Available online at www.sciencedirect.com
SCIENCE DIRECT®
Int. J. Human-Computer Studies 64 (2006) 27–35
International Journal of Human-Computer Studies
www.elsevier.com/locate/ijhcs

Personal assistants: Direct manipulation vs. mixed initiative interfaces

Marcelo Armentano[a,b,*], Daniela Godoy[a,b], Analía Amandi[a,b]

[a]ISISTAN Research Institute, UNICEN University, Campus Universitario, Paraje Arroyo Seco, CP 7000, Tandil, Bs. As., Argentina
[b]Consejo Nacional de Investigaciones Científicas y Técnicas Avda. Rivadavia 1917-CP C1033AAJ Cdad. de Buenos Aires, Argentina

Received 28 October 2004; received in revised form 31 March 2005; accepted 1 June 2005
Available online 21 July 2005
Communicated by C. Sierra

Abstract

Interface agents are computer programmes that provide assistance to users dealing with computer-based applications. The introduction of agents to user interfaces caused the exploration of new metaphors to enhance user ability to directly manipulate interfaces. In this regard, mixed-initiative interaction refers to a flexible interaction strategy in which agents contribute with users by providing suitable information at the most appropriate time. Mixed-initiative approaches promise to dramatically enhance human–computer interaction by allowing agents to resemble human assistants. In this paper, we report a study on how the interaction metaphor can affect the user perception of agent capabilities and, in turn, the final success of agents.
© 2005 Elsevier Ltd. All rights reserved.

1. Introduction

Interface agents, also known as personal agents, are autonomous software entities that provide assistance to users dealing with computer-based applications, such as information filtering, meeting scheduling, entertainment selection and so forth. These agents act as human assistants, collaborating with the user in the same work environment and becoming more efficient as they learn about user interests, habits and preferences.

From human–computer interaction perspective, the emergence of interface agents changed the traditional interaction metaphor of direct manipulation to a complementary style of interaction, which has been referred to as indirect manipulation or mixed-initiative interaction. Instead of user-initiated interaction via commands and/or direct manipulation, the user is engaged in a cooperative process in which both human and software agents initiate communication, monitor events and perform tasks (Maes, 1994).

Although an agent-based interface should support completing cooperative tasks based on the user delegation of tasks to agents, users may be unaware of the actions agents are performing in order to make it possible for the overall system to provide its functionality. The same software can be presented to users with a direct manipulation interface where an agent is acting in background to fulfill agent-related application requirements or with a mixed-initiative interface where synthetic, personified or even animated agents interact with users.

Having an agent operating directly in the user interface rather than as a background process increases the extent to which the user perceives the software as acting like an assistant (Lieberman, 1997). This paper discusses an empirical study which was undertaken to investigate how the perception of agents in a mixed-initiative interaction interface influences users and how this can impact on the learning process. In pursuing this goal, we carried out an experiment with real users interacting with a personal agent by means of interfaces corresponding to direct manipulation and mixed initiative interaction metaphors.

This work is organized as follows. Section 2 discusses adaptive systems and their relation with agent-based interfaces and human–computer interaction issues. Section 3 places this work in the context of related ones. Section 4 describes the study we performed and summarizes the

*Corresponding author. ISISTAN Research Institute, UNICEN University, Campus Universitario, Paraje Arroyo Seco, CP 7000, Tandil, Buenos Aires, Argentina. Tel./fax: +54 2293 440362.
E-mail addresses: marmenta@exa.unicen.edu.ar (M. Armentano), dgodoy@exa.unicen.edu.ar (D. Godoy), amandi@exa.unicen.edu.ar (A. Amandi).

1071-5819/$ - see front matter © 2005 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ijhcs.2005.06.001

achieved results. Concluding remarks and future lines of research are given in Section 5.

2. Personal agents as adaptive systems

Adaptive systems are characterized by the presence of explicit user models or profiles representing user knowledge, goals, interests and other features that enable the system to distinguish among various users (Brusilovsky and Maybury, 2002). An adaptive system collects data for the user model from various sources including implicit observation of user interactions and/or explicit asking of user judgments. Finally, user models are used to provide an adaptation effect, that is, to tailor interaction to different users in the same context or application.

Personal agents are adaptive systems in which the user model is used to make suggestions, correct misconceptions, and guide agent actions in a broad sense (Maes, 1994). Even when these agents may serve a variety of purposes, all of them provide proactive support to users and operate to some degree autonomously. In consequence, agent interfaces are inherently delegative since users turn things over to their agents to be done rather than do them by themselves. Typically, agents assist users by making suggestions in a non-invasive fashion, whereas decisions are still under the control of users.

In a graphical user interface, two main approaches can be distinguished in relation with the presentation of agents. The first consists in adapting the interface of the base application and controlling the agent using direct manipulation. The second focuses on making the agent explicit with a representative figure, which can enhance the level of engagement of the user in a mixed initiative style of interaction.

In the first approach, the effects of adaptation can be visualized in a variety of ways. For example, consider an adaptive Web system (Brusilovsky et al., 1998), in which adaptation can be materialized by three main techniques:

- *Adaptive content selection*: When the user searches for relevant information, the system can adaptively select and prioritize the most relevant items, as in *Letizia* (Lieberman, 1995) and *PersonalSearcher* (Godoy and Amandi, 2000).
- *Adaptive navigation support*: When the user navigates from one item to another, the system can manipulate the links to provide adaptive navigation support (agents like *Syskill & Webert* (Pazzani et al., 1996) and *WebWatcher* (Armstrong et al., 1995) exemplify this case).
- *Adaptive presentation*: When the user gets to a particular page, the system can present its content adaptively, as in *NewsAgent* (Cordero et al., 1999).

