In: Proceedings of the 2018 Workshop on Knowledge Engineering for Planning and Scheduling (KEPS), Delft, Netherlands, June 2018

Evaluating a Knowledge-Based Scheduling Assistant

Neil Yorke-Smith
Delft University of Technology, Netherlands, and
American University of Beirut, Lebanon
n.yorke-smith@tudelft.nl

Abstract

We summarize a recent article that studies the evaluation of a knowledge-based scheduling system. The article considers a user-adaptive personal assistant agent designed to assist a busy knowledge worker in time management. We examine the managerial and technical challenges of designing adequate evaluation and the tension of collecting adequate data without a fully functional, deployed system. The PTIME agent was part of the CALO project, a seminal multi-institution effort to develop a personalized cognitive assistant. The project included a significant attempt to rigorously quantify learning capability in the context of automated scheduling assistance. Retrospection on negative and positive experiences over the six years of the project underscores best practice in evaluating user-adaptive systems. Through the lessons illustrated from the case study, the article highlights how development and infusion of innovative technology must be supported by adequate evaluation of its efficacy.

Evaluation of the Personalized Time Management (PTIME) Agent

The case study article by Berry et al (2017) reports and critiques the *evaluation* of a knowledge-based scheduling system that learns preferences over an extended period. The domain of application is personal time management, in particular, providing assistance with arranging meetings and managing an individual's calendar. The *Personalized Time Management* (PTIME) calendaring assistant agent increased in usefulness as its knowledge about the user increases. The enabling technologies involved were preference modelling and machine learning to capture user preferences, natural language understanding to facilitate elicitation of constraints, and constraint-based reasoning to generate candidate schedules (Berry et al. 2011). Human-computer interaction (HCI) and interface design played central roles.

The PTIME system was part of a larger, seminal project, *Cognitive Assistant that Learns and Organizes* (CALO), aimed at exploring learning in a personalized cognitive assistant. Thus, the primary assessment of PTIME was in terms of its adaptive capabilities, although such a knowledge-based system must necessarily have a certain level of functionality to assist with tasks in time management, in order to provide a context for learning.

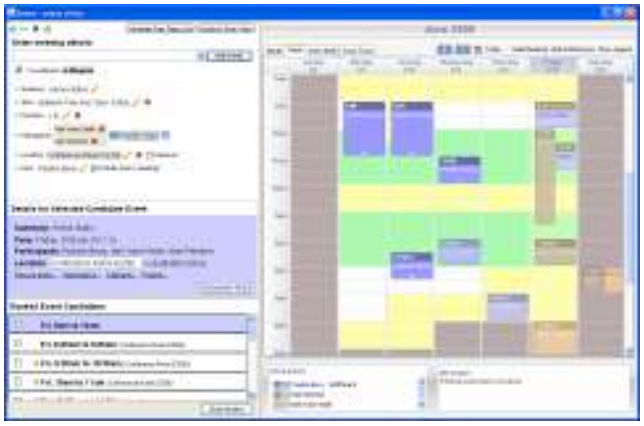

Figure 1: Screenshot of the PTIME system.

At the commencement of the project, however, the degree of robustness and usability required to support evaluation was not immediately obvious. Evaluation was focused almost exclusively on the technology; experiments were designed to measure performance improvements due to learning within a controlled test environment intended to simulate a period of real-life use—rather than in a genuinely 'in-the-wild' environment. Technologists such as the majority of the authors are trained primarily to conduct such 'in-the-lab' evaluations, but—as argued in the article—many situations require placing the technology into actual use with real users in a business or personal environment, in order to provide a meaningful assessment. In retrospect, the authors suggest that the evaluation methodology of CALO gave too little attention to the usefulness and usability of the technology.

Scheduling in PTIME

We briefly review the role of automated scheduling in the PTIME system. PTIME consists of four main components: user interface, calendar proxy, constraint reasoner, and preference learner. In its main mode of operation, PTIME elicits an event request: for the user, this corresponds to stating details on the desired event to be arranged; these details correspond to constraints.

PTIME computes preferred candidate schedules (possi-

bly relaxations) in response to the request and presents a ranked subset of the candidate schedules to the user. Note that, because PTIME will consider moving existing events if necessary, the options presented to the user are schedules rather than single events. The number of such candidate schedules presented depends on the number of feasible schedules. PTIME will typically display 10 candidate schedules, including a mix of more optimal and more diverse options.

PTIME accepts the user's selection from among the presented candidate schedules. PTIME updates the preference model accordingly, based on the implicit feedback of the selected versus non-selected options. The updated model is used in the subsequent interactions.

