Available at
www.ElsevierComputerScience.com
POWERED BY SCIENCE DIRECT®
Int. J. Human-Computer Studies 60 (2004) 129–148
International Journal of Human-Computer Studies
www.elsevier.com/locate/ijhcs

User – interface agent interaction: personalization issues

Silvia Schiaffino[a,b,*], Analía Amandi[a]

[a] *ISISTAN Research Institute, Facultad de Ciencias Exactas, Universidad Nacional del Centro de la Provincia de Buenos Aires, Campus Universitario, Paraje Arroyo Seco, B7001BB0, Tandil, Buenos Aires, Argentina*

[b] *CONICET, Comisión Nacional de Investigaciones Científicas y Técnicas, Argentina*

Received 1 September 2003; accepted 8 September 2003

Abstract

Interface agents are computer programs that provide personalized assistance to users with their computer-based tasks. Most interface agents achieve personalization by learning a user's preferences in a given application domain and assisting him according to them. In this work we adopt a different approach to personalization: how to personalize the interaction between interface agents and users in a mixed-initiative interaction context. We have empirically studied a set of interaction issues that agents have to take into account to achieve this goal and we present our results in this article. Some of these personalization issues are: discovering the type of assistant a user wants, learning when (and if) to interrupt the user, discovering how the user wants to be assisted in different contexts. As a result of our experiments, we have defined the components of a user interaction profile that models a user's interaction and assistance preferences. This profile will enable interface agents to enhance and personalize their interaction with users by discovering how to provide each user assistance of the right sort at the right time.

1. Introduction

Interface agents have become a technology widely used to provide personalized assistance to users with their computer-based tasks. Interface agents are computer programs that have the ability to learn a user's preferences and working habits, and

*Corresponding author. Tel.: 542293440363.
E-mail addresses: sschia@exa.unicen.edu.ar (S. Schiaffino), amandi@exa.unicen.edu.ar (A. Amandi).

1071-5819/$ - see front matter © 2003 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ijhcs.2003.09.003

to provide him proactive and reactive assistance in order to increase the user's productivity.

A commonly used metaphor for understanding interface agent paradigm is comparing them to a human secretary or personal assistant who is collaborating with the user in the same work environment (Maes, 1994). In this relationship, both the human secretary and her boss can initiate an interaction.[1] This also applies to interface agents assisting users in a mixed-initiative context. At any time, one of the participants (user or agent) might have the initiative while the other works to assist him, contributing to the interaction as required. At other times, the roles are reversed; and at other times again the participants might be working independently, interacting with each other only when specifically asked. Mixed-initiative interaction refers to a flexible interaction strategy in which each participant, user or agent, contributes what it is best suited at the most appropriate time (Hearst, 1999).

In this context, Horvitz (1999) has pointed out some problems with the use of interface agents: poor guessing about the goals and needs of users, inadequate consideration of the costs and benefits of each agent action, poor timing of agent actions and inadequate attention to opportunities that allow a user to guide the invocation of agent services and to refine potentially suboptimal results provided by the agent.

In addition, we consider that interface agent developers have paid little attention to two key issues when developing agents that assist users: how to best interact with each user and how to provide them assistance of the right sort at the right time. We have focused our attention on studying the interaction between a user and an agent, as well as on how a user wants to be assisted. We believe that, in the same way as each person works and interacts in a certain way with his/her personal assistant, each user interacts in a personal way with an interface agent.

To prove our hypothesis, we carried out an experiment with real users interacting with different interface agents, and we discovered several user–agent interaction issues that require personalization. The goals of our experiment were the following:

- Firstly, we studied what kind of software assistant different users want. Given that different people require different types of human secretaries, it is likely that they will want different types of interface agents. For example, some users might prefer a collaborator but others might want a submissive agent that only executes their orders. Thus, one of the goals of our study was discovering whether these differences in users' preferences appear when users interact and work with interface agents or not.
- Secondly, we studied users' preferences and assistance requirements concerning three assistance actions: warnings, suggestions and actions on the user's behalf. Our purpose was studying the assistance actions users expect in different situations or contexts. We analysed if users require different assistance actions to deal with a given situation in different contexts, and which these situation–action

[1] For simplicity, we use "she" for the assistant and "he" for the boss, but we do not mean to be sexist.

pairs are. Particularly, we studied the differences among the preferences and requirements of various users.

- Thirdly, we analysed users' reactions towards interruptions, i.e. whether users object to being interrupted by their agents or not. We studied how different users react to interruptions in different contexts (relevant and irrelevant situations) and how users behave when agents interrupt their work.
- Fourthly, we considered users' tolerance of agents' errors, since some users are more tolerant than others. We studied users' tolerance to errors involving warnings, suggestions and actions. We analysed variations in users' tolerance depending on the type of error, the assistance actions involved and the stage in user–agent interaction. We focused our attention on the differences that exist among different users' reactions to agents' errors.
- Fifthly, we investigated how users feel about providing explicit feedback to train their agents. Not all users are willing to give explicit feedback, and their willingness may vary as they interact with the agent. We studied users' reactions towards providing feedback in different stages of the user-agent interaction and to different types of feedback (simple vs. complicated).
- Finally, we analysed users' opinions regarding agents' ability to perform actions on behalf of users and the possibility of giving agents more autonomy. We wanted to discover if human beings are ready for a mixed-initiative interface agent paradigm or if they still want to be in complete control.

As a result of our experiment we found that different users have different preferences and requirements regarding the interaction issues mentioned above. This is an indication that personalization is required if interface agents want to assist users as they expect. Thus, we propose a definition for a user interaction profile that models a user's interaction preferences, requirements and habits. The user interaction profile can be used by an interface agent to personalize its interaction with each user and to adapt its behaviour to each user's interaction and assistance requirements.

The rest of the paper is organized as follows. In Section 2 we discuss the importance of personalization in user–agent interaction. Section 3 describes in detail each of the personalization issues we have experimentally studied. Section 4 summarizes the lessons we have learned. Section 5 presents our definition of a user interaction profile. Section 6 describes some related works. Finally, Section 7 presents our conclusions and future work.