All referenced systems share the feature of adapting themselves according to the user model. Presentation and interaction mechanisms in these agents are always interweaved with the base application and are limited to highlighting links or presenting recommendations in a separate window.

In the second approach, the user retains both full control over the direct manipulable interface and interacts with agents acting autonomously to perform tasks on behalf of users. Although agents are intended to execute tasks in a relatively autonomous fashion, users can customize agent functionality as well as assign, suspend, resume and cancel delegated tasks.

When agents show themselves with an external figure, they are called synthetic, personified or anthropomorphic agents. This representation of agents create engaging environments for users by introducing life-like characters into the interface. Furthermore, they provide the illusion of autonomous, animistic entities with human-like capacities. It is worth noticing that the mere fact of adding a face to the agent does not increase its acceptance. For example, Microsoft's paper-clip assistant,[1] have shown being inefficient and users prefer not to use it since is constantly interrupting users irrespective of their focus of attention (Schiaffino and Amandi, 2004). Further examples of personified agents can be found in Maes and Kozierok (1993), Cypher (1993) and Okonkwo and Vassileva (2001).

Personified agents have the ability of accepting orders from human users and answering in a personalized way. Moreover, these agents can pro-actively realize when they can help their users and either suggest an action or directly act on behalf of the user in a given situation. These systems respond to a mixed-initiative interaction metaphor and consider the pair user–agent as a relation in which the control shifts between the two according to the situation, the shared knowledge and the user model (Cesta and D'Aloisi, 1999). The main idea of mixed-initiative agents (either personified or not) is to develop an active, cooperative, and adaptive system, in which users retain control over agents. In many cases, the acceptance of the system strictly depends on the effective interweaving of the user–agent system as a whole. Examples of this kind of agent can be found in Gruen et al. (1999), Rich and Sidner (1998) and Koda and Maes (1996).

3. Related work

There was an interesting debate regarding the advantages and disadvantages of intelligent agents and direct manipulation (Shneiderman and Maes, 1997). User–agent relationship is conditioned by human–computer interaction issues (Norman, 1994). HCI people have criticized agent-based methodologies that seem to produce systems not easily accepted by users, mainly because of the loss of control caused by the autonomy of the agents. In this regard, some aspects to be considered in order to increase

[1] See *Microsoft's paper-clip assistant killed in Denver* at http://www.cnn.com/TECH/computing/9810/16/clipdeath.idg/; *Born again: Clippy pops up in Office XP* at http://zdnet.com.com/2100-11-267631.html and http://www.winsupersite.com/reviews/office10-whatsnew.asp.

acceptability include: giving users time for trusting their agents, allowing users to resume control, and leaving users the possibility of verifying what agents are doing.

In Shneiderman and Maes (1997) we can find a debate about these issues. However, the question of whether software agents should be presented to the user in the interface of a software application is still open. In Koda and Maes (1996) an experiment was presented using a Web-based poker game in which four computer poker-playing agents with different facial representations and the user played against one another. The results showed that having faces and facial expressions is likable and engaging, but it takes the user more effort to interpret the meaning of the expressions.

In this paper, we present an experiment which was focused on the effect that presenting explicitly the agent to the user has in the user perception of software capabilities. Our experiment is similar in structure, but different in the addressed hypothesis, to that presented in Okonkwo and Vassileva (2001), in which authors investigate the impact of integrating personified pedagogical agents with an emotional model, personality traits and affective reasoning on the learning experience of students. The result of that experiment showed that participants found the anthropomorphic agent motivating and enjoyable. However, the emotional pedagogical agent had no impact on the learning performance of users. Consistently, the studied taken in Mulken et al. (1998) showed that the presence of an agent has no significant impact on the understanding of users when technical explanations are being presented.

A more recent study (Hongpaisanwiwat and Lewis, 2003) investigated the use of an animated character in the role of presenter in a multimedia presentation. Its objective was to determine to what extend animated characters might hold the attention of students during a presentation to enhance their comprehension. The study used two different characters: an anthropomorphic character in the form of a person, and a non-anthropomorphic character represented by a finger. It was found that the presence of an animated character as a presenter had no impact on the comprehension of the presentation.

4. Empirical study

In this study, we analysed a critical aspect of user–agent interaction such as the consequences of different manipulation metaphors in user perception of software. Particularly, the level of satisfaction of users regarding software and their preferences concerning manipulation metaphors were examined under both metaphors. Our work hypothesis is described in Section 4.1. As the main purpose of our study, described in Section 4.2 was to understand the effect of synthetic agents in user-interfaces, the agent employed to carry out the experiment is presented in Section 4.3. The interfaces that users employed to access its functionality are detailed in Sections 4.4 and 4.5. Finally, Section 4.6 reports the results we obtained.

4.1. Work hypothesis

The hypotheses tested in this work are that adding an explicit agent to a software application will cause

- An increase on user expectations regarding the software, since users are aware of the agent existence they might expect from it the same kind of help that a human assistant can provide.
- Inferior level of satisfaction of users regarding the application if it does not meet the expectations.
- Inferior level of engagement if users are not completely satisfied with the received assistance.

