

Visualization or Not of Uncertainty in Context Aware Mobile Applications

¹ Enrico Rukzio, ² John Hamard, ² Chie Noda, ^{2/1} Alexander De Luca,
¹ Embedded Interaction Research Group, Media Informatics Group, University of Munich, Germany
² DoCoMo Euro-Labs, Munich, Germany
enrico@hclilab.org, {hamard, noda, deluca}@docomolab-euro.com

ABSTRACT

Context-aware mobile applications and systems have been extensively explored in the last decade and in the last few years we already saw promising products on the market. Most of these applications assume that context data is highly accurate. But in practice this information is often unreliable, especially when gathered from sensors or external sources. Previous research has argued that the system usability can be improved by displaying the uncertainty to the user. The research presented in this paper shows that it is not always an advantage to show the confidence of the context-aware application to the user. We developed a system for automatic form filling on mobile devices which fills in any web form with user data stored on the mobile device. The used algorithm generates rules which indicate with which probability which input field of a form should be filled in with which value. Based on this we developed two versions of our system. One shows the uncertainty of the system and one not. We then conducted a user study which shows that the user needs slightly more time and produces slightly more errors when the confidence of the system is visualized.

Categories and Subject Descriptors

H.5.2 [Interfaces and Presentation]: User Interfaces — Interaction styles; I.3.6 [Computer Graphics]: Methodology and Techniques — Interaction techniques.

General Terms

Measurement, Design, Experimentation, Human Factors.

Keywords

Context aware, visualization of uncertainty, mobile application, evaluation, user study.

1. MOTIVATION

In the last years we saw a big interest in context-aware systems in industry and academia. Because of this a huge set of approaches for gathering, describing, structuring and the use of context information exist, see [1] for a longer survey. Not surprisingly, all these work deals with people, locations, devices, services, networks and their static and dynamic relationships.

Most applications that take such context information into account assume that this data is highly accurate. But in practice context information is often unreliable, especially when gathered from sensors or when requested from unfamiliar resources. Typical examples therefore are location information gathered from GPS

devices or information about the current activity of a person based on audio or acceleration sensors [2].

Some research projects in this field therefore provided feedback to the users which enabled them to control or to monitor the behavior of the context-aware system. Belotti and Edwards [3] defined four design principles for intelligibility and accountability in context-aware systems. Their first principle is “*inform the user of the current contextual system capabilities and understandings*”. Greenberg [4] proposes also that the user should be able to see and control which context information the system currently uses, how this is combined and which high-level context information the system has inferred from the low-level context information. Chalmers and MacColl [5] go even one step further and argue for a seamful rather than a seamless design in ubiquitous computing applications. Anitfakos et al. [6, 9] reported also positive effects when displaying uncertainty or the confidence of the system in context-aware applications. Most previous research argued that way.

Our work in that area is based on a system for automatic form filling on mobile devices which is similar to the form filling functionalities of the MSN Search Toolbar and the Google Toolbar for Firefox used on a PC [8]. We showed in [7] that forms are currently very seldom used in mobile applications. The reason for that is that it takes much time to fill in information like first name, family name, address, and e-mail address when using the limited input capabilities of mobile devices. Our system overcomes this problem and automatically fills in such forms based on the users profile stored on the mobile phone.

The uncertain context information in our case are the forms of web pages that have to be filled in by our system. The forms that can be found in the web are very different regarding the requested information, the labels related to the input fields, the sequence of input fields and the implementation within an (X)HTML, WAP or i-mode document.

The algorithm which we developed generates the following kind of rules: *The probability is X% that this field should be filled with the family name of the user.* After the development of our system we therefore decided to test whether the visualization of the uncertainty of the algorithm helps the user or not. Herby we measured whether the user is faster or whether he makes fewer errors when the uncertainty is visualized.

We conducted a user study with 18 volunteers and compared three versions of our prototype: one without the visualization of uncertainty, one with the visualization of uncertainty whereby the algorithm was mostly certain regarding the filled in forms and one

with the visualization of uncertainty whereby the algorithm was not so certain regarding the filled in forms.

The paper is organized as follows. The next section discusses the prototype we used in a compact way. Afterwards we describe the setup of the conducted user study. In section 4 we discuss then the results of the user study and its implications. The paper is completed by a discussion and outline of our further work.

