Collaborative Preference Elicitation in a Group Travel Recommender System

Claudia Plua[1] and Anthony Jameson[2] *

[1] DFKI, the German Research Center for Artificial Intelligence
[2] DFKI / International University in Germany

Abstract. When a recommender system requires users to specify their preferences explicitly and in detail, this process can become so tedious or impractical that the system is essentially inaccessible to some users. And where the recommendation is being made to a group, inaccessbility to one or more members may preclude use by the group as a whole. The present report describes work currently in progress on a way of facilitating preference specification which is especially applicable when a group of users is communicating asynchronously. Our preference elicitation interface explicitly encourages users to help each other with the preference elicitation task and to benefit from work that has been done by other members. The basic features of this approach are illustrated with examples, and some initial user testing is summarized.

1 Facilitating Explicit Preference Specification

In some types of recommender systems, users are required to specify their preferences explicitly and in detail (see, e.g., Pu & Faltings, 2000, and the discussion of Tête-à-Tête in Maes, Guttman, & Moukas, 1999). This type of preference elicitation is often unnecessary if recommendations are to be based solely on the user's general tastes, which may have been manifested in simpler ratings of objects, as in collaborative filtering. But when the user is making an important decision about a relatively complex object—such as an automobile or a vacation—he or she may have preferences and priorities that apply to this particular decision and may be only loosely related to longer-term preferences.

One obvious problem with explicit, detailed preference elicitation is that it can be tedious and time-consuming. Another problem is brought to the foreground by the recent attention that has been given to the goal of *universal accessibility* (see, e.g., Stephanidis & Savidis, 2001): Some users may find it difficult or impossible to use such a recommender system at all, because of a lack of general computer sophistication, understanding of the domain in question, and/or time to deal with the complexities presented by the system.

One possible approach to this problem is to tap a resource that has so far been underutilized in recommender systems for preference elicitation: other persons who are

* The research described here is being supported by the German Ministry of Education and Research (BMB+F) under grant 01 IW 001 (project MIAU).

in a good position to help the current user specify her preferences because they possess more relevant knowledge and/or time than the current user.

A scenario in which it is particularly natural for users to help each other in this way is one in which a group of individuals who know each other are specifying their preferences concerning a decision that they intend to make together. For concreteness, consider the case of a geographically distributed family that is planning a vacation that they will take together. A recommender system should allow each member to specify preferences concerning numerous aspects of the vacation, ranging from the nature of the location to the specific facilities available there. Suppose that synchronous communication (e.g., via telephone or an on-line chat system) is impractical. Each member will then have to specify their preferences asynchronously. Suppose that at least one of the family members, to be called $\mathcal{A}$, would have difficulty specifying her preferences unaided. One or more of the other family members ($\mathcal{B}$) may be in an especially good position to help $\mathcal{A}$:

- The preferences of $\mathcal{A}$ and $\mathcal{B}$ may be largely similar, so that $\mathcal{A}$ can get by largely by copying $\mathcal{B}$'s preferences.
  In fact, $\mathcal{A}$ may be willing to copy substantial parts of $\mathcal{B}$'s preferences without even checking them in detail.
- Even if $\mathcal{B}$'s own preferences differ from $\mathcal{A}$'s, $\mathcal{B}$ may know $\mathcal{A}$ well enough to be able to make largely accurate suggestions for $\mathcal{A}$, which $\mathcal{A}$ then needs only to check and perhaps modify.
  $\mathcal{A}$ may trust $\mathcal{B}$ to the extent of being willing to adopt many of $\mathcal{B}$'s suggestions without careful checking.
- The process of communicating about preferences in these ways, in addition to increasing accessibility, can have the advantage of increasing the users' understanding of each others' preferences – which can be useful during a phase of negotiation about compromise solutions.

In a situation like this, it would be natural for the group members to help each other with the preference specification process even if the system offered no specific support for such helping. For example, $\mathcal{A}$ might simply ask $\mathcal{B}$ to do the preference specification for her, perhaps checking the results before submitting them. But there are various ways in which a system can make this type of helping more efficient, reliable, and attractive to users. We are exploring such methods in the research described in this paper.