We claim that a change in the manipulation metaphor can cause an increase in the user expectation related to the interaction with a personal agent. High expectations are in most cases fostered by anthropomorphic, personified and even animated agents which imitate human behaviours and, consequently, are expected to produce the same kind of results. These agents can even use suitable gestures or facial expressions to express their suggestions.

4.2. Experiment description

In order to prove our hypothesis, we carried out an experiment with real users in which they were asked to interact with a same application accessible through two different interfaces in contrasting paradigms: a traditional direct-manipulation interface where users have complete control over interaction while the application access to the agent functionally; and a more flexible interaction strategy given by a mixed-initiative interface where the agent contributes to the task being performed by the user in the application.

Both interfaces were designed in the context of a Web search application in which an agent, called *PersonalSearcher*, assists users by tailoring search results according to their interests. In this case, the application functionality relates specifically to searching documents on the Web which includes posting queries to one or more search engines, fusing results and presenting a unified view to users. In this application, the agent functionality consists in learning user interests by observing user browsing behaviour, building a user profile modeling these interests and assessing the relevance of Web pages according to the user interests in order to suggest the best pages to users.

Forty-two students of an introductory course of agent technologies, equal number of male and female with age ranging from 21 to 28, took part in this experiment. Few of them had some previous experience using interface agents, and none of them had used *PersonalSearcher* before. Participants were randomly divided into two even groups. The first group was provided with access to the application with a direct-manipulation interface, whereas the second group had access to the mixed-initiative interface of the same application in which users interact with a synthetic

agent appearing in the screen. All participants used the different versions of *PersonalSearcher* during 3 weeks. They were allowed to follow any specific searching scenario, i.e. they were free to search for topics according to their own interests. At the end, participants filled a survey with a number of multiple choice questions regarding their experience. Not to influence their answers to the survey, users were not aware about the existence of the other version of the software.

The survey contained subjective questions concerning the degree of fulfillment with the software functionality, including a learning time measure (*It did not learn at all*, *It learned something*, *It learned quite well*, *It learned a lot*, or *It learned too much*), if suggestions were good enough (*Yes*, *To some extend*, *No*), satisfaction with the software functionality (*Satisfied*, *Quite Satisfied*, *Disappointed*), and if their notice any improvement in software performance as they used it (*Yes*, *Just a little*, *No*). Finally, a negative formulated question inquired if the user would prefer the other interaction metaphor. The first group was asked if they would prefer to ask for suggestions instead of receiving them automatically by filtering not recommended pages. On the other hand, the second group was asked if they would prefer not to have the agent, but receiving suggestions from the software directly.

4.3. PersonalSearcher

PersonalSearcher (Godoy and Amandi, 2000) is an intelligent agent that learns about user interests by observing user behaviour while users are carrying out regular activities on the Web. By a content-based analysis of the information extracted by observation, this agent is able to deduce the topics a user is interested in to create a user profile. Fig. 1 depicts the complete agent functionality.

PersonalSearcher carries out a parallel search in the most popular search engines and filters the resultant list of pages according to profiles it builds based on the observation of user browsing on the Web. For each reading in the standard browser the agent observes a set of indicators in order to estimate the user interest in a given Web page. By means of this mechanism, each instance of this agent obtains pages relevant to a user without distracting him from his regular activities.

To build a user profile, pages considered interesting to the user are taken as input to a clustering algorithm, which output is a user interest hierarchy (Godoy and Amandi, 2005). This hierarchy models user interests in several domains (e.g. *sports*, *finances*, etc.) and at different level of abstraction (e.g. *tennis* and *football* within *sports*). A hierarchical organization of user interests not only enhances the semantic of user profiles as it is much closer to the human conception of a set of interests, but also enables agents to have a temporal view of such interests. This is, even when some interests are expected to change over time, users frequently show a certain persistence in other interests. Therefore, interests at the top levels of the hierarchy can be seen as long-term interests, while those at the bottom levels can be seen as short-term interests.

In order to learn profiles, the agent interacts with a user to capture experiences of user interests. Fig. 1 depicts an example of an experience in *PersonalSearcher*. Each experience encapsulates both specific and contextual knowledge that describes a particular situation denoting a user interest, such as the visit to a Web page which was relevant to the user. Experiences can be divided into three main parts: the description of the Web page content, the description of the situation during which it was captured, and the outcome of applying this experience to personalization.

For representing page contents, a bag-of-words approach is used. Thus, pages are identified by term vectors in a space in which each dimension corresponds to a distinct term associated with a weight indicating its importance. An experience also describes the contextual information of the situation in which it was captured, encompassing the URL,

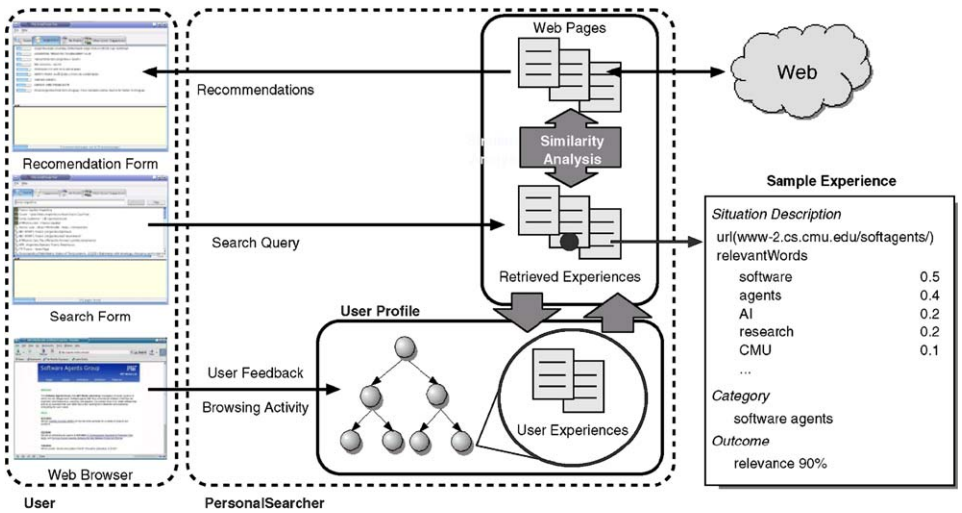


Fig. 1. *PersonalSearcher* functionality.