These steps repeat as necessary, with the system presenting new or refined options after each new detail is entered or modified by the user. Through a collaborative negotiation process, event invitees comment, respond, and counter-propose to reach agreement over the event.

At the heart of the scheduling is the constraint reasoner. This component generates scheduling options in response to new or revised details and constraints from the user, using the current preference model to generate preferred options. The reasoner translates requests such as *"next tues afternoon with nigel and kim"* into a set of soft constraints, and solves a soft constraint problem with preferences (Moffitt, Peintner, and Yorke-Smith 2006). Soft constraints allow all aspects of the user's request—including times, location, and participants—to be relaxed in the case where the request cannot be satisfied, i.e., when the scheduling problem is over-constrained. Details of the constraint solving, and the other aspects of the system, are given in Berry et al. (2006; 2009; 2011).

Lessons Learned

The six lessons that emerged from the evaluation journey with PTIME are not unfamiliar from other experiences of evaluating (non-adaptive) systems (Cohen and Howe 1989; Nielsen and Levy 1994; Chen and Pu 2014):

1. The contexts of the use of technology, and the competing interests of the stakeholders, must be a primary focus in designing an evaluation strategy.
2. Evaluating one component based on an evaluation of a whole system can be misleading, and vice versa.
3. User-adaptive systems require distinct evaluation strategies.
4. In-the-wild evaluation is necessary when factors affecting user behaviour cannot be replicated in a controlled environment.
5. In-the-wild evaluation implies significant additional development costs.
6. Ease of adoption of the system by users will determine the success or failure of a deployed evaluation strategy.

Our hope is that, since the conclusion of the CALO project, these lessons are increasingly understood in Artificial Intelligence and its constituent communities, including the automated planning/scheduling community. Indeed, Foster and Petrick (2017) contrast differences between the latter community and the dialogue systems community. They discuss the overhead of integration, deployment in real-world environments, and the need to evaluate certain types of systems in-the-wild—as all encountered in the case of PTIME.

Summarizing the article, the main lesson from this case study of evaluation of a knowledge-based scheduling system is obvious but under-valued: researchers and project managers benefit from familiarity with and adoption of best practice in evaluation methodologies from the start of a technology project.

Acknowledgements Thanks to the KEPS workshop reviewers. This material is based in part upon work supported by the US Defense Advanced Research Projects Agency (DARPA) Contract No. FA8750-07-D-0185/0004. Views are the author(s) and do not necessarily reflect the views of DARPA. A shorter version of this article abstract was presented at the 29th Benelux Conference on Artificial Intelligence (BNAIC'17).

References

Berry, P. M.; Conley, K.; Gervasio, M.; Peintner, B.; Uribe, T.; and Yorke-Smith, N. 2006. Deploying a personalized time management agent. In *Proc. of AAMAS'06*, 1564–1571.

Berry, P. M.; Donneau-Golencer, T.; Duong, K.; Gervasio, M. T.; Peintner, B.; and Yorke-Smith, N. 2009. Mixed-initiative negotiation: Facilitating useful interaction between agent/owner pairs. In *Proc. of AAMAS'09 Workshop on Mixed-Initiative Multiagent Systems*, 8–18.

Berry, P. M.; Gervasio, M.; Peintner, B.; and Yorke-Smith, N. 2011. PTIME: Personalized assistance for calendaring. *ACM Transactions on Intelligent Systems and Technologies* 2(4):40:1–40:22.

Berry, P. M.; Donneau-Golencer, T.; Duong, K.; Gervasio, M.; Peintner, B.; and Yorke-Smith, N. 2017. Evaluating intelligent knowledge systems: Experiences with a user-adaptive assistant agent. *Knowledge and Information Systems* 52:379–409.

Chen, L., and Pu, P. 2014. Experiments on user experiences with recommender interfaces. *Behaviour & IT* 33(4):372–394.

Cohen, P., and Howe, A. E. 1989. Toward AI research methodology: Three case studies in evaluation. *IEEE Transactions on Systems, Man, and Cybernetics* 19(3):634–646.

Foster, M. E., and Petrick, R. P. A. 2017. Separating representation, reasoning, and implementation for interaction management: Lessons from automated planning. In Jokinen, K., and Wilcock, G., eds., *Dialogues with Social Robots: Enablements, Analyses, and Evaluation*, volume 427 of *Lecture Notes in Electrical Engineering*. Springer. 93–107.

Moffitt, M. D.; Peintner, B.; and Yorke-Smith, N. 2006. Multi-criteria optimization of temporal preferences. In *Proc. of CP'06 Workshop on Preferences and Soft Constraints*, 79–93.

Nielsen, J., and Levy, J. 1994. Measuring usability: Preference vs. performance. *Communications of ACM* 37(4):66–75.