Human Computer Interaction (HCI)

Lectures for 2nd Stage

(IT Department)

Lec. 4: Inference

Prepared by Sumaya Abdulla Hamad

Lecture 4: Inference

Mental models – what the user infers about the system

Don Norman, one of the first generation of cognitive scientists investigating HCI, also wrote the first popular book on the topic – *The Design of Everyday Things1*. What most people remember about this book is the example of door handles that are so badly designed they need labels telling you to pull them. But his key message was to draw attention to the ***gulf of evaluation*** and the ***gulf of execution***– how does the user know what the system is doing, and how do they know what they need to do, in order to achieve their goals?

Computer systems are so complex, that nobody really knows what is happening inside (except, possibly, the designer). In the face of incomplete information, the gulf of evaluation is unavoidable. The user has to make inferences (or guess) what is happening inside. The user's conclusions form a ***mental model*** of the system. One way of thinking about the design problem is that the designer must give sufficient clues to the user to support that inference process, and help the user form an accurate (or at least adequate) mental model. The idea of a ***visual metaphor*** is that the screen display simulates some more familiar real world object, and that the user's mental model will then be understood by analogy to the real world.

The metaphor/analogy approach can potentially help with the gulf of execution too. If the system behaved exactly like the real world objects depicted, then users would know exactly what to do with them. In practice, computer systems never behave exactly like real world objects, and the differences can make the system even more confusing. (Why do you have windows in your desktop? Why do I have to put my USB drive in the rubbish before unplugging it?) Furthermore, designers inadvertently create metaphors that correspond very well to their own understanding of the internal behaviour of the system, but users should not be expected to know as much as designers. User studies can help to identify what users actually know, what they need to know, and how they interpret prototype displays.

Mental models research

Mental models research attempts to describe the structure of the mental representations that people use for everyday reasoning and problem solving. Common mental models of everyday situations are often quite different from scientific descriptions of the same phenomena. They may be adequate for basic problem solving, but break down in unusual situations. For example, many people imagine electricity as being like a fluid flowing originally called the Psychology of Everyday Things –. When electrical wiring was first installed in houses, it appeared very similar to gas or water reticulation, including valves to turn the flow on and off, and hoses to direct the flow into an appliance. Many people extended this analogy and believed that the electricity would leak out of the light sockets if they were left without a light bulb. This mental model did not cause any serious problems - people simply made sure that there were light bulbs in the sockets, and they had no trouble operating electrical devices on the basis of their model.

The psychological nature of unofficial but useful mental models was described in the 1970s, and these ideas have been widely applied to computer systems. Young's study of calculator users in 1981 found that users generally had some cover story which explained to their satisfaction what happened inside the device. Payne carried out a more recent study of ATM users, demonstrating that even though they have never been given explicit instruction about the operation of the ATM network, they do have a definite mental model of data flow through the network, as well as clear beliefs about information such as the location of their account details.

The basic claim of mental models theory is that if you know the users' beliefs about the system they are using, you can predict their behaviour. The users' mental models allow them to make inferences about the results of their actions by a process of ***mental simulation***. The user imagines the effect of his or her actions before committing to a physical action on the device. This mental simulation process is used to predict the effect of an action in accordance with a mental model, and it supports planning of future actions through inference on the mental model. Where the model is incomplete, and the user encounters a situation that cannot be explained by the mental model, this inference will usually rely on ***analogy*** to other devices that the user already knows.

Think aloud studies

A great deal of cognitive psychology research, including some basic research on mental models, has been based on ***think-aloud*** studies, in which subjects are asked to carry out some task while talking as continuously as possible. The data are collected in the form of a ***verbal protocol***, normally transcribed from a tape recording so that subtle points are not missed. Use of this technique requires some care. It can be difficult to get subjects to think aloud, and some methods of doing so can bias the experimental data.

Performance models of users

Early HCI research was largely concerned with the performance of the user, measured in engineering terms as a system component ('cognitive psychology' is closely associated with 'artificial intelligence', investigating human performance by simulating it with machines). One of the most famous findings in cognitive psychology research, and the one most often known to user interface developers, is an observation by George Miller in 1956.

Miller generalised from a number of studies finding that people can recall somewhere between 5 and 9 things at one time - usually referred to as "***seven plus or minus two***".

Surprisingly, this number always seems to be about the same, regardless of what the "things" are. It applies to individual digits and letters, meaning that it would be very difficult to remember 25 letters. However if the letters are arranged into five 5-letter words (apple, grape …), we have no trouble remembering them. We can even remember 5 simple sentences reasonably easily. Miller called these units of short-term memory ***chunks***. It is rather more difficult to define a chunk than to make the observation - but it clearly has something to do with how we can interpret the information. This is often relevant in user interfaces - a user may be able to remember a sequence of seven meaningful operations, but will be unable to remember them if they seem to be arbitrary combinations of smaller elements. ***Short term memory*** is also very different from ***long term memory*** - everything we know.

Learning is the process of encoding information from short term memory into long term memory, where it appears to be stored by association with the other things we already know. Current models of long-term memory are largely based on ***connectionist*** theories - we recall things as a result of activation from related nodes in a network. According to this model, we can improve learning and retrieval by providing ***rich*** associations - many related connections. This is exploited in user interfaces that mimic either real world situations or other familiar applications.

A further subtlety of human memory is that the information stored is not always verbal. Short term memory experiments involving recall of lists failed to investigate the way that we remember visual scenes. Visual working memory is in fact independent of verbal short term memory, display icons combined with associated labels provide this kind of ***dual coding***.

Intelligent interfaces – what the system infers about the user

A further inference problem is that, in addition to the user not knowing what is happening inside the system, the system doesn't 'know' what is happening inside the user. Advanced systems can be designed to record and observe user interactions, and on the basis of that data, make inferences about what the user intends to do next, and present short-cuts, usability cues or other aids. These kinds of 'intelligent user interface' are becoming more common, but they can also introduce severe usability problems. A notorious early example was the Microsoft Word 'Clippy', which analysed features of the document, and offered to help with automatic formatting ("You appear to be writing a letter …"). Although some users found it useful, a far larger number found the tone patronizing and the automated actions inaccurate. Google 'Death to Clippy' to see the extent to which smart user interface technology can get it wrong.

Many intelligent user interfaces emerge from the machine learning community, and especially ***Bayesian inference*** techniques. Bayesian techniques are more appropriate to user interfaces than other techniques for a range of reasons:

- They don't rely on large training sets (as is the case with neural net approaches), so they can adapt more quickly to individual users
- Bayesian consideration of prior probabilities corresponds better to commonsense human reasoning under uncertainty.
- Bayes formula provides a consistent way to combine data from user interactions with historical data and heuristic rules.

An inference framework provides a valuable analytic perspective on many current trends in user interaction. For example, the behaviour of Google, or of recommender systems such as Amazon or Facebook friend finder, use inference techniques to apply statistical data and guess what the user really wants. It remains the case that when the system makes inaccurate inferences, the results will be annoying, confusing, or even damaging. This means that some advanced research areas, such as ***Programming by Example*** (where automated scripts or macros are created by inference, after observing repeated actions) provide a major challenge for HCI.

References

- http://hcibib.org/
- Alan Blackwell, Human Computer Interaction – Lecture Notes, Cambridge Computer Science Tripos, Part II