2 Exploratory User Studies

In accordance with the principles of user-centered design, we began at an early stage presenting potential users of the travel planning system with mockups of possible user interfaces. The first issue of interest was whether our idea of helping users to help each other with preference specification would be well received by typical users. The persons we interviewed consistently confirmed that this basic conception seemed attractive to them—even if they were skeptical about the whole idea of planning a vacation through asynchronous communication. For example, within a family that comprised two parents, a 12-year-old boy, and a 6-year-old girl, we interviewed the mother and the son. They surprised us by volunteering the opinion that even the 6-year-old girl would be

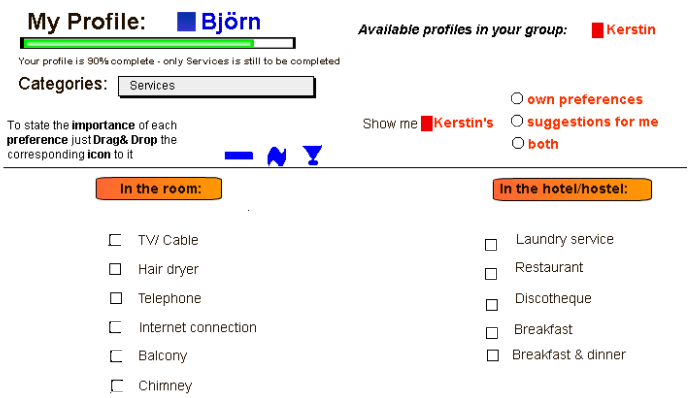

Fig. 1. *Björn's preference specification screen before invocation of Kerstin's preferences.*

able to specify her relevant preferences via a suitable version of the interface, if the older family members were given the opportunity to support her asynchronously.

Having established that the basic conception is in principle acceptable to users, we turned to the tricky issue of how to design the interface so that it really does support collaborative preference specification. (Note that a poorly designed interface could introduce complications that might outweigh any benefits of the collaboration.) Building on the experience of the first interviews with users of mockups, we proceeded to observe 4 subjects working with screens of a simple prototype, tracking their eye movements with a remote eye tracker. Whenever the trace of a subject's eye movements (and/or their retrospective comments when viewing the trace) convinced us that some aspect of the interface was too complex or confusing, we changed that aspect of the interface.

3 Example Screens

Figures 1 shows one of the screen designs that has resulted from these tests—though the design will probably continue to be changed as new functionality is added and further tests are conducted. The current user is Björn, the boyfriend of Kerstin, who had previously used the interface to specify her preferences. Björn could simply specify his own preferences in the lower part of the screen: For each service he wanted, he could drag and drop one of the three blue importance icons ($-$, $\sim$, and !) to the left of the service, thereby also causing the checkbox to be checked. For example, by dragging $\sim$ to the left of "Internet connection", he could specify that he considered an internet connection to be moderately important.

But as is indicated on the upper right in the screen, he can also use Kerstin's preferences and/or Kerstin's explicit suggestions for him as a starting point. In Figure 2, Björn has asked for Kerstin's preferences to be shown. The importance icons that Kerstin specified for herself are now shown in Kerstin's color (red). Also, Kerstin's preferences are copied as preferences for Björn, so that they will apply to him unless Björn changes them individually (or removes them all). In this particular example (see Fig-

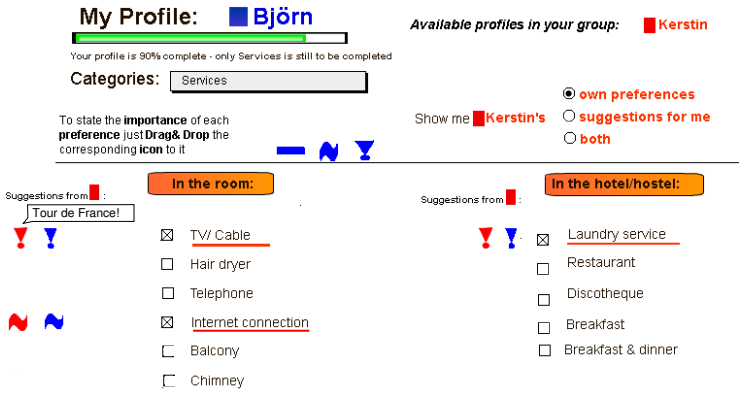

Fig. 2. *Björn's preference specification screen with Kerstin's preferences specified as defaults.*

