Probabilistic Explanation Dialog Augmentation

Florian Nothdurft*, Felix Richter† and Wolfgang Minker*
*Institute of Communications Engineering,†Institute of Artificial Intelligence
Ulm University
Ulm, Germany
florian.nothdurft,felix.richter,wolfgang.minker@uni-ulm.de

***Abstract*—Human-computer trust (HCT) is an important factor influencing the complexity and frequency of interaction in technical systems. Especially incomprehensible situations in human-computer interaction (HCI) may decrease the users trust and through that the way of interaction. However, analogous to human-human interaction (HHI), providing explanations in these situations can help to remedy negative effects. In this paper, we present our approach of augmenting task-oriented dialogs with selected explanation dialogs to stabilize the HCT relationship. We conducted a study comparing the effects of different explanations on HCT. These results were used in a probabilistic trust handling architecture to augment pre-defined task-oriented dialogs.**

***Keywords*-User-centered design, Human factors, User interfaces.**

I. Introduction

HCI has evolved in the past decades from classic stationary interaction paradigms featuring only human and computer to intelligent agent-based paradigms featuring multiple devices and sensors in intelligent environments. For example, ubiquitous computing no longer seems as a vision of future HCI, but has become, at least in research labs and prototypical environments, reality. Additionally, the tasks a technical system has to solve with the user have changed into more complex ones. This change from simple task solver to intelligent assistant requires the acceptance of, and the trust into the technical system as a dialog partner by the user and not only as ordinary servant.

HCT can be defined as "the attitude that an agent will help achieve an individual's goals in a situation characterized by uncertainty and vulnerability" [1]. HCT has shown to be a crucial part in the interaction between humans and technical system. If the user does not trust the system and its actions, advice or instructions the way of interaction may change up to complete abandonment of future interaction [2]. Especially those situations in which the user does not understand the system or does not expect the systems action are critical to have a negative impact on the HCT relationship [3]. Those situations do occur usually due to not matching models of the system. During interaction the user builds a mental model of the system and its underlying processes determining system actions and output. However, if this perceived mental model does not match the actual system the HCT relationship may be influenced negatively, because the expected processes and their outcomes do not match the expected ones [3].

Therefore, the goal should be to detect those critical situations in HCI and react appropriately. If we take a look at how humans detect and handle those situations, we can conclude that they use contextual information combined with interpreted multimodal body analysis (e.g., facial expression, body posture, speech prosody) for recognizing these situations and usually some sort of explanation to clarify the process of reasoning (i.e. increasing transparency and understandability). This process is related to the psycholinguistic concept of grounding [4], which describes the intent to achieve a so-called common ground between at least two participating parties in a conversation. However, as even humans are sometimes insecure judging the opposites state and to decide whether and which type of reaction would be appropriate, it seems logical that a technical system will not overcome this issue of uncertainty. Because of that, we think that the transfer of this problem to a technical system can only be handled effectively by incorporating uncertainty and thus using a probabilistic model. In the remainder of this paper, we will first elaborate how to react to not understandable situations and secondly present how to incorporate these findings into a dialog system using a probabilistic model.

Goals	Details
Transparency	How was the systems answer reached?
Justification	Explain the motives of the answer?
Relevance	Why is the answer a relevant answer?
Conceptualization	Clarify the meaning of concepts
Learning	Learn something about the domain

Table I
GOALS OF EXPLANATION AFTER [6].

II. Handling Critical Situations

Analogous to HHI providing explanations in incomprehensible situations in HCI can reduce the loss of trust [5]. However, HCT is not a one-dimensional simple concept. It consists of several bases, which all have to be intact in order to have the user trust a technical system. Previous studies have concentrated on showing that explanations or different kinds of explanations can influence HCT in general. So, what is currently lacking is a mapping showing which goals or kinds of explanations do influence which bases of trust.

