

Case-based Explanations

and the Integrated Learning of Demonstrations

Michael T. Cox, Mark H. Burstein

The human ability to learn procedures by observing a single demonstration is an amazing skill and a computational challenge. Statistical and traditional inductive methods fail for lack of sufficient examples. Explanation-based learning techniques, by contrast, apply background knowledge to learn from single examples but in recent years have been concentrated on the acquisition of conceptual knowledge. This project report illustrates how an implemented system called POIROT integrates multiple learning algorithms to produce a hierarchical generalized task network that represents an observed demonstration of a sequence of semantically annotated web service transactions. It uses its representation of what it learned to plan out solutions to similar problems, currently in the domain of medical evacuation. The focus of this report is on the case-based explanation mechanism embodied in POIROT's XPLAIN learning component. XPLAIN retrieves explanation patterns from a case library and applies them to portions of the demonstration trace to explain how the observed web service calls are causally related when the evidence for that is not present in the data-flow between calls that POIROT's other learners utilize. These additional explanations help complete the formulation of a complete procedural model.

1 Introduction

POIROT (Plan Order Induction by Reasoning from One Trial) is an integration architecture for controlling a complex, multi-strategy learning process [1]. The goal is to combine the results from multiple learning and reasoning strategies to learn hierarchical procedure models given one or just a few input demonstrations. In general this kind of learning requires the application of inductive, abductive and explanation-guided generalization techniques and ultimately the ability to do self-guided exploration of the space of activities to confirm or enhance confidence in one's ability to perform the task. This research report examines the role of explanation within this larger learning framework.

POIROT currently learns in the domain of medical evacuation planning, where a sequence of web services are called to find flights for patients and reserve seats on those flights so that they get to appropriate treatment facilities. Input consists of the set of semantic web service calls underlying an expert demonstration required to schedule a number of individual patients for transport and to address their need for ambulances and medical equipment en route. Some learning components are responsible for learning step sequential ordering regularities in the trace and for identifying causal dependencies that directly tie steps together, principally based on the observed flow of data values produced by one step to the parameters used in calling another. Unfortunately the conclusions reached by these components, even when carefully combined, form an incomplete model of the process because they cannot identify the role of several kinds of steps: (1) steps that serve as memory aids, storing chosen values for future use, and (2) steps that request permission or make reservations. Neither of these action classes are easily generalized by these learners, because they do not produce new effects upon which other observed actions depend. The former because the data that is stored was already produced. The latter because the actions being requested (by reser-

vations) are not observed during the planning process that constitutes the demonstration. In both instances, however, explanation facilitates learning.

Many of POIROT's learning components can be viewed as processing a stream of observations to identify causal patterns that explain how step relate to form a coherent whole. While inductive hypotheses formers can identify regularities in the stream when several repetitions exist, aspects of the workflow that are not repeated or have no apparent utility because they do not have directly observable causal support are difficult to generalize. But by applying additional background knowledge to identify relevant features and by extending the observations to form an *interpretation of the demonstration* that includes unobserved activities, we can provide learners with appropriate biases. This constitutes a key role of explanation. Explanatory schemas and analogically related cases link otherwise unrelated steps and tie observed events to the goals of the demonstrator. POIROT's XPLAIN (eXplanation Patterns for Learning And INtrospection) component performs this type of explanation. It is based on Meta-AQUA [2], a goal-driven explanation and learning system that has the top-level goal to understand each observation in the input stream.

2 Case-Based Explanation

Explanations are useful for functions including event prediction, blame assignment, and diagnosis for repair [4], but their most important purpose is to establish causal structure and relevance during learning [4, 6, 7]. Explanations provide the basis for elaborating an agent's model of the environment, especially when agents' perceptions diverge from their expectations. In our project, several kinds of explanations are utilized. Learning modules use conclusions about data-flow relationships between steps to determine an important class of causal dependencies directly from the observables. Several modules also consider explicitly the semantic