User Acceptance and Plan Recognition:

Why Even Perfect Intent Inferencing Might Not be Good Enough

Harry B. Funk, Christopher A. Miller

Smart Information Flow Technologies
2119 Oliver Avenue South
Minneapolis, Minnesota, 55405-2440 U.S.A.
{hfunk,cmiller}@SIFTech.com

Abstract

Plan recognition is discussed in the framework of adaptive automation. A case is made for applying the lessons of human-human interactions, such as those found in Cockpit Resource Management, to avoid producing a correct, yet not fully trusted automation partner. Etiquette 'rules' for automation are proposed as a mechanism to achieve this.

Introduction

Our experience with plan recognition systems has been as consumers of their 'intent inferencing' conclusions (e.g., Rouse, Geddes & Curry, 1987) in the context of adaptive aiding and information systems in complex, real-world, "off the desktop" domains such as the U.S. Air Force's Pilot's Associate (Banks and Lizza, 1991) and the U.S. Army's Rotorcraft Pilot's Associate (Miller and Hannen, 1999) programs. In these environments, plan recognition is used to assess a pilot's intent in terms of a task or plan vocabulary. The inferred intent(s) are then used as the basis for adapting information presentation and automation behaviors with the goal of optimizing mission success and/or safety.

As such, the motivation for plan recognition has been to create "adaptive systems" as opposed to "adaptable" ones—these terms come from Oppermann (1994) and the chief distinction is one of who makes the system adaptations. If the user has to make his/her own adaptations, then the system is adaptable; if the system makes its own decisions about what adaptations should be made, then it is adaptive. A chief motivation behind making systems adaptive, as opposed to adaptable, is human workload savings. Oppermann found (as have others) that users of adaptable systems typically bother to exercise only a small fragment of the possible adaptations available and, as a result, frequently make due with suboptimal system configurations. The generally cited reasons for failing to use added adaptation features are the added work required either at run time to configure the system appropriately, or pre-run time in learning how to make alternate configurations. Placing the control of adaptations in the hands of an automated system that uses plan recognition to determine user intent (and therefore, needs) would seem to solve this problem.

Expectations

We have found, however, that applying sophisticated, adaptive and intelligent automation to manage information flow to human consumers in complex systems and domains is not a panacea. Users in complex, high consequence domains are very demanding and critical of automation that does not behave according to their standards and expectations, and it has proven difficult to create systems that are correct enough to achieve user acceptance. The tradeoff is not a simple two-way relationship between human workload and the adaptiveness or fit of the system to the needs of the context, as is suggested above. Instead, we have posited a three-way relationship between adaptiveness, workload and unpredictability—or the tendency for the system to do things in ways other than expected/desired by the human user (regardless of whether those ways were technically right)—as illustrated in Figure 1. An implication of this three way relationship is that it is possible to achieve a given level of adaptiveness through either an expansion in workload or an expansion in unpredictability—or various mixes in between. The spectrum of alternatives that results is roughly equivalent to the spectrum of choices that lies between adaptable/adaptive interfaces or Direct Manipulation (Shneiderman, 1997) and Intelligent Agent interfaces (Maes, 1994). Another implication is that it is probably impossible to achieve both workload reduction and perfect predictability in any system that must adapt to complex contexts.

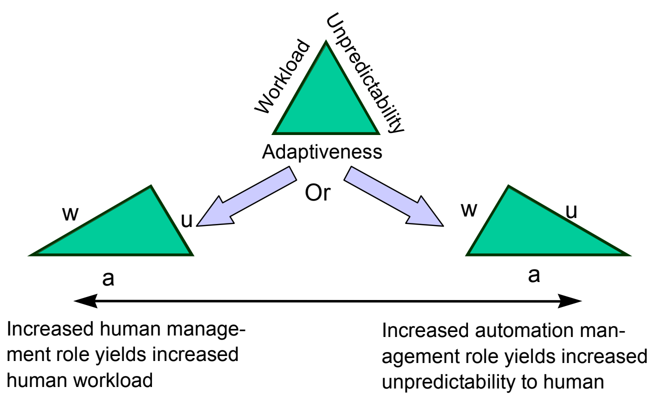

Figure 1. Conceptual view of the relationship between system adaptiveness, human workload and unpredictability.

Intent Visibility

Yet, we have found that intelligent interfaces and behaviors can be designed so that perfection is not required, but that value is still provided. Such interfaces require detailed consideration and design of the human-automation relationship. A critical mistake is attempting to make the system too autonomous in its behaviors—as, to some extent, we did on the Pilot's Associate program. While plan recognition capabilities are very useful for such systems, it is unreasonable and undesirable to cast such capabilities as the sole determiner of system adaptations (i.e., pure adaptive systems). This places plan recognition driven systems in the role of 'strong but silent' partner in the human + system team—a role which has been systematically shown to have undesirable consequences in human-human relationships in work on aviation cockpit resource management (Foushee and Helmreich, 1988).

