A Context-Aware Personal Desktop Assistant

(Demo Paper)

Hung H. Bui, Federico Cesari, Daniel Elenius, David N. Morley
SRI International
Menlo Park, CA, U.S.A.
first.lastname@sri.com

Sriraam Natarajan
Oregon State University
Corvallis, OR, U.S.A.
natarasr@eecs.oregonstate.edu

Shahin Saadati, Eric Yeh, Neil Yorke-Smith
SRI International
Menlo Park, CA, U.S.A.
first.lastname@sri.com

ABSTRACT

We demonstrate an intelligent personal assistant agent that has been developed to aid a busy knowledge worker in managing time commitments and performing tasks. The *PExA* agent draws on a diverse set of AI technologies that are linked within the SPARK BDI agent framework. We focus on our agent's ability to provide assistance within the context of current user activities, based on its recognition of user workflows and their progress, and on its context-sensitive proactive suggestions. We have instrumented a common suite of desktop applications so that, endowed with a sophisticated workflow tracker, PExA has the ability to pervasively monitor the user's desktop activities. PExA follows and responds to the user's progress on shared tasks, and is highly user-centric in its support for user needs and its adaptivity to user working style and preferences.

Categories and Subject Descriptors

I.2.11 [**ARTIFICIAL INTELLIGENCE**]: Distributed Artificial Intelligence—*Intelligent agents*; I.2.1 [**ARTIFICIAL INTELLIGENCE**]: Applications and Expert Systems—*Office automation*

General Terms

Algorithms, Design, Human Factors

Keywords

activity recognition, proactive assistance, integrated cognition, CALO

1. THE PEXA AGENT

The vision of an agent that acts as your personal butler, attentive to your requests, aware of your goals and preferences, and anticipating your needs, requires the agent to act appropriately according to context [4]. Our work on the *PExA* (Project Execution Assistant) agent is part of the *CALO* project, a large-scale effort to build an adaptive, interactive cognitive assistant situated in the office environment [2]. The overall CALO system is designed to support its user in various ways including project and task management, information organization, and meeting preparation and summarization.

A critical aspect of context in this setting is the user's current activity within the larger scope of his or her current and future tasks. The agent must be able to understand what the user is working on in the present and what goals the current activities are directed towards. This demonstration exhibits aspects of PExA's assistance, especially those capabilities which are enabled by the agent's ability to understand the user current activities and goals.

Fig. 1 shows part of the PExA architecture. The *Workflow Tracker* recognizes current user activity, based on Logical Hidden Markov Models (Sect. 2). The *Task Manager*, based on the SPARK agent platform [5], deliberates and performs assistive actions (Sect. 3). Fig. 2 shows an example of suggestions made proactively by PExA.

2. WORKFLOW RECOGNITION

In order to assist its user, a PExA agent requires an understanding of the user's goals on the desktop, and knowledge of means by which the user and agent together can achieve these goals. PExA's task library consists of a dual declarative/procedural representation. Declaratively, the task models describe the subtasks that the user, the agent, or other agents achieve to fulfill a goal, together with constraints between them (such as ordering of subtasks). Procedurally, the task models provide recipes that PExA can instantiate into a plan to achieve a given subtask. Some tasks in the library might be executed only by the user, only by the agent, or both.

We call the declarative representation of tasks *workflows*. A workflow models a pattern of behaviour of the user (perhaps aided by other agents) in achievement of a user goal. For example, in the workflow *journal paper review*, the user first downloads the paper and the review form attached to the review request email. Next, the paper is printed, and the review form is filled out. Finally, the review form is sent back as a reply to the request email.

Knowledge of progress through the workflow (e.g., current step) enables PExA to volunteer information and suggestions (e.g., related documents, emails, web links) specifically chosen for the right context, to provide summarization of progress (e.g., "waiting on Alice to complete this step"), and to itself act (e.g., offer to perform the next step, prepare for future steps).

Keeping track of progress is challenging for steps being executed by the user. It would be burdensome for PExA to require the user to explicitly indicate commencement and completion of every step. We call the problem of automatically identifying the workflow and the user's current step *workflow recognition and tracking*. We instrumented the desktop (Windows Explorer) and common applications such as email clients (Thunderbird), web browsers (Firefox and Internet Explorer), and office applications (Word, PowerPoint, Excel) so that user-performed actions are captured and logged. A *Workflow Tracker* (Fig. 1) identifies whether the stream of captured interaction events matches with any of the workflows in the task library, and if so, tracks its current progress.

Cite as: A Context-Aware Personal Desktop Assistant (Demo Paper), H. H. Bui, F. Cesari, D. Elenius, D. N. Morley, S. Natarajan, S. Saadati, E. Yeh, N. Yorke-Smith, *Proc. of 7th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2008)*, Padgham, Parkes, Müller and Parsons(eds.),May,12-16.,2008,Estoril,Portugal,pp.1679-1680.

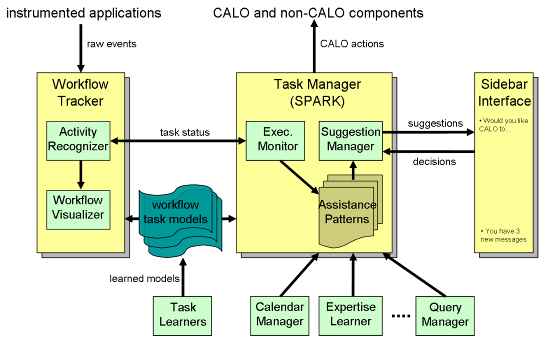

Figure 1: Partial architecture of the PExA agent.