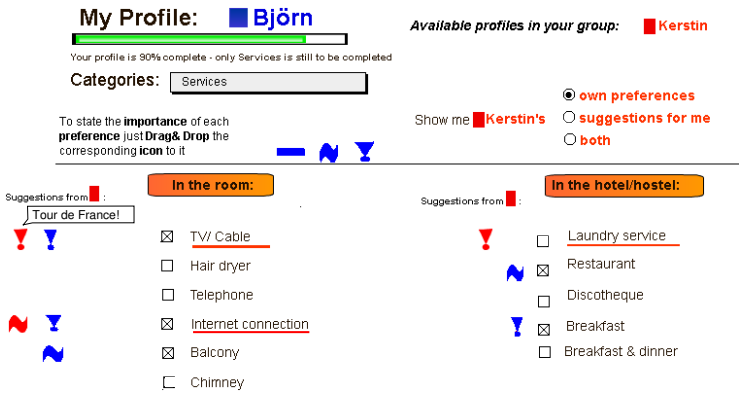

Fig. 3. *Björn's preference specification screen after editing by Björn himself.*

ure 3), Björn leaves only one preference of Kerstin's untouched: the strong preference for "TV / Cable". Here, he is influenced in part by the textual explanation that she added to the specification of her preference, which reminds Björn that the Tour de France will be broadcast during the period of the vacation. (This explanation is actually displayed only when Björn moves the cursor over the exclamation mark.) Björn makes changes to the other two preferences of Kerstin: He increases the suggested importance of "Internet connection" to "!", and he removes the suggested preference for "Laundry service". In addition, he specifies three preferences that Kerstin didn't specify: for a balcony, a restaurant, and included breakfast.

The benefit of being able to use Kerstin's preferences as a starting point would of course have been greater in this case if Björn had been willing to adopt more of them. In the extreme case in which Björn trusts Kerstin implicitly with regard to "services", Björn can even *link* Kerstin's preferences to his own (a feature not visible on the example screens): In this case, any future changes that Kerstin might make in her preferences will be copied into Björn's profile.

There are many possible alternative screen designs for the functionally shown in Figures 1–3, and some aspects of this particular design may appear to be simplistic and/or redundant (e.g., the use of red underlining to draw additional attention to Kerstin's preferences). But this particular design is in fact the result of detailed testing with eye tracking, which revealed the need for simplicity and redundancy. The claim is not that this particular screen is really optimal but rather that the effective design of screens for this type of preference elicitation (and probably other types as well) requires careful attention and user studies.

4 Other Types of Support

The examples just given have illustrated quite concretely several ways in which one user $\mathcal{B}$ can help another user $\mathcal{A}$ with preference specification. Other methods that we are implementing include the following:

- $\mathcal{B}$ can ensure in advance that $\mathcal{A}$ will skip over entire portions of the preference specification procedure that are of no relevance to $\mathcal{A}$.
  For example, entire groups of preference questions may be of no relevance to children; eliminating these from the start might make the preference specification process accessible to children even though it would not otherwise be manageable for them. This type of filtering can often be done partly or fully automatically, even on the basis of a simple user model; but it may still be useful to have a human involved.
- $\mathcal{B}$ can set some interface parameters that increase $\mathcal{A}$'s ability to deal with the preference specification process,
  Parameters such as interaction style and density of information presentation can have a large impact on the usability of a system for a given user, but they can be hard for less sophisticated users to specify themselves—or for the system to adapt automatically to the individual user.

5 Related Work

The idea of helping computer users to help each other has appeared previously in many contexts. Various systems help a user to identify and contact experts or peers who can help them with a particular problem (see, e.g., Bull, Greer, McCalla, Kettel, & Bowes, 2001). Relative to these systems, our approach is novel in its emphasis on specific interface techniques that facilitate collaboration among persons who already know and trust each other and who may resemble each other in useful ways.

Many systems support the synchronous helping of remote users (see, e.g., Aberg & Shahmehri, 2001), for example by computer support personnel who may even log onto a user's remote computer to help them out with a problem. This type of synchronous help, while presenting many technical challenges, is relatively easy in the sense that the helper can straightforwardly communicate with the user being helped (e.g., via an audio channel), explaining what he is doing and asking the user to approve actions where appropriate. Such meta-level communication requires more explicit support where asyn-

chronous communication is involved, as is illustrated by the examples given earlier in this paper.