Instead, the opportunity for explicit and dynamic collaboration about how the system may best serve the human is critical—at least to achieving trust and user acceptance, and perhaps to achieving overall acceptable levels of human + system performance as well. With hindsight it seems nearly obvious: would you (if you were piloting a fighter jet in combat) want an agent, whether human or machine, to always silently hand you the correct information or automation capability at the right time—even if it were 100% correct? Would you trust such a system, even if it had been right in the past? Wouldn't your level of trust and acceptance increase if you could communicate your intent explicitly to the system (including corrections to the system's current behaviors), and see it accept and adapt to that communication?

Or, closer to home, consider the case of an intelligent navigation aid in your car. You have just had a discussion with some third party in which you identify a need for, say, milk. Your intent changes to include a stop at the neighborhood mart. The navigation system tells you that you should turn right. Absent a visible change in display, or some explicit statement that your intent has changed, what confidence do you have that the systems directions match your current intent?

The Rotorcraft Pilot's Associate (RPA) adaptive information management system provides some insight. RPA achieved acceptable levels of usability and statistically significant workload reduction compared to an unaided condition in a series of complex and realistic human-in-the-loop mission simulations (Miller, Hannen and Guerlain, 1999). It is important to note that these results were obtained *in spite of* less than perfect tracking of the pilot's intent and pilots' reports of having to 'Now and Then' override or correct RPA's behaviors. One innovation we employed in the RPA cockpit may have influenced these results: a 'Crew Coordination and Task Awareness' display that, unlike previous some previous systems, gave the two human crew members direct insight into and some control over RPA's notion of the mission context and main tasks of each crewmember. Pilots' acceptance of this display was very high, averaging 4.25 on a scale of 1-5 where 4 corresponded to 'Of Considerable Use' and 5 to 'Extremely Useful.'

Well-Behaved Automation

The success of this interface innovation has led us to think more seriously about the implications of the associate metaphor for adaptive automation in many domains. Given our experience in working on intelligent information systems, and our familiarity with others in the literature, we have recently drafted a set of 'Etiquette Rules' for adaptive system behavior. In fact, in our initial design of the RPA Cockpit Information Manager (CIM), there were a number of etiquette-like metrics that were applied to a proposed cockpit configuration to rate it relative to other candidates. For example, it was considered "poor form", in a quantitative sense, to violate a pilot's expectations about placement of information in the cockpit. CIM could still choose to violate that expectation if the gains were sufficient, but in a sense, it paid a price. Similarly, it would be considered poor form to burst into a meeting with a customer – unless it were to announce that the building is on fire. While these metrics are in place, they are largely invisible to those that did not participate in the design or implementation, and thus have lost some of their effectiveness or power in ongoing efforts.

Revitalizing or reincarnating the design intent as the notion of 'etiquette rules', then, seems to have an appropriate focusing effect—both placing an emphasis on acceptable behavior to a human supervisor, and requiring a degree of anthropomorphic thinking about the system that seems to be productive. These rules will be presented and the general notion of human-machine etiquette will be discussed—along with additional examples from RPA concerning the quantification and tradeoff among rules implemented in that program.

References

Banks, S. and Lizza, C. (1991). Pilot's associate; A cooperative knowledge-based system application. IEEE Expert, June. 18-29.

Foushee, C. and Helmreich, R.L. (1988). Group interaction and flight crew performance In E. L. Wiener & D. C. Nagel (Eds.) Human Factors in Aviation. Academic Press, San Diego, CA.

Maes, P. Agents that Reduce Work and Information Overload. Communications of the ACM, 37(7), 1994. 31-40.

Miller, C. and Hannen, M. (1999). The Rotorcraft Pilot's Associate: Design and Evaluation of an Intelligent User Interface for a Cockpit Information Manager. Knowledge Based Systems, 12. 443-456.

Miller, C., Hannen, M. and Guerlain, S. (1999). The Rotorcraft Pilot's Associate Cockpit Information Manager: Acceptable Behavior from a New Crew Member. In Proceedings of the American Helicopter Society's FORUM 55, Montreal, Quebec, May 25-27.

Oppermann, R. (1994). Adaptive User Support. Lawrence Erlbaum: Hillsdale, NJ.

Parasuraman, R. & Riley, V. (1997). Humans and automation: Use, misuse, disuse, abuse. Human Factors, 39(2), 230-253.

Rouse, W., Geddes, N., and Curry, R. (1987). An architecture for intelligent interfaces: Outline of an approach to supporting operators of complex systems. Human Computer Interactions, 3(2), 87-122.

Shneiderman, B., Direct manipulation for comprehensible, predictable, and controllable user interfaces, Proceedings of the ACM International Workshop on Intelligent User Interfaces '97,(New York, NY, 1997) 33-39.