STEPHAN SCHLÖGL: SKETCHING EXPERIENCES WITH LANGUAGE TECHNOLOGY

The use of Language Technology Components (LTCs), e.g. Automatic Speech Recognition (ASR), Machine Translation (MT) and Speech Synthesis (SS), has significantly increased in recent years as their performance has improved. Examples include speech-based interaction in cars that keep a driver's attention on the road, and the use of web-based translation tools such as *Google translate*[1] and *Yahoo! Babel Fish*,[2] which facilitate the understanding of text written in a foreign language.

As with applications based on a Graphical User Interface (GUI), software that uses LTCs also needs to be tested early in the design process. Whereas low-fidelity prototyping for GUI applications can be done relatively quickly and inexpensively, through sketches and wireframes, the development of prototypes evaluating applications that use LTCs can be both cost and time intensive.

One technique that has been used in the past to test software involving speech and language is Wizard of Oz (WOZ). Based on Baum's famous novel [1], a WOZ experiment uses a 'human wizard' to mimic some of the functions of a future system that currently do not exist or are too faulty. Since the technical requirements for such a prototype can be reduced to a minimum, the WOZ technique is particularly useful for early stage evaluations, and thus a good candidate for addressing this lack of low-fidelity prototyping methods evaluating the use of LTCs.

Yet, there are more things required than just pen and paper in order to somewhat realistically 'sketch' speech-based interaction. In the case of WOZ, for example, one needs to create a WOZ tool consisting of a wizard as well as a client component, design a certain dialogue to be tested, and define the machine-like behaviour that would represent the possibilities of a future system. In order to convey the same user experience there is also some sort of synthesised speech output or at least a distortion mechanism needed. In summary it might take a couple of days of programming, recording, and testing until a prototype is built that can be used to run experiments and therefore get feedback on the interaction. Compared to some quick sketches on a sheet of paper, this seems too much of an effort for low-fidelity prototyping.

My work aims to address this problem by coming up with a WOZ prototyping framework that is as easy and efficient to use as pen and paper. The designer should

only be concerned with the dialogue that needs to be tested. No programming work should be needed and technology components like ASR, MT and SS in various quality levels should be integrated through the click of a button. The goal is to make running WOZ experiments as easy as sketching screen layouts and therefore pointing to its qualities as a low-fidelity prototyping technique.

Why something new?
The question arises: Why do we need yet another prototyping tool to cope with the lack of WOZ support? Why can't we just use one of the standard prototyping tools that are out there and adapt it to test speech-based interactions? The main reason here is the imperfect nature of the technology to be tested. Traditional prototyping techniques have the goal of coming up with a rather flawless final product – iterative testing aiming for the identification of all the problems and uncertainties a user would eventually struggle with. Due to the fallibility of speech-based applications, however, a different prototyping methodology is required and needs to be supported by the tool. Of course, as designers and researchers we are always searching for an optimal solution. Yet, with speech this perfect solution in which the recogniser would understand every word, the MT would not make any translation errors and the speech output would be as natural as a real person, is rarely possible or might just require a disproportionate amount of resources. Therefore designers need to find solutions that might not be perfect but will be accepted by the users of the system.

What is my goal?
Knowing the restrictions of the domain, the goal of my work is to support designers and researchers by providing them with a tool that helps them to explore the acceptability and possibilities of using LTCs for different application scenarios. More concretely, I am aiming for a WOZ prototyping framework that allows for the generic creation of WOZ prototypes, which support the testing and evaluation of software applications using LTCs. In addition to creating this tool I want to understand and optimally support the task of the wizard. It was highlighted that playing the wizard is cognitively highly demanding [3] and that supporting this task helps to create a more realistic experiment setup. My goal therefore is to design a generic wizard interface that takes away some of the cognitive load and helps the wizard to be consistent and as machine-like as possible.

What has been done and what is next?
In order to create a WOZ tool that would meet the stated requirements, I use a User-Centred Design [4] development process. A first iteration has already been performed. The goal here was to get insight into the process of running WOZ experiments and to obtain a basic understanding of the task of the wizard. Furthermore it was sought to generally discover the domain of prototyping software applications using LTCs.
Based on the literature and inspired by situations that were explored in previous WOZ studies I was able to define four experimental scenarios in which WOZ could be used as a prototyping technique. Elaborating on one of them I built a first WOZ tool and evaluated it. The evaluation took place in two steps.

First a formal usability study [2] was conducted in which four users were confronted with the task of the wizard and asked to interact with the provided wizard interface. A third person was observing them while I was sitting in a different room mimicking the test subject. In a second study I was acting as the wizard myself, using the wizard interface to interact with 12 real test subjects while being observed by a third person. The aim of this second study was to foster my own understanding of what it means to be a wizard and supplement this with the data collected during the usability tests.

Reflecting on the results of this first study I am currently working on an improved version of the WOZ tool, which will hopefully allow me to identify further aspects of the task of the wizard as well as the challenges of designing speech-based interaction.

1 **translate.google.com**
2 **babelfish.yahoo.com**

Stephan Schlögl is a 3rd year PhD student in the School of Computer Science and Statistics at Trinity College Dublin. He works and is funded within the Centre for Next Generation Localisation (**www.cngl.ie**), a dynamic academia–industry partnership with over 100 researchers developing novel technologies that address the key localisation challenges of volume, access and personalisation.
stephan.schloegl@gmail.com

REFERENCES

1 Baum, L.F. (1900). *The Wonderful Wizard of Oz*. Hill.
2 Dumas, J.S., & Redish, J.C. (1999). *A Practical Guide to Usability Testing*. Intellect.
3 Salber, D., & Coutaz, J. (1993). A wizard of oz platform for the study of multimodal systems. In *Proceedings of INTERACT and CHI*, 95–96.
4 Travis, D. (2009). *The Fable of the User-Centred Designer*. Userfocus.

MY PHD

If you are a PhD student just itching to tell the world about your research or if you've enjoyed reading about some of the emerging areas of research that the My PhD column has recently discussed then we would like to hear from you. We are currently accepting one to two page summaries from PhD students in the UK and across Europe with a focus on being open and accessible to everyone in the HCI community.

If you would like to submit or would just like more information please contact xx using the contact information below.

Dr Shaun Lawson, Reader in School of Computer Science, Director, Lincoln Social Computing (LiSC) Research Centre, University of Lincoln, UK

http://lisc.lincoln.ac.uk/shaun