Title
Learning Only Good Things From Others

Journal
Proceedings of the Annual Meeting of the Cognitive Science Society, 20(0)

Author
Upal, Muhammad Afzal

Publication Date
1998

Peer reviewed

Learning Only Good Things From Others

Muhammad Afzal Upal
Department of Computing Science
University of Alberta, Canada
email: upal@cs.ualberta.ca

A number of protocol studies comparing the performance of experts and novices have concluded that differences exist not just between the amount of knowledge possessed by novices and experts but also in the way experts and novices organize, retrieve and use knowledge to solve problems. Active apprenticeship has been suggested as an effective technique to teach novices to becme experts. Recently machine learning as well as knowledge acquisition researchers have turned to apprenticeship learning as an effective way of acquiring problem solving knowledge.

A large number of AI apprenticeship systems have been developed to apply to a variety of problem solving tasks. However, these systems assume that only one user interacts with the system and that that user is infallible. Human apprentices are able to learn from multiple mentors (who may at times be wrong) by learning from only those experiences in which they can explain to themselves that the teacher's way of solving the problem is better. To address these limitations, we employ PIPP, a multi-strategy learning and planning system that learns to produce improve the efficiency as well as quality of the solutions produced by a partial order planner. PIPP remembers the good planning episodes along with the justification as to why they are good. It also learns local control rules that suggest a preferred alternatives at a particular decision point.

Previous case-based planning systems such as PRODIGY/ANALOGY and DerSNLP that only learn to improve planning efficiency index their solutions by *relevant initial conditions*. These are the initial conditions that are needed as preconditions for actions in the final plan. Presence of these conditions in the new problem specification guarantees that the plan created under the guidance of the previous case will satisfy the currently active goals but not that it will lead to a high quality plan. To ensure that the retrieved case's guidance will lead to a high quality solution we also need to remember the conditions under which the previous plan was *better* (we call these conditions the *betterness conditions* of a plan). Betterness conditions are the range of the values of variables (provided in the problem specifications) under which a plan is better. Together with the relevent initial conditions they form the *distinguishing features of the plan* that PIPP uses to store and retreive plans.

Given a problem description, PIPP searches for a similar case in its case memory and uses the retrieved case to guide plan construction. If no such case is found, it uses the control rules (if any are applicable) that it has learned to guide plan construction. A generative planner is only used when no guiding knowledge is available. The input to PIPP is a problem statement and user's plan in the form of an ordered set of actions. Following is a brief outline of PIPP's learning algorithm.

1. Given a problem description generate the system's solution to the problem. Generate a trace for the user's solution identifying the differences in selections from the system's solution at various choice points.
2. Compare the user's and the system's solution for quality and identify the better solution and the worse solution. Determine the distinguishing features of the better and the worse plan. Index (re-index) each case with the distinguishing features.
3. Compare the planning traces of both plans and form preference rules that indicate 'prefer the choices made during the "better" planning episode if the partial plan under refinement has the distinguishing features that the "better" partial plan possessed at this point in the refinement.'

We have tested the performance improvements obtained by storing and retrieving the cases with betterness-conditions. We randomly generated 100 problems from the modified logistics transportation, trained PIPP on first 10 of them and compared its performance on the remaining 90 problems with DerSNLP. Problems contained two trucks and one plane which were distributed among two cities. We varied the number of packages from one to two and ran five trials for each case. The results are quite encouraging in that they indicate that this relatively small amount of training provided substantial improvements in planning performance (10-15 %). PIPP does not produce worse solutions on any of the test examples. Preliminary results are encouraging but more extensive testing will have to be done to see that whether the improvements in plan quality come at the cost of planner efficiency.