2. PROTOTYPE

Figure 1 shows the architecture of our application for automatic form filling on mobile devices [7, 8]. The form filling proxy installed on the mobile device acts as a mediator between the web browser and the web server hosting a requested web page. The form filler in the proxy fills input fields of forms with user data locally stored by analyzing the context of the input fields. Hereby initial rules (locally stored or downloaded) are used to generate dynamic rules. This leads to the highest probable user data to be filled in a form. The rule server is an external component, which stores and provides the form filling rules. The rule repository of the rule server enables updating the rules, for example monitoring users' behaviours and adding new concept names. The proxy of the mobile device uses it to keep the local rule set up-to-date.

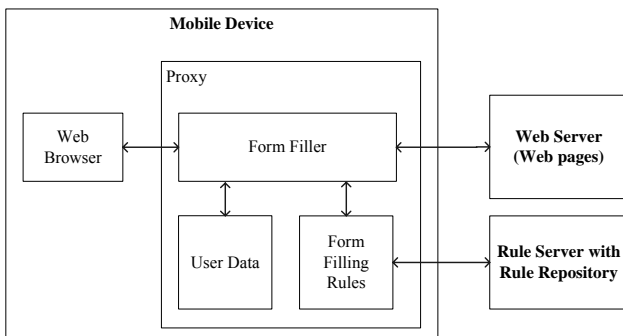


Figure 1. Architecture of the automatic form filling prototype

We implemented the prototype using Nokia Series 60 phones (Nokia 6630 / 6600 / N90) and J2ME (MIDP 2.0, CLDC 1.1).

3. EVALUATION SETUP

The user study evaluating the effect of the visualization of uncertainty consisted of three phases which were conducted by every participant: a preliminary interview asking for personal data, performing the tests with the prototype and an interview at the end of the test. Our team consisted of two persons, one filled in all information that was provided by the test person and the other person spoke with the test person and gave her the mobile phone with the test settings.

We had 18 volunteers that participated in our study, 9 women and 9 men, aged from 20 to 26. They were all students studying computer science, communication science, politics, ethnology, a foreign language or literature.

At the beginning we explained to every participant of the study the idea of automatic form filling on mobile devices. We discussed with them the problem of filling in forms. We said that the forms are just pre-filled and the user has to check them before they are transmitted to a server. In addition to that we explained that in some of the forms the input fields are marked by colors

reflecting with what probability the content is correct or not. This is depicted in figure 2. We said that the colors have similar meanings as they have with traffic lights. Red means “danger”, there is probably not the correct content in this field. Yellow means “pay attention”, there might not be the correct content in this field. Green means “OK”, there is probably the correct content in this field.

Field	2a	2b	2c
First Name	Marco (Green)	Marco (Green)	Polo (Green)
Last Name	Polo (Green)	Polo (Green)	Polo (Green)
Address	Polostr. 33 (Green)	marco@polo.ifg (Yellow)	80333 (Yellow)
Postal Code	80333 (Green)	80333 (Green)	80333 (Red)
City	München (Green)	München (Green)	München (Yellow)
Phone Number	555-352612 (Green)	555-352612 (Green)	555-352612 (Red)
Email	marco@polo.ifg (Green)	marco@polo.ifg (Red)	marco@polo.ifg (Green)

Figure 2. Examples of filled forms

As the next step the users were asked to provide their first name, last name, address, postal code, city, phone number and email address. We stored this information into an application on a nearby PC. Later on this information was used by the form filling application to automatically fill in the forms with the personal data of the current test person.

Every test user was asked to check 20 differently pre-filled forms. Thus we presented 360 forms altogether (= 18 participants * 20 forms). These 20 settings are depicted in table 1. The first 3 cases (No. 1-3, see also figure 2a) did not show any information about the uncertainty of the algorithm. In the cases 4-7 the probabilities were visualized by showing a form filled with values that were correct with a high certainty (5 fields were marked green, 1 was marked yellow and only 1 red; see also figure 2b). In the cases 8-14 (see also figure 2c) the probabilities were visualized by showing a form filled with values that were correct with low probability only (3 fields were marked green, 2 were marked yellow and 2 red). The column labeled “Number of included errors, how visualized” of table 1 shows how many errors were included in the pre-filled form. In those cases where probabilities were visualized, this column also shows the color(s) (r-red, y-yellow, g- green) with which the incorrectly filled fields were marked.

To avoid that a test user gets used to a particular setting, we randomly generated different sequences of the 14 test cases depicted by table 1.