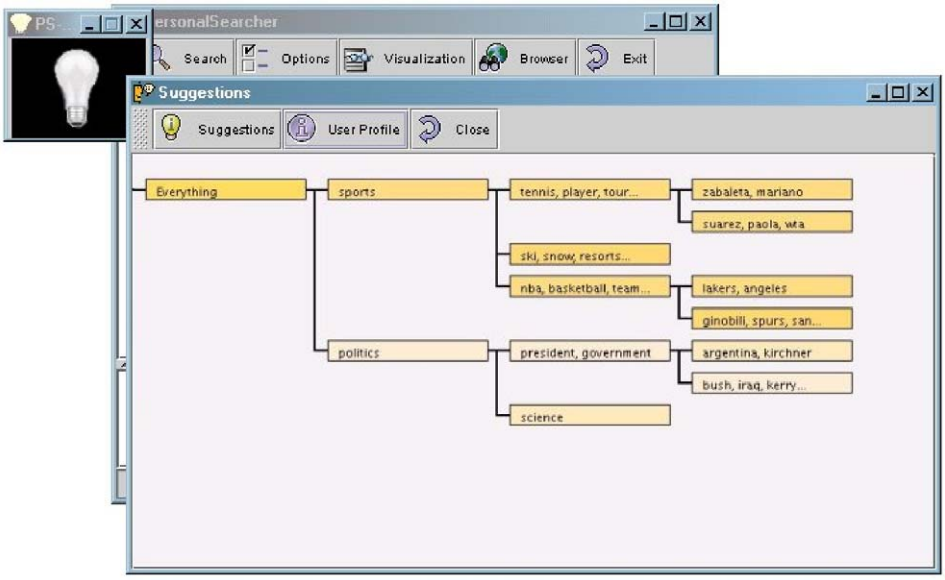


Fig. 2. Example of user interest hierarchy.

date and time the experience was registered and, additionally, the level of interest the user showed in the page according to an agent criteria. The criteria used by *PersonalSearcher* include the time spent reading a Web page in regards to its length, the amount of scrolling in a page, and whether it was added to the list of bookmarks. Finally, an experience also registers the user feedback to actions carried out by the agent based on the knowledge it provides about the user interests. The right-hand side of Fig. 1 shows an example of an experience.

Experiences of user interests obtained by extracting feature vectors from Web pages are incrementally presented to a clustering algorithm, which is concerned with conforming hierarchies of concepts or categories starting from them. These hierarchies are classification trees where internal nodes represent concepts and leaf nodes represent clusters of experiences. A hierarchy root corresponds to the most general category within the user interests, which comprises all experiences the algorithm has seen, while inner concepts become increasingly specific as they are placed lower in the hierarchy, covering only subsets of experiences by themselves. In turn, terminal concepts are those with no further child concepts. Fig. 2 shows an example a hierarchical clustering solution constituting a user profile.

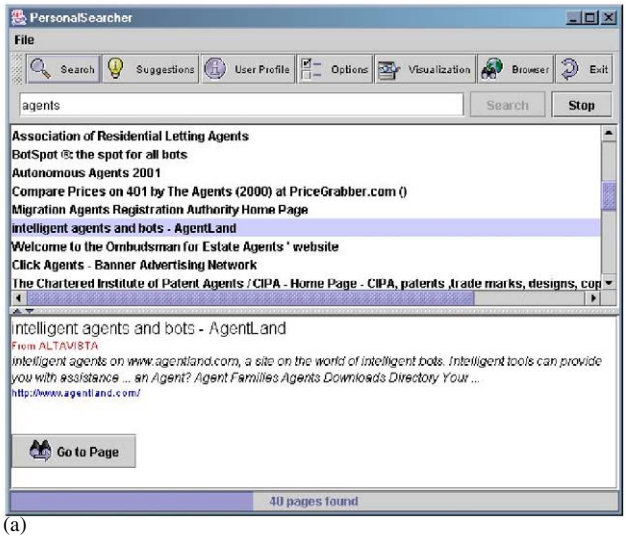
The clustering algorithm allows agents to incrementally acquire profiles without an a priori knowledge of user interest categories, so that the learning process is completely unsupervised. In addition, it belongs to the conceptual clustering paradigm which includes not only clustering, but also characterization i.e. the formation of intentional concept descriptions for each extensionally defined cluster. In consequence, the clustering process offers comprehensible profiles that can be easily interpreted by both users and other agents.

