make it an attractive subject. From an AI point of view, it differs from assault and battery in being centrally concerned with interpreting what people have said to one another. On the legal side, several major writers on jurisprudence have also had a special interest in the law of contracts (Llewellyn 1938, 1960; Fuller 1969; Fuller and Eisenberg 1981; Gilmore 1974, 1977). In much writing about legal reasoning in general, then, it is often fair to take reasoning about contracts cases as a prototype.

The Choice of Law Examination Problems

As a vehicle for studying reasoning about contracts cases, law examination problems have several advantages. First, they are at a reasonable level of difficulty for an AI program to undertake. In one sense they are toy problems, since they do not present real cases. They are not, however, toy problems made up by the programmer. And although they may be more straightforward to analyze than many real cases—for example, the Supreme Court cases that McCarty considers (McCarty 1977; McCarty and Sridharan 1982)—it will be correspondingly easier to get a consensus as to the adequacy of whatever analysis the program produces.

In another sense examination problems are very real: the careers of would-be lawyers depend on being able to answer them satisfactorily. The skills they require, therefore, are those that all lawyers can be expected to share. In contrast, many other legal skills—such as interviewing clients, drafting documents, negotiating agreements, and trying cases—are developed mostly by apprenticeship and only as one's practice requires.

Law examination problems have some conveniently simplifying features. One is that all the relevant events have already occurred; there is no question of legal reasoning as planning in an uncertain world. The program need not find a plan of action to bring about a legal consequence, like keeping one's taxes low or ensuring that a prospective transaction will have the desired effect, but need only consider the consequences of actions already taken.