Asynchronous meta-level communication has been supported in various collaborative authoring systems (see, e.g., Brush, Bargeron, Grudin, & Gupta, 2002, for a recent discussion). Indeed, some of the mechanisms embodied in our system were inspired by functions provided by collaborative authoring systems; but the content and the function of the communication are different.

Turning to related work that concerns preference elicitation, we see that there exist a number of systems that elicit preferences of members of a group so as to be able to make recommendations to the group as a whole (see, e.g., McCarthy & Anagnost, 2000; O'Connor, Cosley, Konstan, & Riedl, 2001). But the emphasis here has been on ways of arriving at recommendations that satisfy all of the group members adequately once their preferences have been specified, not on facilitating the preference specification in the first place.

Terveen, McMackin, Amento, and Hill (2002) have recently proposed a novel way of giving users a starting point for preference specification, making it unnecessary for them to specify everything from scratch: Their system supplies a default set of specifications that is based on a record of the current user's past behavior (e.g., the types of music that they have listened to). While the motivation and even some aspects of the user interface as similar to those in our system, the information on which the initial proposal is based is quite different, and it could not in general not be a substitute for the help that can be provided by collaborating users.

6 Conclusions

Supporting preference elicitation through collaboration appears to be a promising approach for the recommendation of complex products to distributed, heterogeneous groups. Whereas the presence of several decision makers might be expected to make the recommendation task more complicated than it is for a single decision maker, we have seen that the opposite may be the case: Decision makers who might otherwise hesitate to use any but the simplest recommender system may find that they can participate in a fairly sophisticated decision process with the help of their fellow decision makers. This natural type of collaboration should not be left to chance, especially when the users are geographically distributed and have limited communication channels.

References

Aberg, J., & Shahmehri, N. (2001). An empirical study of human web assistants: Implications for user support in web information systems. In J. A. Jacko & A. Sears (Eds.), *Human factors in computing systems: CHI 2001 conference proceedings* (pp. 400–411). New York: ACM.

Brush, A. J., Bargeron, D., Grudin, J., & Gupta, A. (2002). Notification for shared annotation of digital documents. In L. Terveen, D. Wixon, E. Comstock, & A. Sasse (Eds.), *Human factors in computing systems: CHI 2002 conference proceedings* (pp. 89–96). New York: ACM.

Bull, S., Greer, J., McCalla, G., Kettel, L., & Bowes, J. (2001). User modeling in I-Help: What, why, when and how. In M. Bauer, P. Gmytrasiewicz, & J. Vassileva (Eds.), *UM2001, User Modeling: Proceedings of the Eighth International Conference* (pp. 117–126). Berlin: Springer.

Maes, P., Guttman, R. H., & Moukas, A. G. (1999). Agents that buy and sell. *Communications of the ACM*, *42*(3), 81–91.

McCarthy, J. F., & Anagnost, T. D. (2000). MUSICFX: An arbiter of group preferences for computer supported collaborative workouts. In *CSCW 2000 video program.* New York: ACM.

O'Connor, M., Cosley, D., Konstan, J., & Riedl, J. (2001). PolyLens:A recommender system for groups of users. In *Proceedings of the European Conference on Computer-Supported Cooperative Work.*

Pu, P., & Faltings, B. (2000). Enriching buyers' experiences: The SmartClient approach. In T. Turner, G. Szwillus, M. Czerwinski, & F. Paternò (Eds.), *Human factors in computing systems: CHI 2000 conference proceedings* (pp. 289–296). New York: ACM.

Stephanidis, C., & Savidis, A. (2001). Universal access in the information society: Methods, tools and interaction technologies. *Universal Access in the Information Society*, *1*(1), 40–55.

Terveen, L., McMackin, J., Amento, B., & Hill, W. (2002). Specifying preferences based on user history. In L. Terveen, D. Wixon, E. Comstock, & A. Sasse (Eds.), *Human factors in computing systems: CHI 2002 conference proceedings* (pp. 315–322). New York: ACM.