Predicting Resource Use with Case-based Plan Recognition

Jung-Jin Lee and **Robert McCartney**

Department of Computer Science and Engineering
University of Connecticut
Storrs, CT 06269-3155
jjl@cse.uconn.edu and robert@cse.uconn.edu

When managing a computing system, it is better to predict problems before they occur, rather than just observing them when they occur. Our research develops self-interested agents designed for the *prediction* of resource usages, that is, assessing the likelihood of upcoming demands by users on the limited resources and detecting potential problems from the observations of human-computer interactions (Etzioni & Weld 1994). Furthermore, these agents will have behavior that changes over time based on their own experiences, which will improve their predictive ability and allow them to adapt to changing usage patterns.

There are two major sources of uncertainty : there is usually more than one plan consistent with the observations, and uncertainty about whether future actions in any given plan will be executed. The plan recognition problem in this work is to observe a user's ongoing interactions with the computing environment, analyze those interactions by recognizing behaviors relevant to the agent's interest, and use those relevant behaviors to make predictions about the user's future behaviors. This is done incrementally: as the user performs actions, the system adjusts its set of possible plans (those consistent with the observations), and bases its predictions on the likelihood of each candidate plan and the likelihood of the behavior of interest within each plan. The predictions for individual users are then aggregated over all current users to predict system behavior.

Some difficulties inherent in recognizing a user's plans in this domain are the nonstrict temporal orderings of actions, the interleaving of multiple tasks, the large space of possible plans, and conditional plans where the condition is neither explicit nor directly observable. On the plus side, the language of discourse (UNIX commands and system responses) is relatively small, grammatically simple, and easy to monitor, given we are only concerned with textual interactions.

Our approach to the plan recognition-prediction problems is based on a mixture of case-based reasoning and statistical prediction. Cases (past plan executions) are used to suggest plans consistent with observations, while statistical information (Markov model) is used to distinguish and evaluate alternatives and come up with numerical predictions. We propose to use *case-based plan recognition* with *situated* cases to deal with our incomplete knowledge about the environment. These situated cases are the source of the model for the prediction system; the predictions result from using the cases to explain observed actions and behaviors.

The stored cases are organized as graphs to represent both ordering and branching information while graphs are built from observation incrementally. This organization supports the aggregation of cases, as well, so we can make explicit the relationships among the parts of a given case, and the relationships among related cases and their parts, as in (Redmond 1990) and (ZitoWolf & Alterman 1993). Templates of how users access resources are used as abstractions of actions, with structural procedures for interpreting a case being observed and for organizing cases to bridge both retrieval and indexing. The relevance of information is determined with both linkages of actions from the case graphs and analysis of coherences with currently observed actions.

The CBPR technique which this work develops uses a *hidden Markov model* to represent the likelihood of behaviors in the statistical models, and provides the probability of resource usages from ongoing observations. States represent actions, and transition probabilities are derived from observed executed cases, which suggests a fairly straightforward approach to updating predictive behavior. This model provides a principled way to both assess the likelihood of competing consistent plans, and generating probabilistic predictions given a set of candidate plans that can be updated incrementally.

References

Etzioni, O., and Weld, D. 1994. A softbot-based interface to the internet. *Communications of the ACM 37(7)* 72–76.

Redmond, M. 1990. Distributed cases for case-based reasoning; facilitating use of multiple cases. In Proceedings AAAI-90. *AAAI*, 304–309.

ZitoWolf, R., and Alterman, R. 1993. A framework and an analysis of current proposals for the case-based organization and representation of procedural knowledge. In Proceedings AAAI-93. *AAAI*, 73–78.