A Relational Hierarchical Model for Decision-Theoretic Assistance

Sriraam Natarajan, Prasad Tadepalli, and Alan Fern
School of EECS, Oregon State University, Corvallis, Oregon, USA

Abstract. Building intelligent assistants has been a long-cherished goal of AI and many were built and fine-tuned to specific application domains. In recent work, a domain-independent decision-theoretic model of assistance was proposed, where the task is to infer the user’s goal and take actions that minimize the expected cost of the user’s policy. In this paper, we extend this work to domains where the user’s policies have rich relational and hierarchical structure. Our results indicate that relational hierarchies allow succinct encoding of prior knowledge for the assistant, which in turn enables the assistant to start helping the user after a relatively small amount of experience.

1 Introduction

There has been a growing interest in developing intelligent assistant systems that help users in a variety of tasks ranging from washing hands to travel planning [2,6,3]. The emphasis in these systems has been to provide a well-engineered domain-specific solution to the problem of reducing the users’ cognitive load in their daily tasks. A decision-theoretic model was proposed recently to formalize the general problem of assistantship as a partially observable Markov decision process (POMDP). In this framework, the assistant and the user interact in the environment to change its state. The goal of the assistant is to take actions that minimize the expected cost of completing the user’s task [9]. In most situations, however, the user’s task or goal is not directly observable to the assistant, which makes the problem of quickly inferring the user’s goals from observed actions critically important. One approach to goal inference [9] is to learn a probabilistic model of the user’s policy for achieving various goals and then to compute a posterior distribution over goals given the current observation history. However, for this approach to be useful in practice, it is important that the policy be learned as early in the lifetime of the assistant as possible. We call this the problem of “early assistance”, which is the main motivation behind this work.

One solution to the early assistance problem, advocated in [9], is to assume that (a) the user’s policy is optimal with respect to their goals and actions, the so called “rationality assumption,” and that (b) the optimal policy can be computed quickly by knowing the goals, the “tractability assumption.” Under

In this work, we use the words task and goal interchangeably.
these assumptions, the user’s policy for each goal can be approximated by an
optimal policy, which may be quickly computed. Unfortunately in many real
world domains, neither of these assumptions is realistic. Real world domains
are too complex to allow tractable optimal solutions. The limited computational
power of the user renders the policies to be locally optimal at best.

In this paper, we propose a different solution to the early assistance prob-
lem, namely constraining the user’s policies using prior domain knowledge in
the form of hierarchical and relational constraints. Consider an example of a
desktop assistant similar to CALO [4] that helps an academic researcher. The
researcher could have some high level tasks like writing a proposal, which may
be divided into several subtasks such as preparing the cover page, writing the
project description, preparing the budget, completing the biography, etc. with
some ordering relationships between them. We expect that an assistant that
knows about this high level structure would better help the user. For example,
if the budget cannot be prepared before the cover page is done, the assistant
would not consider that possibility and can determine the user’s task faster. In
addition to the hierarchical structure, the tasks, subtasks, and states have a class
and relational structure. For example, the urgency of a proposal depends on the
closeness of the deadline. The deadline of the proposal is typically mentioned on
the web page of the agency to which the proposal is addressed. The collaboration
potential of an individual on a proposal depends on their expertise in the areas
related to the topic of the proposal. Knowing these relationships and how they
influence each other could make the assistant more effective.

This work extends the assistantship model to hierarchical and relational set-
tings, building on the work in hierarchical reinforcement learning [10] and statis-
tical relational learning (SRL). We extend the assistantship framework of [9] by
including parameterized task hierarchies and conditional relational influences as
prior knowledge of the assistant. We compile this knowledge into an underlying
Dynamic Bayesian network and use Bayesian network inference algorithms to
infer the distribution of user’s goals given a sequence of their atomic actions.
We estimate the parameters for the user’s policy and influence relationships by
observing the users’ actions. Once the user’s goal distribution is inferred, we de-
terminate an approximately optimal action by estimating the Q-values of different
actions using rollouts and picking the action that has the least expected cost.

We evaluate our relational hierarchical assistantship model in two different toy
domains and compare it to a propositional flat model, propositional hierarchical
model, and a relational flat model. Through simulations, we show that when
the prior knowledge of the assistant matches the true behavior of the user, the
relational hierarchical model provides superior assistance in terms of performing
useful actions. The relational flat model and the propositional hierarchical model
provide better assistance than the propositional flat model, but fall short of the
performance of the relational hierarchical approach.

The rest of the paper is organized as follows: Section 2 summarizes the ba-
sic decision-theoretic assistance framework, which is followed by the relational