2. The importance of personalization

Personalization involves a process of gathering user-information during interaction with the user, which is then used to provide appropriate assistance or services, tailor-made to the users needs (Bonett, 2001). Personalization is motivated by the recognition that a user has needs, and meeting them successfully is likely to lead to a satisfying relationship with him (Riecken, 2000).

We address personalization within a mixed-initiative interaction context in which an interface agent assists a user. The most common approach to personalization is learning a user's preferences or interests regarding a particular domain or application and assisting the user according to them. In this work we consider personalization from the point of view of the interaction between the user and the agent. Our personalization approach aims at improving the interaction between the user and the agent so that the user can count on a mindful assistant and a collaborator that knows not only his interests but also when and how he wants to be assisted.

Each person works and interacts in a certain way with his/her personal assistant. Therefore, it is natural to think that each user will interact in a personal way with his interface agent. Thus, the kind of assistant a user wants, the actions a user expects from his agent, the agent's errors he tolerates and the type of assistance a user requires in a given context will vary from one user to another. For example, a given user would probably not mind being interrupted by an agent with notifications or suggestions, if he often accepts this from a human assistant. Other user would probably dismiss an assistant of such a kind, which means that he would never tolerate such a behaviour in an interface agent.

The consequences of not fulfilling users' expectations can be disastrous for an interface agent. When an agent makes mistakes, which are especially likely in early, learning stages, users may place less trust in the agent. In an extreme case, users may adopt a strategy of completely ignoring or disabling the agent (Tiernan et al., 2001). For example, a disappointed user may limit the agent's capabilities to a minimum, allowing it to act only upon his request, or he can diminish his interaction with the agent ignoring its actions although some of them might be useful. An annoyed user because of its agent's behaviour (e.g. one that always interrupts the user with irrelevant information) may prefer working without the agent and, thus, abandon it.

As an example of disastrous consequences for software assistants we can mention the decision of Microsoft product managers to "kill" the Microsoft Office Assistant, named Clippit, because it has been the source of wide scorn among developers. Microsoft listened to users' complaints and made Clippit easier to hide in Office 2000. But the complaints kept coming in and now Clippit does not even show up by default in Office XP.[2]

In summary, the importance of personalization lies on its potential to allow interface agents to maintain a good relationship with users by considering each user's interaction preferences and assistance requirements. The more user–agent interaction aspects interface agents can personalize, the more successful they will be in their interaction with users. The following section describes the set of personalization issues we have studied, which should be taken into account when developing an interface agent.

[2] 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.

3. Personalization issues: empirical study

The goal of our experiment was to study different aspects of the interaction between a user and an interface agent that might require personalization. Besides, we studied various critical aspects of user–agent interaction such as how much control to delegate to an agent and users' reactions towards actions performed on the user's behalf, interruptions and explicit feedback. As we have said before, we believe that personalization is needed because different users expect different things from their agents and users have different interaction and working preferences.

This section is organized as follows. Section 3.1 describes our experiment participants' characteristics and background. Section 3.2 describes our experiment procedure. Finally, Section 3.3 presents the results we have obtained.

3.1. Experiment participants

Forty-two users ranging in age from 21 to 50 yr old, 39 male and 13 female, participated in the experiment. Some participants had developed and actively used interface agents in several application domains. Other participants had occasionally used some interface agents in various domains, and others had only interacted with those assistants that we can find in MS Word or MS Excel. We categorized the first group of participants as expert users (12%), the second group as intermediate users (64%), and the last group of participants as inexpert users (24%) regarding their experience at interacting with interface agents. Fig. 1 shows the expertise levels of the participants.

As we have said before, expert participants had developed interface agents in various application domains, which makes them highly experienced regarding the experiment context. Intermediate users and most of the expert users had interacted with a subset of these interface agents, namely: an agent assisting users with calendar management (Schiaffino and Amandi, 2002), an agent assisting users in web searching (Godoy and Amandi, 2000), an agent that generates personalized newspapers (Cordero et al., 1999), and an agent that assists database users (Schiaffino and Amandi, 2003). These agents can provide different types of assistance, such as warning or alerting the user about relevant or interesting situations, making suggestions and automating some activities. The way in which the assistance is provided varies from one agent to another. For example, some agents

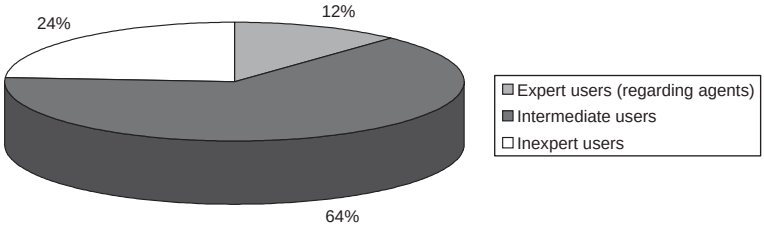

Fig. 1. Levels of expertise of the participants.

alert the user about a suggestion they can make to the user and then, upon request, they show him the suggestion. Other agents show the suggestions directly, without any previous notification. This variety of assistance actions and user interfaces allowed users to have a big picture about how interface agents can assist them and how they can interact and work with them.

3.2. Experiment procedure

All participants were given a survey with a set of questions regarding user–agent interaction and user assistance. The users were asked to answer the questionnaire providing as much information as possible. Those users who had interacted with interface agents developed within our research group answered the questions in terms of their personal experience with these agents. Those users who had only had interaction with MS Office assistants answered the questions according not only to their experience with these assistants, but also to their expectations about an interface agent's behaviour in those cases where they had no practical experience. We consider that both kinds of answers are equally important and valuable for our study since we want to know how the different users feel about interacting with software assistants.

3.3. Experiment results

The following sections describe in detail the results we have obtained from our experiment. Each section addresses a particular issue regarding personalization in user–agent interaction, or a combination of different issues.

3.3.1. Do users prefer submissive or authoritative agents?

Each person having a personal assistant works and interacts with her in a certain way. For example, some people delegate most of their tasks to their assistants and trust them completely. Others probably do not like to delegate their tasks to their secretaries and they only let them do what they ask them to do. Therefore, it is natural to think that the same occurs with interface agents: different users are likely to prefer different types of software assistants. The type of relationship between a user and an interface agent can lead to different user–agent interaction styles, and probably can originate different user stereotypes, such as authoritative or dominant user (one that wants a submissive agent), interactive user (he wants an interactive and collaborative agent), delegator (he delegates everything to his agent), among others. An interface agent must discover which type of assistant the user wants as a first step towards personalization.