Table 1. Tested Combinations and Results

No.	Runs per person	Visualized probabilities	Number of included errors, how visualized	Frequency of errors: percent (sum of all errors)	Average time in seconds
1	3	No	0	0% (0)	5,19
2	3		1	2% (1)	5,66
3	3		2	11% (6)	6,26
4	1	5 green,	0	0% (0)	5,30

5	1	1 yellow,	1 (r)		5,90
6	1	1 red	1 (y)	2% (1)	6,01
7	1		1 (g)		5,57
8	1	3 green,	0	0% (0)	5,83
9	1	2 yellow,	2 (r+r)	11% (2)	6,78
10	1	2 red	2 (g+g)	11% (2)	6,86
11	1		2 (y+y)	6% (1)	6,36
12	1		2 (y+r)	6% (1)	6,24
13	1		2 (g+r)	33% (6)	6,89
14	1		2 (g+y)	6% (1)	7,07

Each run was executed according to the following scheme: After selecting the test case (No. 1-14), we waited until the form was completely loaded and filled in. Then we put the mobile phone onto the table in front of the user with its display faced down. As shown in Figure 3, the task of the user was to turn the mobile phone around, check the pre-filled form, find potentially existing errors and turn the mobile phone around again when ready. There was no need for scrolling because the whole form fitted on the screen.



Figure 3. Actions done by the test user

We used these gestures to exactly measure the time between the first look on the display and the moment where the user turned the display back. Afterwards we asked the test users to tell if the form was correctly filled in with their personal data or if there were errors. Through asking for the errors after the mobile phone was turned again we prevented to measure the time the different participants needed to explain the errors. We measured the time the users needed to recognize the errors only. In the cases where the user found errors we asked in which field a wrong content was filled in.

At the end of the 20 runs we asked the participants the following questions:

- Would you use such a system for automatic form filling?
- What do you prefer? The cases where the probabilities were visualized or the cases without any visualization of probabilities?
- How did you take into consideration the visualization of the uncertainty expressed with colors?
- Do you think you were faster when the probabilities were visualized?
- Do you think you found more errors when the probabilities were visualized?

4. RESULTS AND OBSERVATIONS

In generally it can be said that that there were no significant differences in the time the participants needed for the different test cases.

One result is that the more errors were included the more time the participants needed for completing a run as shown in the following table 2. The participants needed 23% more time when 2 errors were included when compared to the cases without any errors. Furthermore not surprisingly the frequency of errors was higher than 0% when 1 or 2 errors were included in the pre-filled form. An error could be either an incorrectly filled in input field that was not recognized by the test person or the test person mentioned an error in a field that did not exist.

Table 2. Needed time and frequency of errors related to the included errors


Errors	0	1	2
Runs	90	108	162
Average needed time in seconds	5,34	5,75	6,55
Frequency of errors: percent (sum of all errors)	0%	2% (2)	12% (19)




Other results of the user study are depicted in table 1. The fifth column shows the frequency of errors in percent and the sum of all errors in brackets. A value of “6” means, e.g., that of all runs of all participants, 6 errors were recognized. The last column shows the average time the participants needed for checking the pre-filled form.

As depicted in table 3 the visualization of probabilities did not have any influence on the frequency of errors. It was 0% when no errors were included, it was 2% when 1 error was included and it was 11-12% when 2 errors were included. In the last case with 2 errors one could even argue that the visualization of probabilities raised the frequency of errors.

When looking at the average time the participants needed, then it can be seen that they needed more time for checking the forms when probabilities were visualized when compared to the cases without visualization. This also corresponds to the feedback we got from the participants. Many of them mentioned that they were distracted by the colors. They stated that they would check every pre-filled field before submitting the form anyway because things like booking a hotel room or renting a car is very important for a business trip. They would use the automatic form filling but they would carefully check every field. Many of them stated that they first checked all fields and then double-checked the fields marked red and yellow. Therefore they needed more time for checking a pre-filled form when uncertainty was visualized.