Users interact with *PersonalSearcher* expressing their information needs by keywords. The agent posts these queries to the most popular search engines, receiving documents that cover a wide portion of the Web. *PersonalSearcher* determines the convenience of suggesting a Web page to the user by computing its relevance degree regarding to the user interest hierarchy. Those pages that exceed a user relevance threshold as regards to some category in the profile are sent to the user as a result of his query. *PersonalSearcher* allows the user to customize the desired level of assistance at any moment, by adjusting the relevance a suggestion should have from the graphical user interface (GUI). Once the agent has presented some suggestions, the user behaviour is again observed to perform adaptations of the profile in terms of the user approval to the agent suggestions.

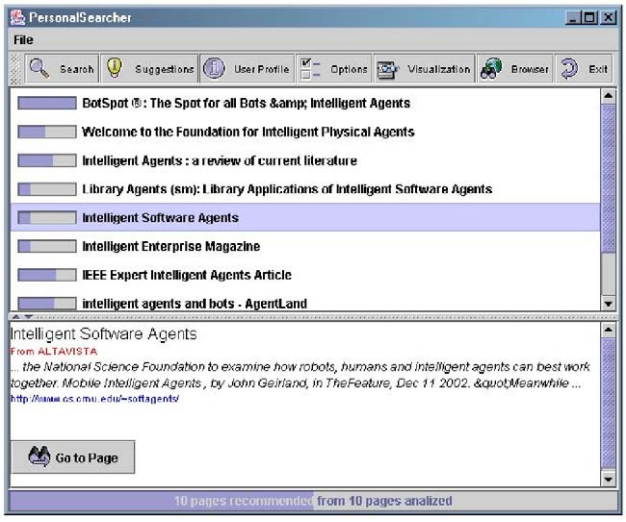
4.4. Direct-manipulation interface

In the direct-manipulation interface of *PersonalSearcher* both search results of a Web search and suggested pages are presented to the user after posting a query without explicit requirement. Fig. 3(a) illustrates how search results are presented, whereas Fig. 3(b) depicts a list of Web pages the agent found relevant to the user out of those in the search results. Both options are directly accessible from a tool-bar in the upper part of the interface. The results of the query as well as the suggestions are presented along with a short description of the page, the search engine it was retrieved from, and a button to open the Web browser with the desired page. Each suggestion also displays a small percentage bar indicating the confidence of the agent in it based on the user profile.

In this interface, users are not aware of the actions the agent is performing in order to learn about their interests



(a)



(b)

Fig. 3. *PersonalSearcher* screenshots in the direct-manipulation interface: (a) Web search results and (b) Suggested Web pages.

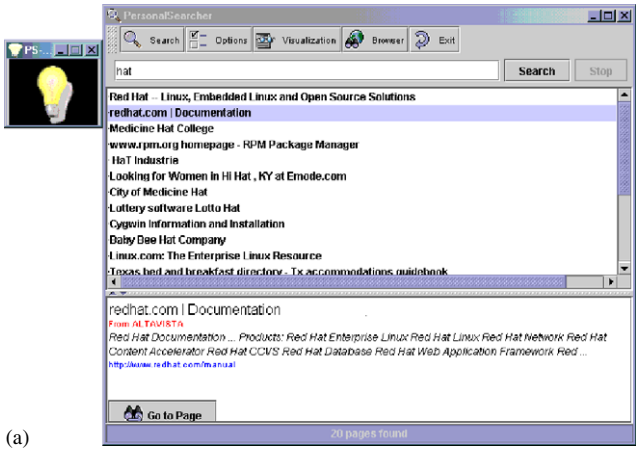
and personalize the search results. However, the agent is acting in background to create the user interest hierarchy and filtering the search results.

4.5. Mixed-initiative interaction interface

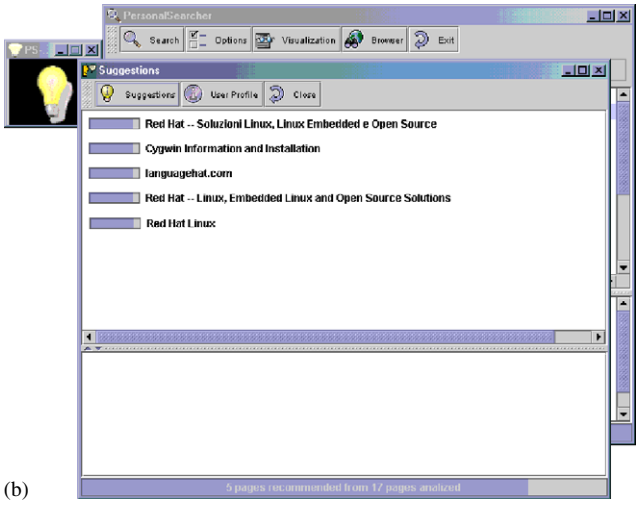
In the mixed-initiative interaction interface, a synthetic agent is displayed in the interface so that suggestions are only accessible through user–agent iteration. Initially, users can only enter queries to obtain search functionally. A small window with a lamp image and soft colors which is shown in the upper corner of the application represents the agent. Even though users are not aware of the actions the agent is performing in background, they have to explicitly ask for agent functionality by clicking on the image of the agent to ask for suggestions. Thus, users can seen suggestions as an *extra* functionality of the software application.

Human beings tend to pay attention to changes in the mobility, volume and contrast during interactions. Nevertheless, the techniques employed in this process have to be carefully designed not to disturb the user when there is no interaction with agents. In this experience, the agent is limited to suggest interesting pages to users so that it does not possess further ways of attracting user attention, more than "lighting up" the lamp. This is the reason for choosing soft gray colors to represent the agent when it does not have suggestions to make.

