based on just input signals ("weak" conditions) in LEAP [254].

ACKNOWLEDGMENTS

The author wishes to express his sincere appreciation to Professors Tom Mitchell and Lou Steinberg of Rutgers University for their fruitful guidances when he was at Rutgers as a visiting researcher. He would like to thank Dr. Katsuya Hakozaki, Dr. Masahiro Yamamoto and Mr. Nobuhiko Koike of NEC Corporation for their encouragement.

In Machine Learning: A Guide to Current Research, T.M. Mitchell, J.G. Carbonell, and R.S. Michalski (eds.) Kluwer Academic, 1986.

(V-3)

OVERVIEW OF THE ODYSSEUS LEARNING APPRENTICE

David C. Wilkins, William J. Clancey, and Bruce G. Buchanan

Stanford University, Department of Computer Science
Stanford, CA 94305

ABSTRACT

Human specialists employ impressive learning methods during their apprenticeship training period to augment their fledgling expertise. We describe an apprentice learning system under development that allows an expert system to use some of these same methods. These methods aid an expert system in transferring expertise to and from its knowledge base (i.e., in knowledge acquisition and intelligent tutoring).

Our approach to apprenticeship learning is embodied in a computer program, Odysseus, that watches the observable actions of a specialist. Justifications are created for each action of the specialist via a process of differential modeling between the specialist and the expert system. A learning opportunity occurs when no action justification is judged sufficiently plausible. This paper describes the three phases that Odysseus uses to learn via differential modeling: setting the stage for differential modeling by expanding the initial rule base and deriving rule justifications, detecting knowledge base differences by observing actions of a specialist and ranking proposed action justifications, and effecting knowledge base repair by rationalizing discrepancies and postulating new rules.

INTRODUCTION

An apprenticeship learning period is an important phase on the path to master expert status for human specialists.[1] During this phase, an apprentice specialist *learns by watching master specialists* and *learns by doing problem solving* under the supervision of master specialists. Our research investigates how to give an expert system the benefits of an

[1] By *specialist*, we mean a problem solver whose abilities are at the novice or master level, and who is either a human or an expert system.

apprenticeship period.

Our method of apprenticeship learning is embodied in a computer program, Odysseus, that learns by watching specialists in the domain of medical diagnosis. The central task of Odysseus is to *rationalize* each observable action of a specialist during problem solving sessions. In medical diagnosis, these actions consist of all data requests made by a physician and the final diagnosis. Actions are rationalized by a process of *differential modeling* between the expert system and the specialist. Failure to find an adequate rationalization signals a possible deficiency in the expert system's domain or strategy knowledge. Using a taxonomy of deficiencies in conjunction with theoretical and experiential knowledge of the application domain, Odysseus automatically generates and tests conjectures to explain its inability to justify a specialist's action.

Odysseus is designed to work in conjunction with Heracles, an expert-system shell for solving heuristic classification problems, that was created by removing the medical knowledge from Neomycin [72]. Neomycin is a reorganization of the Mycin expert system that simulates the diagnostic process of medical experts, via a large body of abstract domain-independent strategy knowledge for hypothesis-directed reasoning. This strategy knowledge is used by Odysseus as a framework for detecting differences between the domain knowledge of a Heracles-based expert system and of a specialist. Odysseus has an abstract strategy language that allows comparison of the strategic behavior of the expert system and of a specialist [394].

ODYSSEUS' METHOD

Expanding Rule Base and Deriving Rule Justifications

There are two ways in which an existing expert system must be augmented before differential modeling of a human specialist can commence. First, the set of heuristic rules must be expanded via induction over past problem solving cases. The original set of rules is adequate for problem solving but, in our experience, is too impoverished to model the alternate problem solving behavior of other specialists in an apprentice context. Second, rules should be justified from first-principle knowledge or experience. Rule justifications allow a learning system to reason about the rules during the process of rationalizing discrepancies. The Leap learning apprentice for circuit design justifies rules in terms of circuit theory, a strong theory of the domain [254]. By contrast, only a weak theory generally underlies medical diagnosis, and Odysseus's justifications for rules

rely strongly on their empirical predictive power.

The induction subsystem of Odysseus is principally concerned with searching the space of rules of the form *lhs* → *hypothesis* (*cf*), where *cf* is a Mycin-type certainty factor. A constrained rule generator and a candidate rule evaluator find all *lhs* forms that meet given constraints of minimal rule generality (coverage), minimal rule specificity (discrimination), maximal rule collinearity (similarity), and maximal rule simplicity (number of conjunctions and disjunctions). The rule evaluator always gives preference to collinear forms of heuristic rules contained in the original rule base. The expanded rule set produced by the induction subsystem is necessarily incomplete; however, it bootstraps the differential modeling process that leads to its refinement. Later, we will discuss how the induction subsystem suggests missing rules to the repair subsystem during the process of rationalizing discrepancies.