Our experiment revealed that 12% of the users prefer an agent who can collaborate and actively interact with them. The 40% of the users expect their agents to live "in the background", becoming active only when the user requests the agent's assistance. The 48% of the users want their agents to take part only in certain situations, probably predefined by users, in which they would interact and collaborate with their agents. Fig. 2 shows the preferences of the different users.

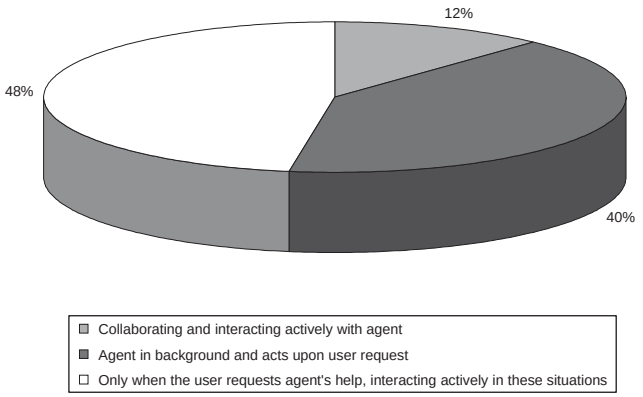

Fig. 2. User–agent interaction styles.

As regards the different types of users, we found that 20% of the expert users want a submissive agent, 40% of this type of users want a collaborator, and 40% prefer an agent that can act autonomously only in certain occasions. Concerning intermediate users, we discovered that 30% of the intermediate users want a submissive agent, 11% want a collaborative agent, and 59% prefer an agent that can act autonomously only in certain occasions. Finally, the 70% of inexpert users want their agents to live in the background, none of them want a collaborative agent, and the 30% of these users want an agent that takes part only in certain situations. These last figures reflect the fear of inexpert users towards having an autonomous assistant. Most of these users want to be in control, calling their agents just when they need them. Fig. 3 shows these results.

The type of assistant a user wants is closely tied to the types of actions the user allows the agent to execute. For example, a user who delegates most of his tasks to his agent will let it make warnings, make suggestions and mainly perform tasks on his behalf. On the other hand, a user that only allows his agent to execute his commands will not let it make warnings or suggestions and he will never tolerate autonomous actions. Hypothetically, an intermediate user will allow his agent to make warnings and suggestions, but he will probably not enable the agent to execute autonomous actions unless he asks it to do that.

As shown in Fig. 4, most users (49%) would allow their agents to make warnings, alerts and suggestions, but they would not allow them to perform autonomous tasks. About 27% of the users would let their agents make only warnings in an autonomous fashion, and the rest of the assistance actions only upon request. Only 22% of the users would allow their agents to execute every type of assistance actions, and 2% would only let the agents act when they are told.

3.3.2. Allowing actions on the user's behalf

One of the most controversial aspects of agents is their ability to perform tasks on users' behalf in an autonomous fashion. Moreover, the possibility that the user could

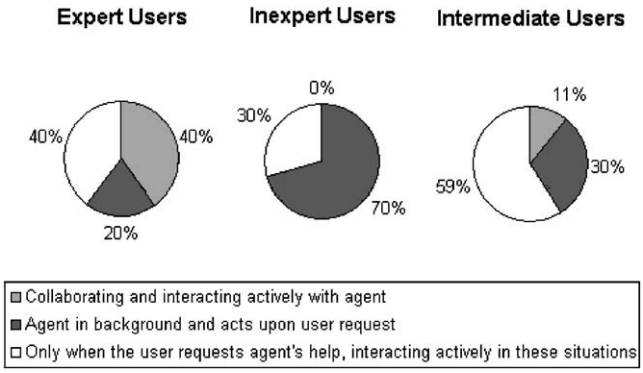

Fig. 3. Preferences of the different types of users.

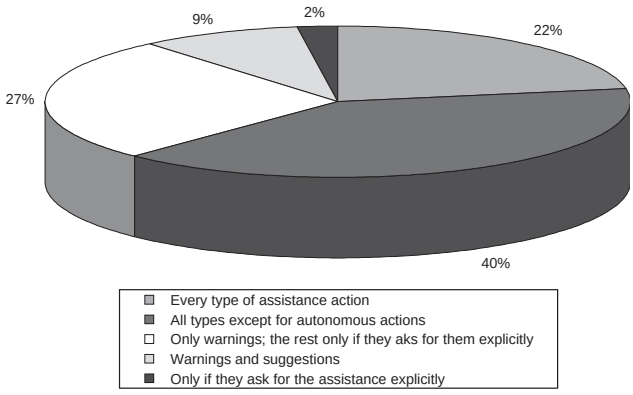

Fig. 4. Assistance actions allowed by users.

not completely control the agent is considered as negative from a human–computer interaction (HCI) point of view (Whittaker and Sidner, 1996). So far, some interface agents have dealt with this problem by enabling users to control agents' behaviour by setting some parameters and thresholds (Kozierok and Maes, 1993; Maes, 1994; Fleming and Cohen, 1999). In other cases, these thresholds are set by the developers.

Our experiment has confirmed users' fear towards having a completely automated agent. The results regarding this issue are shown in Fig. 5. The 4% of the users would never allow their agents to execute tasks autonomously without any kind of control. The rest of the users would enable their agents to act autonomously provided that: users know exactly what the agent is going to do (45%); only if they ask their agents to perform the tasks (36%); only in situations that do not compromise them (15%). Most users agreed in that they "do not want to lose control of their computers" and that autonomous actions are "dangerous". On the contrary, some users find useful harmful actions such as the automatically filling of certain fields in calendar management for example.