Like in the direct manipulation interface, in parallel to the search process the agent analyses Web pages which are gathered from search engines and measures their relevance according to the user profile. As soon as the agent finds some interesting Web pages to recommend, it will try to attract user attention by lighting up the lamp. Fig. 4(a) shows this situation. In contrast to the direct manipulation interface in which suggestions are given to users once the query is posted, in this interface users remain in control of the interaction. They can continue working with the searching application, ignoring the agent, or they can ask the agent to display its suggestions as it is shown in Fig. 4(b). The format of both results and suggestions is the same



(a)



(b)

Fig. 4. *PersonalSearcher* screenshots in the mixed-initiative interface: (a) Web search results and (b) Suggested Web pages.

as that of the direct manipulation interface. However, the method to access the suggestions varies.

4.6. Experiment results

Participants in both groups responded to a survey with questions about their experience with the application. Mainly, questions pursued the goal of assessing the level of satisfaction of users and analysing the preferences of users in regards to the interaction metaphors.

Analysing user answers, we found that 80% of the users using the mixed initiative iteration interface thought that the agent learned to make suggestions quite well. However, part of the users using the direct-manipulation interface thought the application learned a lot (21.4%) and even too much (14.3%). In both interfaces users can explore the user interest hierarchy which constitutes their profiles in order to determine if the model matches their real interests. Fig. 5(a) plots the survey results regarding this issue. Consistently, 75% of the users of the mixed-initiative interaction interface also responded that the suggestions they received were good but only to some extended, or even they were no good enough (5%), whereas an important proportion of users in the other group qualified suggestions as very good (21%). Fig. 5(b) shows the proportion of users according to their answers and the group they belong to.

With respect to suggestions, users were also asked whether they perceived any change on suggestion accuracy over time. Both groups shared the same learning and suggesting mechanisms so that the application was expected to behave similarly under similar circumstances. Fig. 5(c) depicts answers regarding this issue. Surprisingly, a high percentage (28.6%) of users dealing with the direct manipulation interface did not perceive changes on suggestion accuracy over time. Although it was clear from previous answers that these users considered they had received good suggestions, they also claimed to have received similar suggestions from the beginning. In contrast, users using the mixed-initiative interaction interface reported to have noticed some improvement in agent suggestions.

A more general question regarding user satisfaction with software revealed that regardless the interface, all users showed a good level of acceptance of the received functionality. Fig. 5(d) depicts the levels of satisfaction in both groups of users. This level of satisfaction can be attributed to the fact that users using the direct-manipulation interface believed to have received the same level of suggestions from the beginning, whereas users in the mixed-initiative interface noted an improvement in suggestions.

In conjunction, answers to the previous questions revealed a better reception of users using the direct-manipulation interface to agent suggestions. In the other group, the presence of the lamp denoting an assistant collaborating in Web page search seems to have increased the expectations of users about the kind of help this

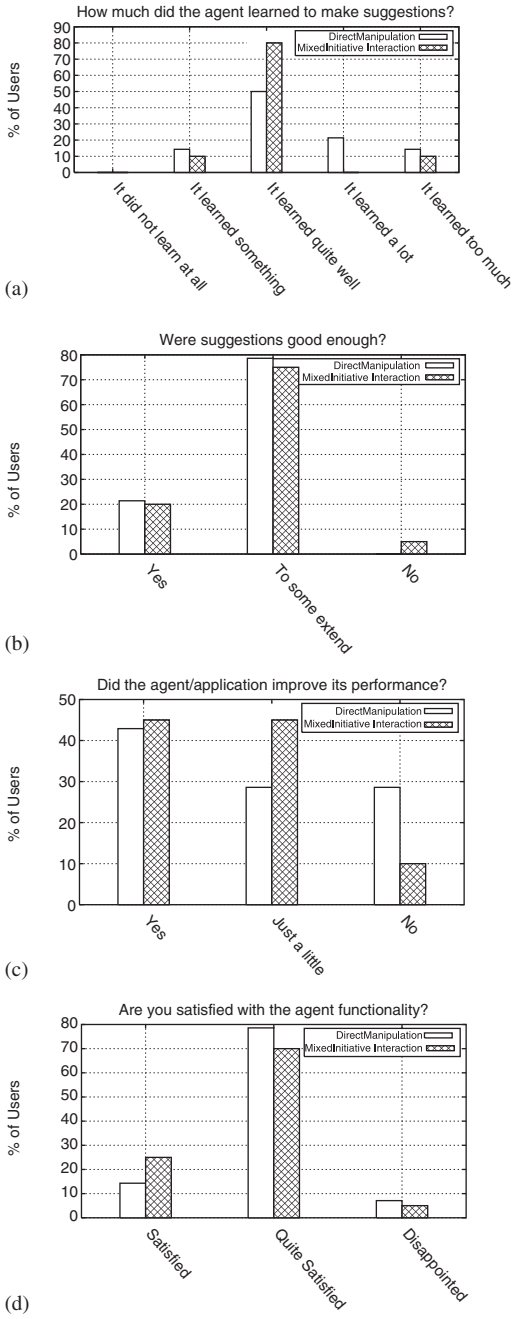


Fig. 5. User perception of agent/application actions.

assistant could have provided. In fact, these users realized the agent improvement which occurs in both groups since *PersonalSearcher* improves with experience. As a result of high expectations, answers from the mixed-initiative interaction group demonstrated a more critical position concerning agent suggestions.