Table 3. Needed time and frequency of errors related to the included errors and the visualization of uncertainty

Included errors	Visualized probabilities	Frequency of errors: percent (sum of all errors)	Average time in seconds
0	No	0% (0 in 54)	5,19
	 5 g, 1 y, 1 r	0% (0 in 18)	5,30

	 3 g, 2 y, 2 r	0% (0 in 18)	5,83
1	No	2% (1 in 54)	5,66
	 5 g, 1 y, 1 r	2% (1 in 54)	5,83
2	No	11 % (6 in 54)	6,26
	 3 g, 2 y, 2 r	12% (13 in 108)	6,70

We observed an interesting effect depicted at number 13 of table 1. Here 2 errors were located in 2 fields; one was marked red and the other one green. Some test users recognized the error in the red field, but they missed the error in the green field. This result is very interesting because most of the participants stated that they did not take the visualized probabilities into account. But this example shows that the visualization of the probabilities can subconsciously mislead the participants.

After the runs we asked the participants five questions based on their experience with the 20 runs. 83% (15 of 18) of the test user would use such an automatic form filling function if available. Then we asked whether and how they used the visualized probabilities. All 18 participants responded that she or he did not use or pay attention to the colors. 17% (3 of 18) mentioned, even we did not ask this question, that they think they subconsciously took the visualized probabilities into account. Furthermore 22% (4 of 18) mentioned that they think that the colors were more disturbing than helpful.

Only 28% (5 of 18) mentioned that they prefer the visualization of the probabilities whereas the others answered in negative. At the end we asked if they thought they were faster when the probabilities were visualized. Here 88.9% (16 of 18) answered in a negative way.

5. CONCLUSION

This paper addressed the question whether a context-aware mobile system should visualize its confidence regarding its decisions or not. Previous research often assumed that context information is always highly accurate which is not true for practical applications. Most projects that addressed this problem argue for the visualization of the uncertainty. Their research showed that if the user can control or monitor the current state or the behavior of the system, then this improves the usability of such a system.

Our research has shown that this is not always true. We conducted a user study based on an application for automatic form filling on mobile devices. We compared two versions of that system. One showed the uncertainty of it and one did not.

The study showed that visualizing the uncertainty or certainty of the system was mostly not used nor was it helpful. Most people checked every field to see whether it was filled in correctly and they stated that they would probably not take the visualized

probabilities into account. From this we conclude that in general the visualization of uncertainty in context-aware systems is still questionable.

An important factor for this is the importance of the visualized uncertainty. In our case the pre-filled forms can be used to order a product or to book a car. These are critical actions in which the user wants to be sure that she gets the product or that the hotel room is really booked. But there are other application areas in which a wrong decision based on imprecise context information is not so important. An example for this is a mobile application that recognizes the situation of the user based on sensor data and changes the profile of the mobile phone (e.g. normal, quiet or noisy environment) according to this. It could be regretful if the mobile phone rings in a business meeting but this is often not so critical.

Currently we work on other prototypes of context aware mobile applications that can visualize the uncertainty or not. Based on this we will conduct further user studies to analyze in which situations the confidence should be visualized or not. Our goal is the development of guidelines which define in which situations the uncertainty should be shown and how this should look like.

6. ACKNOWLEDGMENTS

This work was performed in the framework of the Simplicity project and of the SMS project, partly funded by the European Union. The authors wish to express their gratitude to the members of the Simplicity and SMS consortium for their valuable contributions.

7. REFERENCES

- [1] Chen, G. and Kotz, D. *A Survey of Context-Aware Mobile Computing Research*. Technical Report: TR2000-381 Dartmouth College, 2000.
- [2] Schmidt, A. and van Laerhoven, K. *How to Build Smart Appliances?* IEEE Personal Communications. 8 (4), 2001, 66-71
- [3] Bellotti, V. and Edwards, K. *Intelligibility and Accountability: Human Considerations in Context-Aware Systems*. Human-Computer Interaction, Volume 16, 2001, 193-212
- [4] Greenberg, S. *Context as a Dynamic Construct*. Human-Computer Interaction, Volume 16, 2001, 257-268
- [5] Chalmers, M. and MacColl, I. *Seamful and Seamless Design in Ubiquitous Computing*, Technical Report-Equator-03-005.
- [6] Antifakos, S., Schwaninger, A., Schiele, B. *Evaluating the Effects of Displaying Uncertainty in Context-Aware Applications*. In Ubicomp'04, 6th International Conference on Ubiquitous Computing, Nottingham, September 2004.
- [7] Removed for blind review process.
- [8] Removed for blind review process.
- [9] Antifakos, S., Kern, N., Schiele, B., Schwaninger, A. *Towards Improving Trust in Context-Aware Systems by Displaying System Confidence*. In 7th International Conference on Human Computer Interaction with Mobile Devices and Services (MobileHCI), Salzburg, Austria, 2005.