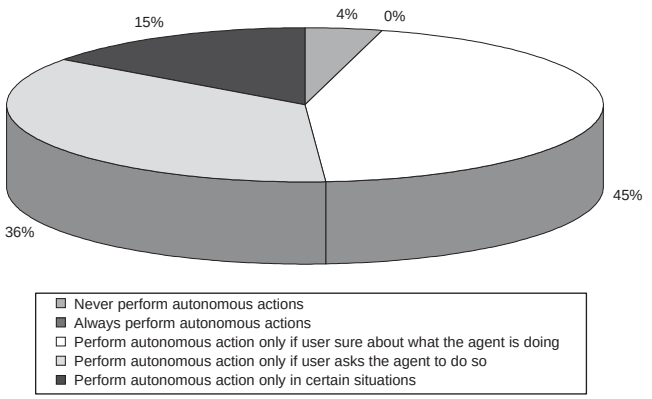

Fig. 5. Users' acceptance of agents' actions on their behalf.

As regards the different types of users, we found that none of the expert users would let their agents execute autonomous actions without any kind of control, 40% of the expert users would let agents perform autonomous tasks only if they are sure about their actions, 20% of these users would let agents act only in certain occasions, and 40% want their agents to act only upon users' request. Concerning intermediate users, 48% of them would let agents act on their behalf only when they are sure about what agents are doing, 41% want agents to act only when they are told, and 11% of these users want their agents to take part only in certain situations. As regards inexpert users, 20% of them would never allow agents to perform actions on their behalf without any kind of control; 30% would enable agent to act autonomously when users know exactly what they will do; 30% of the inexpert users would enable agents to act when they request their assistance and the remainder 20% would allow agents to perform autonomous actions only in certain predefined situations. Fig. 6 shows these results.

3.3.3. Does the user want suggestions or just warnings?

If an interface agent wants to be competent and wants to be accepted by the user it is assisting, it has to discover what exactly the user needs in a given context or situation. For example, consider an agent assisting a user with his calendar management, and suppose that the user is scheduling a meeting with several participants for the following Saturday in a free time slot. The agent detects that one of the participants will surely disagree with the meeting date because he never attends meetings on Saturdays. The agent can merely warn the user about this problem, it can suggest the user another meeting date considering all participants' preferences and priorities, or it can do just nothing. Our hypothesis is that a certain user probably prefers a warning about the inconvenient, but another user surely wants a suggestion of another meeting date in that situation. Thus, the agent will have to discover what each user needs in each situation or context.

Our experiments revealed that the 60% of the users could identify some situations in which they clearly prefer a particular type of assistance action rather than the

others, but 40% of the users could not identify these kind of situations. In the case of users belonging to the first group, it is clear that the agent has to learn when to provide each type of assistance because each user has different preferences. Most of the users who could not identify situations in which they would prefer a particular type of assistance were intermediate and inexpert users. Probably, their inexperience did not let them identify particular situations, but this does not mean that such situations do not exist. The agent will have to discover them. Fig. 7 shows users' preferences regarding warnings and suggestions.

Concerning the choice between warnings and suggestions, we also studied how users prefer to be notified when the agent has something to suggest. The 67% of the users wants to be notified first about the potential suggestions and then obtain the suggestions. The rest of the users (33%) wants to get the suggestions directly, without any previous notification. We studied this issue by providing two different user interfaces in one of our agents, and asking users which of them they found more convenient or practical. Fig. 8 shows users' preferences regarding this issue.

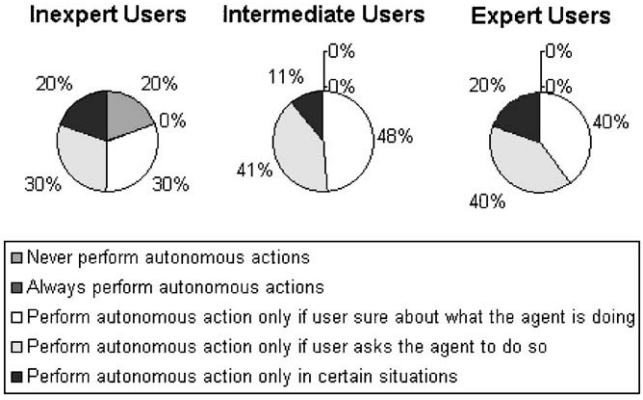

Fig. 6. Preferences of the different types of users.

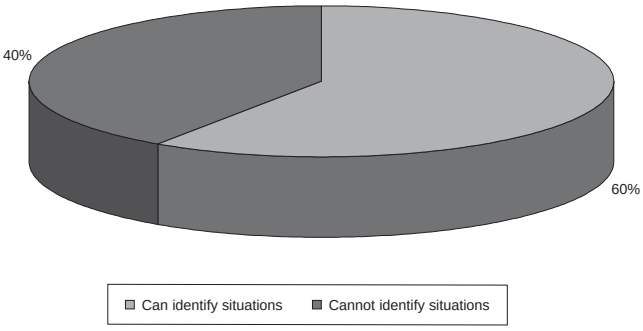

Fig. 7. Ability to distinguish different assistance requirements.

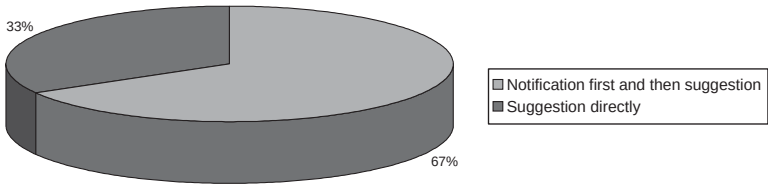

Fig. 8. Users' preferences regarding suggestions.

In summary, we found that different users have different preferences regarding the three assistance actions we are considering (i.e. warnings, suggestions and actions on the user's behalf) and that event a single user wants different actions to deal with a given problem depending on the context. Thus, personalization is required in order to assist each user properly.

3.3.4. Do users tolerate agents' errors?

Tolerance of agents' errors is a key point in user–agent interaction. Some experiments have demonstrated that users tolerate their agent's errors as far as they achieve an acceptable performance (Ruvini and Gabriel, 2002). In our experiment, we focused our attention on errors regarding the different types of assistance actions, i.e. warnings, suggestions and actions on the user's behalf.