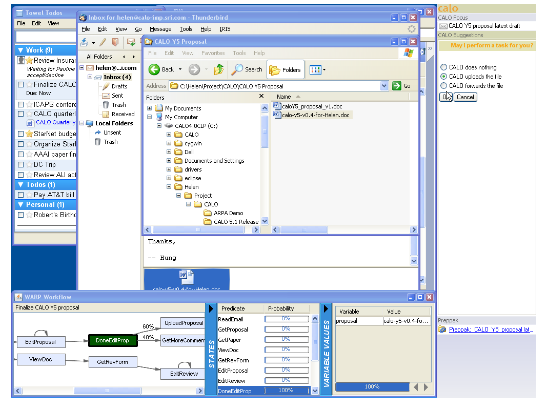

Figure 2: Context-aware assistance provided by PExA.

Variants of the *Hidden Markov Model* (HMM) have been used for this kind of activity tracking problem. However, in the desktop domain, steps in a workflow are often associated with a particular desktop object such as an email, file, or webpage, best described as a parameter for the step (e.g., *OpenDocument('review.doc')*). In order to accommodate workflow parameters, PExA uses a *Logical HMM* [3] as its representation of the workflow model.

The Logical HMM extends the HMM state to take the form of a ground atom in a first-order language. State transitions can be written in the form of logical rules, such as $OpenDocument(X) \rightarrow EditDocument(X) : 0.8$. Here, variable X ranges over the set of documents in the system, and 0.8 represents the transition probability. Similarly, the observation model is $OpenDocument(X) \rightarrow WordOpenEvent(X) : 1.0$. In order to model irrelevant activities between workflow steps (e.g., the user reads some other emails), a special 'Background' state is included in the model; it can generate any observable event uniformly. Workflow recognition can then be viewed formally as a filtering problem on the Logical HMM representing the workflow. We adopt a particle filter approach to avoid the prohibitive cost of exact inference. Given a stream of user interaction events, the algorithm returns a distribution over the possible steps in the workflow (including the 'Background' state). This allows PExA both to identify the most likely step and to identify the most likely parameter values for this step.

3. ASSISTIVE CAPABILITIES

The core ability of a PExA agent to act is its *Task Manager*, a module built in the BDI-based SPARK framework [5]. The Task Manager enables PExA to decide *what* to do, given the rich multitude of state and environmental information provided in the CALO architecture — including estimates of workflow states and parameters from the Workflow Tracker — and *when and how* to do it.

A limited subset of the Task Manager is shown in Fig. 1, focusing on the *Execution Monitor*, which collates information from the Workflow Tracker with internal PExA agent activity and reports from other agents, and the *Suggestion Manager*, which deliberates over whether and how to proactively act. This capability complements situations where PExA is obliged to act, such as when a subtask is explicitly delegated by the user.

The basis of the Suggestion Manager's deliberation are *Assistance Patterns*: specialized meta-level task models that describe proactive agent actions. They include, for example, executing the next step of a workflow (if it is enabled and agent executable), reminding the user of a forthcoming appointment, and suggesting that a task be delegated to another agent. The Suggestion Manager accounts for the user's interaction preferences, her current activity (to avoid acting or interrupting out of context), the potential consequences of its actions (cost, reversibility), the certainty of its information (e.g., confidence of workflow state), and adjustable autonomy permissions. PExA therefore acts, asks, suggests, or does nothing, in order to assist and not irritate the user [1]. Fig. 2 shows a suggestion unobtrusively manifest in the *CALO Sidebar* (right).

Participants in the demonstration can appraise the full functionality, including meeting scheduling, task delegation, and (simulated) item procurement; perform joint tasks with PExA in the instrumented desktop environment and observe (and correct) workflow recognition; and specify preferences and guidance, oversee the agent's operation, and respond to or disregard its suggestions.

4. FUTURE PLANS

We are currently working on improving the scope and depth of PExA's capabilities in a number of ways. Specific to the focus of this demonstration, we are exploring an approach based on learning by demonstration that allows the user to quickly and naturally specify a workflow. Ongoing work seeks to improve the robustness of the Workflow Tracker, with focus on tracking multiple interleaving workflows. At the same time, we are extending the Suggestion Manager's reasoning over the timing and modality of its proactive interaction with the user.

Acknowledgements. We thank all the members of the CALO project, especially the PExA team. This work was supported by the Defense Advanced Research Projects Agency (DARPA) under Contract NBCHD030010. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the view of DARPA or the Department of Interior-National Business Center.

5. REFERENCES

[1] E. Horvitz. Principles of mixed-initiative user interfaces. In *Proc. of CHI-99*, pages 159–166, Pittsburgh, PA, 1999.

[2] K. Myers, et. al. An intelligent personal assistant for task and time management. *AI Magazine*, 28(2):47–61, 2007.

[3] K. Kersting, L. De Raedt, and T. Raiko. Logical Hidden Markov Models. *Journal of Artificial Intelligence Research*, 25:425–456, 2006.

[4] H. Lieberman and T. Selker. Out of context. *IBM Systems Journal*, 39(3–4):617–632, 2000.

[5] D. Morley and K. Myers. The SPARK agent framework. In *Proc. of AAMAS'04*, pages 714–721, New York, NY, 2004.