Observing Actions and Detecting Knowledge Base Differences

Odysseus must decide whether an action of the specialist suggests a significant domain or strategic knowledge difference between the specialist and the expert system. For each observed action of the specialist, Odysseus generates an action justification set: $J(A) = (j_1, j_2, \ldots, j_n)$. An action justification structure, j_k, relates an action A to an abstract strategic goal G via a skeletal rule path, that is, $A \rightarrow R_1 \rightarrow R_2 \rightarrow \cdots \rightarrow G$. A typical goal might be the confirmation of a particular hypothesis. All skeletal rule paths beginning with A and leading to a goal are in the set J(A); thus the set delimits the possible interpretations that can be attributed to the specialist's action. Using the original Neomycin rule base, the average size of J(A) is 20 and the maximum size is approximately 400.

Action justification sets are posted on a blackboard, and a variety of knowledge sources (KSs) attach confirming and disconfirming evidence to individual action justifications. The more important KSs are as follows: The *Heracles simulator* KS processes the information obtained during the problem solving session and relates the current status of findings, hypotheses, and rules to individual action justifications. For example, if this KS believes that particular hypotheses have already been concluded, then it attaches negative evidence to all action justifications whose goal is to confirm one of these hypotheses. The *multiple interpretations* KS consists of heuristic rules that medical domain experts use to arbitrate between multiple interpretations. For instance, early in the consultation session with different action justifications confirming different hypotheses, the more general hypotheses are preferred. The *user model* KS records user

characteristics such as individual diagnostic style preferences, and employs these to help arbitrate between competing action justifications. For example, some problem solvers have a depth-first problem-solving style, meaning that they pursue a particular hypothesis as soon as there is weak evidence confirming it. Such a heuristic adds support to those action justifications consistent with this style. The *strategic distance* KS determines the similarity between the expert system's preferred strategic action and the strategic action associated with each action justification. The *patterns of interpretation* KS rates competing justifications according to the overall coherence they lend to the specialist's strategic plan.

After evidence has been gathered for and against members of J(A), the action ranking subsystem must decide if any of the top ranked justifications are equal to the specialist's justification j_s. This is the most difficult task the apprentice learner faces, since there are often weakly plausible interpretations under which any action of the specialist is reasonable; yet to learn, the program must recognize when none of its action justifications obtain. Some apprentice systems do not need to confront this problem. In Leap, the specialist's design actions implicitly contain the domain knowledge to be acquired; while in Odysseus the medical specialist's actions only indirectly reflect the domain knowledge to be learned.

When the specialist being observed is Neomycin, Odysseus always selects the correct action justification from J(A). However, when observing other novice and master medical specialists, j_s was not in the set of justifications generated from the original Neomycin knowledge base 75% of the time. This incompleteness was due to sparse domain knowledge, and motivated the initial induction phase to expand the rule base. Observing actual specialists also showed that more heuristics needed to be added to the KS's.

Rationalizing Discrepancies and Postulating New Rules

A learning opportunity exists when Odysseus concludes that no member of J(A) adequately explains the specialist's action. At this point a repair subsystem, under implementation, engages the specialist in a dialogue to determine j_s. If j_s is not in J(A), then there is a domain knowledge difference between the specialist and the expert system. If j_s is indeed in J(A), then there is a domain or strategy knowledge difference, or a problem with the ranking heuristics. The evidence explicitly linked to the action justification structure by the KSs should allow Odysseus to isolate the cause of the difference.

Odysseus will also learn without engaging the specialist in a dialogue. Given an unexplained action, the action justification subsystem will provide the repair subsystem with the most likely goal(s) to which the specialist's action relates. The repair subsystem will perform a bidirectional search for a skeletal path between the action and the goal, calling on the the induction subsystem to check for rules that could connect the two search frontiers.

SUMMARY AND FUTURE WORK

This paper has provided an overview of the three phases used by Odysseus to automate the transfer of expertise for expert systems, and given the results of implementing the first two phases. We are currently implementing the method for rationalizing discrepancies and the expanding the heuristics associated with the KSs.

Odysseus is to be tested as a knowledge acquisition subsystem for the Heracles expert system, and also tested as a student modeling subsystem for a Heracles-based intelligent tutoring system (Guidon2). A third, more systematic validation exercise involves apprenticeship learning between two expert systems. In this exercise, a novice expert system will always have one less piece of knowledge than the master expert system. Each type of knowledge relation, such as trigger properties on rules, will be systematically removed from the knowledge base of the novice. For each removal, we will test whether the novice can learn the missing piece of knowledge in an apprentice setting. These multiple perspectives should aid us in becoming experts in the transfer of expertise.

ACKNOWLEDGMENTS

The apprentice learning framework described here owes much to discussions with Avron Barr, Jim Bennett, Tom Dietterich, Paul Scott, and Derek Sleeman. Marianne Winslett Wilkins and Paul Rosenbloom gave insightful comments on earlier drafts. For critiquing Odysseus during its development, we are grateful to doctors Larry Fagan, Curt Kapsner, Randy Miller, Mark Musen, Roy Rada and Ted Shortliffe.

This work was supported in part by NSF grant MCS-83-12148 and ONR/ARI contract N00014-79C-0302. Computational resources were provided by SUMEX-AIM (NIH grant RR 0078) and Xerox PARC.