We discovered that only 7% of the users do not get angry if their agents provide them the wrong type of assistance and that they would fix the situation in the case of an error. The rest of the users would be disappointed if their agents provided them the wrong type of assistance. The 14% of the users would be highly disappointed, 50% said that their disappointment depends on the magnitude of the mistake, 27% would try to train the agent by means of explicit feedback so that these mistakes could decrease in the future, and the 2% would fix the situation by themselves without interacting with the agent (i.e. without providing explicit feedback). Fig. 9 shows these results.

Many users get really angry if the type of assistance involved in an error is an action or if the mistake is severe (85%), but they can tolerate a suggestion instead of a warning. The 15% of users are equally tolerant (or intolerant) of errors involving the different types of assistance actions. Fig. 10 shows these proportions.

As shown in Fig. 11, 90% of the users are more tolerant of agent's errors when they start the interaction with their agents, but they do not tolerate mistakes once they have interacted with their agents for a long time. The 10% of users do not tolerate mistakes at any stage of the user–agent relationship.

3.3.5. Do users object to providing explicit feedback?

One of the main sources of learning in interface agent technology is user feedback. This feedback may be explicit, when users explicitly evaluate an agent's actions through a user interface provided for that purpose, or implicit, when the agent observes a user's actions after assisting him to detect some implicit evaluation of its assistance. The explicit feedback can be simple or complex. It is simple when, for

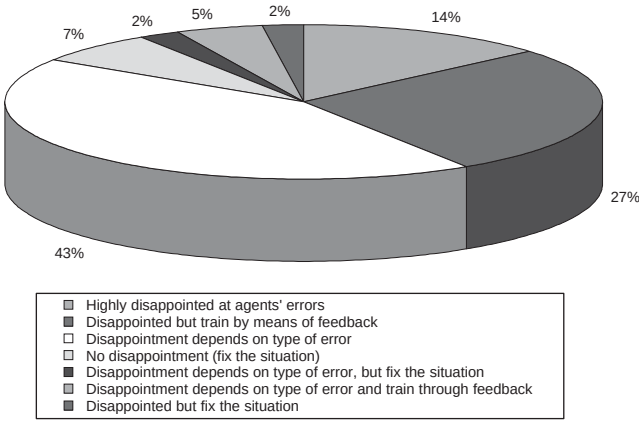

Fig. 9. Users' tolerance to agents' errors.

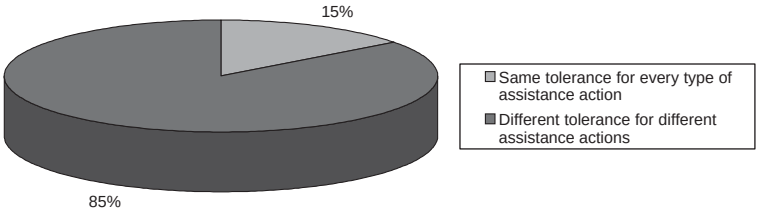

Fig. 10. Users' tolerance to different types of errors.

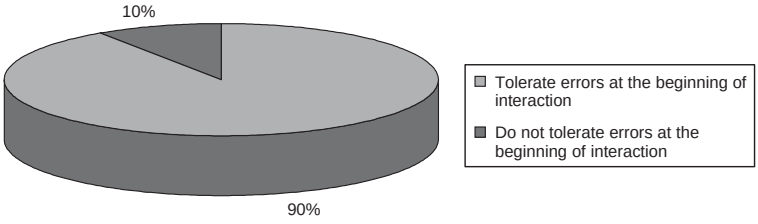

Fig. 11. Users' tolerance in different interaction stages.

example, the user is required to evaluate the agent's assistance according to a quantitative or a qualitative scale (e.g. 0–10, relevant or irrelevant) or to just press a dislike/like button. However, it becomes more complicated when the user is required to provide big amounts of information in various steps.

Mostly, an interface agent has to learn from implicit feedback since the explicit feedback is not always available. The reason is that not all users are willing to provide explicit feedback, mainly if this demands them a lot of time and effort.

We discovered that 35% of the users do not mind providing explicit feedback when they start their interaction with an agent, because they consider that they have to train it in early stages. However, they find annoying to provide feedback when they have interacted with the agent for some time. The 19% of the users do not mind giving explicit feedback provided that the feedback mechanisms are simple and that they do not have to spend a lot of time and effort evaluating the agent's behaviour. The 24% of the users answered that they do not complain about giving feedback if the two previous conditions are met. In the extreme cases, only 5% of the users finds completely bothersome to provide explicit feedback, and the 17% do not complain about giving feedback at any stage of the interaction because they believe it is necessary for the agent to learn about them. Fig. 12 shows users' reactions towards providing explicit feedback.

3.3.6. Do users accept agents' interruptions?

One of the key decisions agents must take not to hinder the user's work is discovering whether the user minds being interrupted or not, and in which contexts this occurs. Some studies have demonstrated that notifications can be disruptive, both frustrating users and decreasing the efficiency with which they perform ongoing tasks (Cutrell et al., 2001). Some of the consequences for an agent showing a disruptive behaviour were discussed in Section 2.

The experiment we carried out indicates that 10% of the participants do not mind being interrupted by their agents at all. On the contrary, 14% of them do not tolerate being interrupted under no circumstances and 76% of the users do not mind being interrupted if the underlying situation is important and relevant to them. The big problem here is discovering which situations are relevant or urgent for each user and which are irrelevant. Some users told us that they would appreciate having some mechanisms to tell the agent which a relevant situation is and which not. In Fig. 13 we can visualize users' preferences concerning interruptions.

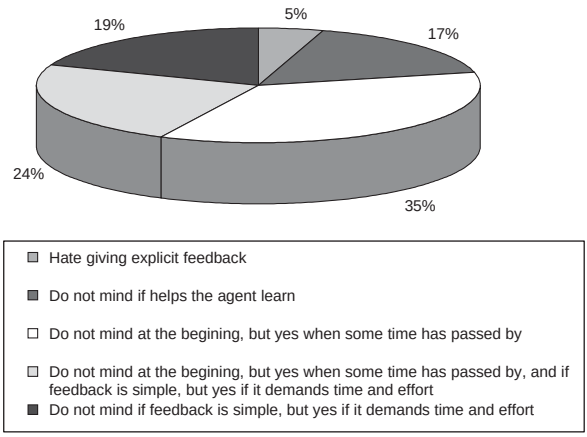

Fig. 12. Users' reactions towards providing explicit feedback.