In order to establish the preferences of users concerning both manipulation metaphors, distinct questions were addressed to users in each group. Users who used the direct manipulation interface were asked whether they would have preferred not to receive the suggestions directly but ask for them. Fig. 6(a) summarizes answers to this

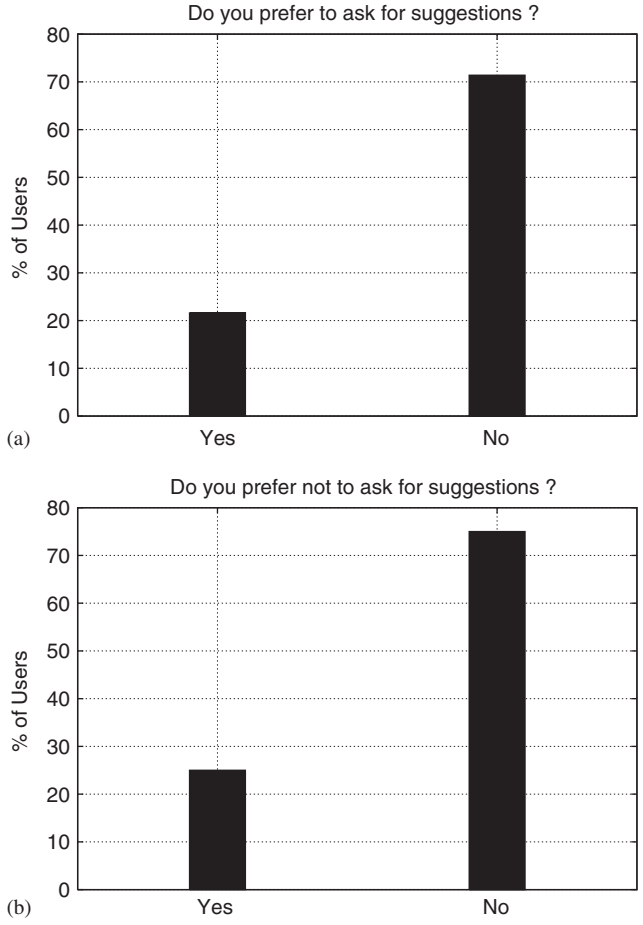


Fig. 6. Preferences about interaction metaphors.

questions. Likewise, users who used the mixed-initiative interface were asked whether they would have preferred a direct access to suggestions without the agent intervention. Fig. 6(b) summarizes answers given by the second group of users.

In both cases, more than a half users found no reason to change the interaction metaphors. However, the number of users preferring a change in the group of users using the mixed-initiative interface was slightly higher in this group than in the group of users using the direct manipulation interface (75% and 71.4%, respectively). This difference can be explained by the level of satisfaction with suggestions shown by the former group, which was lower than the one of the second group.

A further important factor in agent–user interaction and an essential component of agent learning is the relevance feedback users explicitly and/or implicitly provide about agent actions. A user gives explicit feedback by using one or more ordinal or qualitative scales, whereas implicit feedback is estimated by the agent according to observation of a group of interest indicators. Explicit feedback can be as simple as pressing a dislike/like button or more complex like scoring an item in a qualitative scale. It is generally assumed that implicit feedback, although less reliable and more difficult to obtain, does not burden the

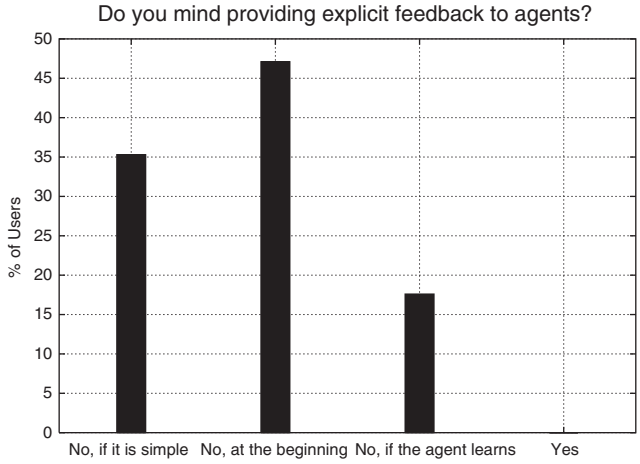


Fig. 7. Opinions of users regarding feedback.

user with an additional cognitive load caused by the necessity of evaluating items.

In the survey we questioned users about the kind of feedback they preferred to give. In this case, results indicate that 47.1% of the users do not mind providing explicit feedback when they start their interaction with an agent. We have to consider that some bias was introduced by the fact that participants of the experiment were students of a course of agent technologies, and therefore they recognize the importance of training an agent in early stages. However, participants decline to provide feedback when they have interacted with the agent for some time. Other users do not mind giving explicit feedback provided that the feedback mechanisms are simple (35.3%) so that they will not have to spend a lot of time and effort in this task. A considerable low number of users (17.6%) answered that they did not complain about giving feedback because they believed it was necessary for the agent to learn and improve its behaviour. None of the users in this study refused completely the possibility of giving feedback. Fig. 7 depicts users reactions toward explicitly provide feedback.

4.7. Discussion

The first four questions of the survey were related to all of the hypotheses formulated in Section 4.1. The analysis of the answers allowed us to confirm that users using the mixed-initiative interface had a more critical appraisal of the suggestions presented by the agent given their higher expectations and level of engagement with the application. This conclusion is supported by the fact that although in both groups the learning mechanism used by the agent is the same, users of the direct manipulation interface perceived a good performance of the agent from the beginning but they did not noticed substantial improvements in the agent suggestions over time. On the other hand, users of the mixed-initiative interface did not perceive such good suggestions at first, but notice a gradual improvement in the agent performance. This explains why

the fourth question, which directly address user satisfaction with the agent functionality, did not evidenced strong differences between both groups of users.

