Improving assistive technologies by learning user’s experience from traces : a theoretical contribution for next generation solutions

Bruno Mascret
Université de Lyon, CNRS
Université Lyon 1, LIRIS, UMR5205
F-69622, France
bruno.mascret@liris.cnrs.fr

1 A brief survey of assistive technologies issues

New technologies are leading us to be connected for everything. They make digital data access easier, they can be found in more and more connected devices, at work, at home, in the street, in your dishwasher, everywhere! Internet is now an interactive space where we can do practically everything. The notions of “cloud computing”, “ubiquitous computing”, “intelligent environments” are used in common language. This general trend represents a great opportunity for assistive technologies, and a lot of improvements have been made thanks to them. But it has also created new handicap situations, essentially with computer software and websites.

G.Uzan[11] introduces the “time disability” problem : for many people with disabilities, performing a task is possible but they need too much time to complete it compared to their colleagues. One of the great challenges in assistive software engineering is not so much to give access to software and documents but to guarantee that this access is temporally efficient. The problem is that most of the time, even with the generalization of the “designed for all” paradigm, assistive processes are triggered once the software has been released for the general public. Software designers cannot take into account the specificities of the future users, even with dedicated solutions. At least they may follow ergonomic and technical guidelines and recommendations, but they can not foresee every possible situation because it is just not possible.

Does that mean that assistive technologies will always come too late ? Not necessarily, but they need to be rethought. People with disability offer really rich and varied ways of using technologies, which could become a real advantage for everybody if we were able to understand and to model them. Let’s give an example : one of my teachers investigating the wheelchair area[1] was conducting an experiment with an automated arm. He didn’t understand why one of the subjects was using this arm with a fork all day long. He asked him why he was doing so, and the subject answered slightly embarrassed that it was the solution he had found to scratch his back! What does this story teach us ?

– Disabled people cannot wait for us and they are really skilled to find work-around solutions;
– They are the final users, so their uses are the real ones;
– They have a very valuable amount of implicit knowledge about their own disabilities, which would be of utmost interest if we were able to catch it.
One of the common mistakes is not having trusted enough the user’s capability to find their own solutions: we tried to build so powerful expert systems or dedicated software that we forgot the public they were intended for!

The questions now raised are: How could we acquire this user’s knowledge and how can we channel it into our solutions to improve our systems? To let them be more flexible, less dependent from the environment changes and more sensitive to the context? And first and foremost: how to share this knowledge? A great many of these questions are not so new at all in fact[2]! I modestly propose the onset of an answer in the next section.

2 Learning contextual knowledge from contextual experience with users

Trace based reasoning[3] is a paradigm of knowledge engineering strongly inspired by case based reasoning. We explained in [3] how it could improve accessibility to Internet by acquiring user’s experience[4] thanks to activity traces[7]. In my opinion, this paradigm of reasoning is intrinsically close to the real world and fully user-oriented, which is specifically what we need to acquire from user’s experience.

The TBR’s principle could be summarize as follows:

1) user’s activity is stored as (primary) traces: a trace is a sequence of observed elements or obsels (user’s interactions, events, documents used, and more generally activity information) temporally located and collected by the system;

2) a primary trace is not very interesting, because it has to be analyzed and documented to be understood and re-used. That’s why TRB proposes to manage transformations to build reformulations of the trace. These transformations are materialization of knowledge: a transformation is performed to adapt a traced experience to specific needs. And a more important point is that if someone sets up a transformation, this transformation can be learnt by the system. Thus transformations allow us to give a more abstracted vision of an activity, from a particular point of view.

3) a very important step is the elaboration of episodes and their signature. Given one or more transformations (that is to say one or several point of views about the traced activity), a signature is a description of relevant characteristics (obsels types, sequencing of obsels, temporality, etc.) describing a sequence of interest, something done which could be useful to memorize and to learn.

4) At a relevant level of abstraction, a current activity trace can be compared to a former one in real time: the principle is to recall similar episodes and adapt them to the current situation in a semi-automated way.

Thus people with disabilities would be helped reducing the time needed to use software and computer adaptations. They will have first to explain to the system how they use their computer or their devices, thanks to the collected traces, and what is important to learn for supporting them. Then the system would be able to retrieve past experiences and adapt them to their current situation.

Several kinds of knowledge are involved into the whole process: similarity, transformation, contextualization, recall, domain, etc. The main challenge is being able to mobilize the right knowledge at the right moment. I think that

---

[2] this relevance depends of course on the context and is not so easy to ascertain
this could be done by considering the whole assistive process as a workflow of knowledge mobilizations, but no more with fixed steps and frozen algorithms.

3 From theory to practice

This theoretical overview could seem a nice dream to many people. But we have good reasons to be confident about the possibility of such an approach. First, several systems using activity traces have been built over the last years\cite{7}\cite{5}. They do not implement all the TBR process but have shown good results. Second, recent works from major researchers about case based reasoning show the limit of this paradigm and the need for more provenance awareness\cite{9} or human interactions\cite{6}. Third, the idea of knowledge acquisition from the user’s interactions is a rising issue in assistive technologies\cite{8}. To me there are important signs toward which we have to strive.

My current work deals with knowledge acquisition for pedagogical graphs personalization in web based education\cite{10}. A pedagogical graph is an oriented organization of tasks assigned to a learner to reach a pedagogical objective. It includes different pedagogical resources linked together. Depending on a particular learner’s profile, capabilities or other contextual criteria, a teacher will propose different graphs. We want to facilitate the adaptation of these graphs to very different learners. One of the main objectives is to fight for literacy, but our approach has been designed to take into account other kind of profile and disabilities. Figure 1 shows the process of knowledge acquisition. During the pedagogical graph elaboration, the activity of the teacher is traced and memorized (step 1). We then ask the teacher to explain his work by transformations and the system will help him to formalize his knowledge. The graph is assigned to a specific learner (step 2) who interacts with it (step 3). These interactions are traced (step 4) and shown to the teacher who may revise the initial graph and explain the reasons for his changes.

![Fig. 1. The acquisition of knowledge in pedagogical graphs elaboration](image-url)
The knowledge acquired will then be re-used for new learners. Some adaptation skills are now formalized, so the teacher may save a lot of time. Moreover, the system may be able to directly propose a graph (maybe not the best one of course) if it has enough knowledge about the learner profile. The more the system is used, the more efficient it is.

4 To conclude

We have introduced the trace based reasoning, a new paradigm which the assistive technologies may benefit from. In my opinion, TBR seems a promising way to learn from users’ experience which is one of the major challenges in helping people with special needs. Nowadays, it is a knowledge ingeneering problem. I think that assistive technologies researchers could propose very great contributions in this field. They have developed new assistive solutions for many years, and a strong awareness in this area.

Their work in ergonomics, design for all and adaptations have put the user at the core of the processes. TBR is now one of the possible mechanisms which may help them formalize this experience and share it on a large scale.

References