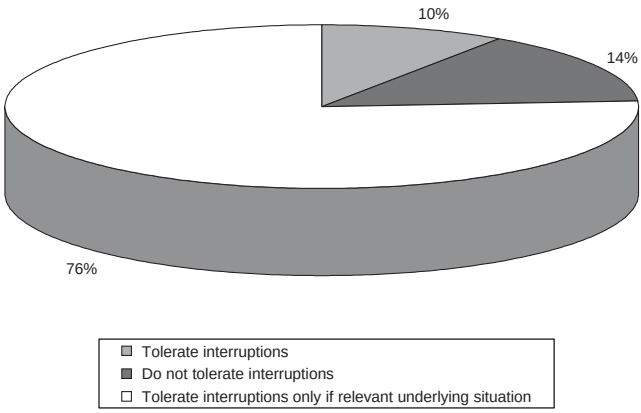

Fig. 13. Users' reaction towards interruptions.

4. Lessons learned

The experiment we carried out revealed us some interesting aspects of user–agent interaction and confirmed some assumptions about users' reactions towards agent technology. As yet, interface agents lack the ability of personalizing their interaction with users according to the user's interaction preferences and requirements.

We discovered that there are several issues interface agent developers have to address if they want to personalize and improve the interaction between interface agents and users:

- Discovering the type of assistant each user wants.
- Considering the particular assistance requirements users have in different contexts.
- Analysing users' tolerance to agents' errors.
- Discovering when (context awareness), and when not, to interrupt the user.
- Providing the means to provide simple (but useful) explicit user feedback.
- Providing the means to capture as much implicit feedback as possible.
- Discovering how much control the user wants to delegate to the agent.
- Providing the means to control and inspect agent behaviour.

To personalize its interaction with the user, an interface agent has to discover what type of assistant a user wants. Some users want just a secretary that executes their orders without any autonomous intervention. Other users enable their assistants to make some suggestions and warnings, whilst others wants an assistant capable of actively collaborating with them and of performing tasks on their behalf. The agent has to observe and analyse the user's reactions towards the different assistance actions and discover then which are those he mostly accepts. The agent has to pay also especial attention to those actions the user explicitly requests his agent to execute. This information will enable the agent to discover which the user's interaction style is, i.e. authoritative, collaborator or delegator.

Second, the agent has to distinguish which type of assistance the user wants in the different situations that may arise. Different users require different agent actions in the same context, and even a user may require different agent actions in different instances of a given problem situation. The agent must discover when the user wants it to propose a solution to a problem or to deal with a given situation, when the user wants the agent to solve the problem on the user's behalf, when he wants it to simply warn him about the situation and when he wants the agent to do just nothing.

Third, once the agent has learned what type of assistance the user needs, it has to learn how to provide it, i.e. interrupting the user's work or not. Most users agreed in that they tolerate interruptions if the underlying problem or situation of interest is relevant to them. Thus, the agent has to analyse the relevance the situation to be notified has for the user, probably depending on the task the user is carrying out and on the different contexts.

Fourth, the agent has to discover how tolerant the user is of its assistance mistakes. Two users can react differently to the same agent mistake. For example, one of our users hates being interrupted when he is working, even by a "new email" message. He told us that his reaction towards useless interruptions would be uninstalling the agent. On the contrary, other users are more tolerant and patient. In the former case, the agent has to be very confident when providing assistance, but in the latter cases it can attempt to make some guesses regarding the user's preferences since these users are more tolerant.

As regards agents' autonomy, HCI people have criticized agent-based methodologies that seem to produce systems not easily accepted by the user: one of the main reasons is the autonomy of the agents that can cause a loss of control by the user (D'Aloisi et al., 1997; Scerri et al., 2002). We agree with these authors in that one of the reasons for this complain is the limited research dedicated to examining the role of interface agents as a mixed-initiative interaction system, opposite to a completely automated systems. As these authors stated, an agent should be endowed with the capability of acting and autonomously proposing solutions according to the current problem, but the user must have the possibility of controlling and inspecting the agent's decisions. Our experiment confirmed that, in general, users are not ready yet for this kind of interaction.

Concerning agent design, a set of users told us through the survey that they would like to configure their agents' behaviours. For example, some users would like to be able to set the type of actions the agent can execute, or set some threshold values to control agents' behaviour, or to tell the agent what is relevant or irrelevant. Thus, the interface agent developer has to provide the facilities to allow users to configure certain parameters, which can in turn be used by the agent learning module to personalize its interaction with the user.

Another important aspect that has to be taken into account when designing an interface agent is how users will provide feedback. It is well known that most users do not like to provide explicit feedback. However, they would probably provide some if this does not require a lot of time and effort. Thus, the designer must carefully chose the appropriate interaction metaphor and the feedback interfaces in order to capture as much explicit feedback as possible. Many users would like the

request for feedback to diminish as they interact with the agent. However, this situation can originate the following problem: in case of changing interests, the agent might not be able to perceive them if it cannot obtain user feedback. The developer has to find a balance between these two conflicting issues.

5. Implications in agent development: How to achieve personalization

In order to provide personalized assistance to computer users, interface agents need some knowledge about them. This knowledge is generally contained in the user profile. A profile is a description of someone containing all the most important or interesting facts about him/her. In our approach, a user profile should contain all the information concerning user–agent interaction that an interface agent needs to provide personalized assistance to a user. The user profile should contain information regarding the personalization issues discussed in Section 3.

In general, a user profile contains both application dependent and application independent information about a user, and its content varies from one application domain to another. Application-independent information includes mainly personal information about the user, such as name, age, job, and hobbies. Application-dependent information includes a user's interests, preferences, goals, working habits, behavioural patterns, knowledge, needs, priorities and commitments regarding a particular domain. For example, in an on-line newspaper domain, the user profile contains the types of news (topics) the user likes to read, the types of news (topics) the user does not like to read, the newspapers he usually reads, and the user's reading habits and patterns. In a calendar management domain the user profile contains information about the dates and times when the user usually schedules each type of activity, the priorities each activity feature has for the user, the relevance of each user contact and the user's scheduling and rescheduling habits.