Finally, the fifth question, regarding user preferences about the manipulation metaphor, did not bring enough evidence to confirm a choice between one interface over the other, although the percentage of users that would prefer a change is a bit higher in the mixed-initiative group.

5. Conclusions

In this paper we have discussed the impact the manipulation metaphors and, particularly, the interaction with synthetic agents in agent-based interfaces has on the user perception of software. In the experiment we carried out with real users, two groups of users interacted with an application through interfaces presenting different interaction metaphors: direct manipulation and mixed-initiative interaction. The purpose of this study was to contribute to the better understanding of these issues in order to help designers in building agent-based interfaces. Results revealed that the interaction with an agent generates a greater perception of the computer as a collaborative decision maker and increases the user demand for high quality results.

Even though the representation of agents in the user interface increases the level of engagement of users with the application, such representation can lead to misinterpretations about agent actions and expected results. This problem can be mitigated by the design of appropriate personification metaphors allowing users to assess the capabilities and limitations of agent-based systems.

Acknowledgements

Research reported in this paper has been partially supported by Fundación Antorchas.

References

Armstrong, R., Freitag, D., Joachims, T., Mitchell, T., 1995. WebWatcher: a learning apprentice for the World Wide Web. In: Workshop on Information Gathering for Heterogeneous Distributed Environments. AAAI Spring Symposium Series, pp. 6–12.

Brusilovsky, P., Maybury, M.T., 2002. From adaptive hypermedia to the adaptive Web. Communications of the ACM 45 (5), 30–33.

Brusilovsky, P., Kobsa, A., Vassileva, J., 1998. Adaptive Hypertext and Hypermedia. Academic Publishers, Dordrecht.

Cesta, A., D'Aloisi, D., 1999. Mixed-initiative issues in an agent-based meeting scheduler. User Modeling and User-Adapted Interaction 9 (1–2), 45–78.

Cordero, D., Roldan, P., Schiaffino, S., Amandi, A., 1999. Intelligent agents generating personal newspapers. In: International Conference on Enterprise Information Systems, ICEIS'99. Setubal, Portugal, pp. 195–202.

Cypher, A., 1993. Watch What I Do—Programming by Demonstration. The MIT Press, Cambridge, MA.

Godoy, D., Amandi, A., 2000. PersonalSearcher: an intelligent agent for searching Web pages. In: Advances in Artificial Intelligence, Lecture Notes in Artificial Intelligence, vol. 1952. Springer, Berlin, pp. 43–52.

Godoy, D., Amandi, A., 2005. Modeling user interests by conceptual clustering. Special Issue on the Semantic Web and Web Services, Information Systems Journal, to appear.

Gruen, D., Sidner, C., Boettner, C., Rich., C., 1999. A collaborative assistant for E-Mail. In: Proceedings of Human Factors in Computing Systems, pp. 196–197.

Hongpaisanwiwat, C., Lewis, M., 2003. Attentional effect of animated character. Human–Computer Interaction—INTERACT'03, pp. 423–430.

Koda, T., Maes, P., 1996. Agents with faces: The effects of personification of agents. In: Proceedings of Human–Computer Interaction, London, UK, pp. 239–245.

Lieberman, H., 1995. Letizia: an agent that assists Web browsing. In: Proceedings of the 14th International Joint Conference on Artificial Intelligence, pp. 924–929.

Lieberman, H., 1997. Autonomous interface agents. In: Proceedings of the ACM Conference on Computers and Human Interface, pp. 67–74.

Maes, P., 1994. Agents that reduce work and information overload. Communications of the ACM 37 (7), 31–40.

Maes, P., Kozierok, R., 1993. Learning interface agents. In: Proceedings of the 11th National Conference on Artificial Intelligence, Washington, DC, USA. MIT-Press, AAAI-Press, Cambridge, MA, pp. 459–465.

Mulken, S., Andre, E., Muller, J., 1998. The persona effect: how substantial is it? In: Johnson, H., Nigay, L., Roast, C. (Eds.), People and Computers XIII: HCI'98. Springer, Berlin.

Norman, D., 1994. How might people interact with agents. Communication of the ACM 37 (7), 68–71.

Okonkwo, C., Vassileva, J., 2001. Affective pedagogical agents and user persuasion. In: Proceedings of the Ninth International Conference on Human–Computer Interaction, pp. 397–401.

Pazzani, M., Muramatsu, J., Billsus, D., 1996. Syskill Webert: identifying interesting Web sites. In: Proceedings of the 13th National Conference on Artificial Intelligence and Eighth Innovative Applications of Artificial Intelligence Conference, Portland, USA. AAAI Press/MIT Press, Cambridge, MA, pp. 54–61.

Rich, C., Sidner, C., 1998. COLLAGEN: a collaboration manager for software interface agents. User Modeling and User-Adapted Interaction 8 (3/4).

Schiaffino, S., Amandi, A., 2004. User–interface agent interaction: personalization issues. International Journal of Human–Computer Studies 60 (1), 129–148.

Shneiderman, B., Maes, P., 1997. Direct manipulation vs. interface agents. Interactions 4 (6), 42–61.