However, the information considered thus far is not enough to personalize the interaction with a user. As we have seen, there are many other aspects interface agents have to take into account to personalize their interaction with users. Thus, we will add into the user profile information about the user's interaction habits and styles, about his assistance needs in different contexts, about his reactions towards different assistance actions such as warnings, suggestions, reminders and interruptions and about his preferences towards delegating tasks to the agent.

The profile items regarding user–agent interaction consist of:

- The contexts in which the user requires a suggestion to deal with a problem or handle a situation of interest.
- The situations when the user needs only a warning about a problem or a situation.
- The situations or problems when the user wants the agent to perform a task on his behalf.
- The circumstances under which the user accepts an interruption from the agent.
- The situations in which the user needs a notification and does not want an interruption.

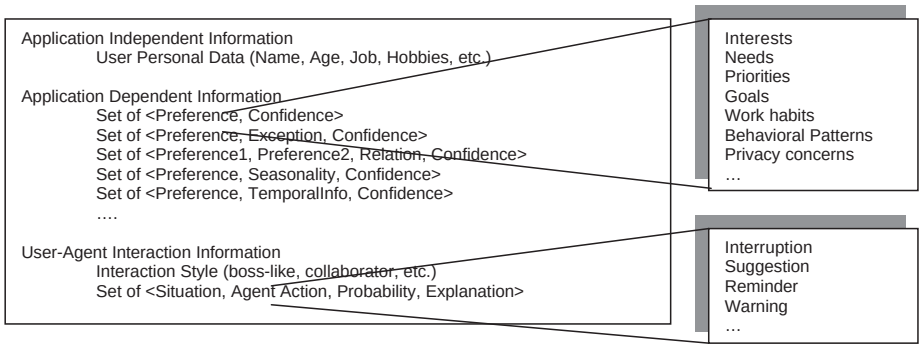

Fig. 14. User interaction profile.

- The situations or problems the user wants to handle by himself without requiring the agent's help.
- The minimum levels of confidence required for different assistance actions, specified by the user or learned by the agent.
- The user interaction style, i.e. collaborator, delegator, authoritative.

The profile items mentioned above are represented as a set of situations together with the required agent action and a parameter indicating the probability that the user wants the agent to execute that action to assist him in that particular situation. Each context or situation is described by a set of feature-value pairs. For example, in the case of calendar management, an opportunity to assist the user is when the user is scheduling an event. The situation will be described by the attributes characterizing the event. The agent action associated with a given situation may be a warning, a suggestion, an action on the user's behalf or an interruption. The probability value associated with a certain situation–action pair may include different factors in its calculus, such as the urgency and the relevance the situation at hand has for the user, the knowledge the agent has to solve the problem and how willing the agent is to handle the problem by himself. This probability value can be considered also as a certainty value indicating how sure the agent is about the user needing or wanting a certain assistance action to deal with a given situation.

The user interaction profile can be used by interface agents to personalize their interaction with users and to provide them the type of assistance they expect. Fig. 14 shows the different components of our proposed user profile.

6. Related work

In this paper we have presented and discussed the results of a study of different user – interface agent interaction issues that require personalization. Some other studies have been carried out to test the impact of different aspects of agent technology in users. For example, some related experiments include those that have

been carried out to test the impact of animated and anthropomorphic interface agents in users (Schaumburg, 2001; Dehn and van Mulken, 2000). The experiments reported in Schaumburg (2001) show how people react to MS Office assistant, which is one of the most popular applications of an anthropomorphic agent. The work in Dehn and van Mulken (2000) presents a review of some empirical researches regarding animated interface agents. We agree with the authors of these works in that some more studies are needed to discover whether there is an advantage of an animated agent over a non-animated one.

Somewhat more related to our work, the two interaction issues that have been studied are interruptions, users' tolerance to agents' errors and agent autonomy. As regards interruptions, although little research has been performed on this topic within agent technology, there have been numerous studies exploring interruptions more generally. McFarlane (1999) examined four methods for deciding when to interrupt someone during multitasked computing. He explored several interruption policies, including immediate (requiring an immediate user response), negotiated (user chooses when to attend), mediated (an intelligent agent might determine when best to interrupt), and scheduled (interruptions come at prearranged time intervals). He concluded that giving people the control to negotiate for the onset of interruptions resulted in good performance.

Some studies regarding notification systems, those that attempt to deliver current, important information to the computer screen in an efficient and effective manner without causing unwanted distraction to ongoing tasks such as instant messaging systems, email alerts and system updates, are presented in Special Issue On Notification Systems, IJHCS (2003). As regards this topic, people at Microsoft Research have deeply studied the effects of instant messaging (IM) in users, mainly on ongoing computing tasks (Czerwinski et al., 2000a,b; Cutrell et al., 2001). Their studies revealed that the disruptiveness of IM to productivity is reduced if the incoming message is highly relevant to the current task. Besides, they demonstrated that the delays associated with a IM disruption depends on the point in a computing task that a user is presented with it. They found that a good time for notifications is early in the task, before the user has become deeply engaged in the task goal, and that notifications arriving during the evaluation, planning and executions phases were harmful. IMs that are relevant to ongoing tasks are less disruptive than those that are irrelevant. However, as yet there is no study that analyses the relationship between interruptions and the relevance the situation underlying the interruption has for the user given different contexts.

Regarding agents performing tasks on users' behalf, the work presented in (Milewski and Lewis, 1997) describes some issues regarding task delegation. The authors consider that the interaction between agents and users is most often based on delegation, and although often useful it is often unnatural and cognitive costly for users. These results coincide with those we have found, since users are not willing to delegate their tasks to agents unless they know exactly what the agent will do.

Finally, the experiments reported in (Ruvini and Gabriel, 2002) evaluate Smartlook (an email classifier assistant) users' satisfaction by comparing its actual prediction accuracy with the users' estimation of this accuracy. The authors found

that as far as an assistant achieves reasonable performances, users tolerate errors from it. Our study coincides in the average with this results, but it has revealed that some users are extremely intolerant.

7. Conclusions

In this paper we have discussed a set of issues that interface agent developers have to take into account in order to personalize the interaction between an interface agent and a user in a mixed-initiative interaction context. We have presented the results of an experiment we carried out to study these personalization issues and we have exposed the lessons we have learned.

Our study contributes to the understanding of user – interface agent interaction. The results of this study have implications on how to design interface agents in order to personalize their interaction with users. There are eight major issues interface agents have to achieve:

- Discovering the type of assistant each user wants.
- Learning the particular assistance requirements users have in different contexts.
- Analysing users' tolerance to agents' errors.
- Discovering when (context awareness), and when not, to interrupt the user.
- Discovering how much control the user wants to delegate to the agent.
- Providing the means to provide simple (but useful) explicit user feedback.
- Providing the means to capture as much implicit feedback.
- Providing the means to control and inspect agent behaviour.

To deal with the problems mentioned above we have proposed a new definition of a user profile that contains the information interface agents need to personalize their interaction with users. We are also working on some user profiling algorithms that can obtain users' interaction preferences and requirements from the observation of user–agent interactions. The outcomes of these algorithms are the components of the user interaction profile.

References

Bonett, M., 2001. Personalization of web services: opportunities and challenges. In: Ariadne Issue 28, http://www.ariadne.ac.uk/issue28/personalization/intro.html.

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

Cutrell, E., Czerwinski, M., Horvitz, E., 2001. Notification, disruption and memory: effects of messaging interruptions on memory and performance. In: Hirose, M. (Ed.), Human-Computer Interaction INTERACT '01, Tokyo, Japan, July 9–13, IOS Press (for IFIP), pp. 263–269.

Czerwinski, M., Cutrell, E., Horvitz, E., 2000a. Instant messaging and interruption: influence of task type on performance. In: Paris, C., Ozkan, N., Howard, S., Lu, S. (Eds.), OZCHI 2000 Conference Proceedings, Sydney, Australia, December 4–8, pp. 356–361.

Czerwinski, M., Cutrell, E., Horvitz, E., 2000b. Instant messaging: effects of relevance and time. In: Turner, S., Turner, P. (Eds.), People and Computers XIV: Proceedings of HCI 2000, Vol. 2, British Computer Society, pp. 71–76.

D'Aloisi, D., Cesta, A., Brancaleoni, R., 1997. Mixed-initiative aspects in an agent-based system. In: AAAI Technical Report SS-97-04 AAAI.

Dehn, D., van Mulken, S., 2000. The impact of animated interface agents: a review of empirical research. In: International Journal Human–Computer Studies 52, 1–22.

Fleming, M., Cohen, R., 1999. User modeling in the design of interactive interface agents. In: Proceedings of the Seventh International Conference on User Modeling, Banff, Canada, June 20–24, pp. 67–76.

Godoy, D., Amandi, A., 2000. Personal searcher: an intelligent agent for searching web pages. In: Monard, M.C., Sichman, J.S. (Eds.), Advances in Artificial Intelligence, International Joint Conference, 7th Ibero-American Conference on AI, 15th Brazilian Symposium on AI, IBERAMIA-SBIA 2000, Atibaia, SP, Brazil, November 19–22, 2000, Proceedings. Lecture Notes in Computer Science, Vol. 1952, Springer, Berlin, pp. 43–52.

Hearst, M.A., 1999. Trends and controversies: mixed-initiative interaction. IEEE Intelligent Systems 14 (5), 14–23.

Horvitz, E., 1999. Principles of mixed-initiative user interfaces. Proceedings of CHI '99, ACM SIGCHI Conference on Human Factors in Computing Systems, Pittsburgh, PA, May, 1999.

Kozierok, R., Maes, P.A., 1993. A learning interface agent for scheduling meetings. In: Proceedings of the 1993 International Workshop on Intelligent User Interfaces (January 1993), ACM Press, New York, pp. 81–88.

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

McFarlane, D., 1999. Coordinating the interruption of people in human-computer interaction. Human Computer Interaction—INTERACT '99, Edinburgh, Scotland—IOS Press, The Netherlands, pp. 295–303.

Milewski, A., Lewis, S., 1997. Delegating to software agents. International Journal Human Computer Studies 46, 485–500.

Riecken, D., 2000. Personalized views of personalization. Communications of the ACM 43 (8), 26–28.

Ruvini, J.D., Gabriel, J.M., 2002. Do users tolerate errors from their assistant?: experiments with an E-mail classifier. IUI 2002, Proceedings of the 2002 International Conference on Intelligent User Interfaces, January 13–16, 2002, San Francisco, CA, ACM, New York, pp. 216–217.

Scerri, P., Pynadath, D.V., Tambe, M., 2002. Why the elf acted autonomously: towards a theory of adjustable autonomy. The First International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS 2002. July 15–19, 2002, Bologna, Italy, Proceedings, pp. 857–864.

Schaumburg, H., 2001. Computers as tools or as social actors? The users' perspectives on anthropomorphic agents. International Journal of Cooperative Information Systems. Double Special Issue on "Intelligent Information Agents: Theory and Applications" 10 (2), 217–234.

Schiaffino, S., Amandi, A., 2002. On the design of a software secretary. In: Proceedings ASAI 2002, Argentine Symposium on Artificial Intelligence. September 2002, Santa Fe, Argentina, pp. 218–230.

Schiaffino, S., Amandi, A., 2003. Assisting database users in a web environment. In: Cueva Lovelle, J.M., González Rodríguez, B.M., Joyanes Aguilar, L., Labra Gayo, J.E., del Puerto Paule Ruiz, M. (Eds.), Web Engineering—Lecture Notes in Computer Science, Vol. 2722, Springer, Berlin, pp. 26–29.

Special Issue on Notification Systems, 2003. International Journal Human-Computer Studies, 58 (5), 509–632.

Tiernan, S.L., Cutrell, E., Czerwinski, M., 2001. Effective notification systems depend on user trust. In: Hirose, M. (Ed.), Human–Computer Interaction–Interact '01, IOS Press, pp. 684–685.

Whittaker, S., Sidner, C., 1996. Email overload: exploring personal information management of email. In: Conference proceedings on Human factors in computing systems, CHI 96, ACM Press, New York, 276–283.