A. Explanations

In general explanations are given to clarify, change or impart knowledge, with amongst other things, the implicit idea of aligning the mental models of the participating

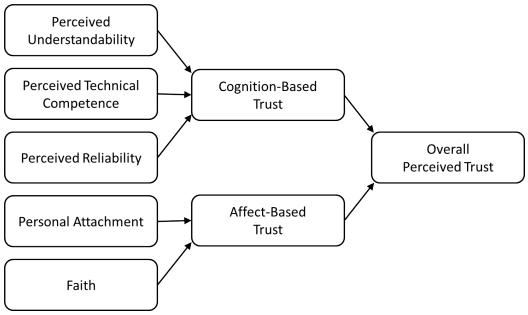

Figure 1. Human-computer trust model: Personal attachment and faith build the bases for affect-based trust, whereas perceived understandability, perceived technical competence and perceived reliability build those for cognition-based trust.

parties. The mental model is the perceived representation of the real world, or in our case of the technical system and its underlying processes. In this context explanations try to establish a common ground between the parties in the sense that the technical system tries to clarify its actual model to the user. This is the attempt of aligning the user's mental model to the actual system. However, there exist different goals of explanation (see table I for a listing of explanation goals).

B. Human-Computer Trust

A definition of trust mapped to HCI is for example, "the attitude that an agent will help achieve an individual's goals in a situation characterized by uncertainty and vulnerability" [1]. However, HCT should, as already mentioned, not be viewed as a simple concept but as a complex one consisting of several bases. For HCT Madsen and Gregor [7] constructed a hierarchical model (see figure 1) resulting in five basic constructs of trust. Important in this case are the cognitive-based components, which may be influential in short-term HCI. Perceived understandability can be interpreted in the sense that the human supervisor or observer can form a mental model and predict future behaviour of the system. The perceived reliability of the system, in the usual sense of repeated, consistent functioning. Technical competence means that the system is perceived to perform the tasks accurately and correctly based on the input information.

III. Related Work

Previous work on optimising trust issues in technical systems has been done for example by Glass et al. [5]. They investigated factors that may change the level of trust users are willing to place in adaptive agents. Among these verified findings were statements like "provide the user with the information provenance for sources used by the system", "intelligently modulating the granularity of feedback based on context- and user-modeling" or "supply the user with access to information about the internal workings of the system". However, what is missing in this work is the idea of rating the different methods to uphold HCT in general and a more complex model of HCT as well. Related work on how different kinds of explanations can improve the intelligibility of context-aware intelligent systems was done by Lim et al. [8]. They concentrate on the effect of Why, Why-not, How-to and What-if explanations on trust and understanding the system's actions. The results showed that Why and Why-not explanations were the best kind of explanation to increase the user's understanding of the system, though trust was only increased by providing Why explanations. Drawbacks of this study were that they did only concentrate on understanding the system and trusting the system in general and did not consider that HCT is on the one hand not only influenced by the user's understanding of the system but on the other hand that if any base of trust is flawed, the HCT in general will be damaged [9].

IV. Experiment

If we want to use system-generated explanations to influence the HCT relationship in a directed rather than arbitrary way, we need to find the most effective mapping of explanation goals to HCT bases. Thereby, undirected strategies to handle HCT issues can be changed into directed and well-founded ones, substantiating the choice and goal of explanation.

For that, we conducted a web-based study inducing events to create incomprehensible or unexpected situations and then compared the effects of the different goals of explanations on the bases of trust. For our experiment we used justification and transparency explanations. Justifications are the most obvious goal an explanation can pursue. The main idea of this goal is to provide support for and increase confidence in given system advice or actions. The goal of transparency is to increase the user's understanding of how the system works and reasons. This can help the user to change his perception of the system from a black-box to a system the user can comprehend. Through this, the user can build a mental model of the system and its underlying reasoning processes.

The experiment consisted in total of four rounds. The first two rounds were meant to go smoothly and were supposed to get the subject used to the system and through that building a mental model of it. After the first two rounds a HCT questionnaire was presented to the user. As expected the user did build a solid HCT relationship to the system by gaining an understanding of the system's processes. The next two rounds were meant to influence the HCT-relationship negatively by *unexpected* to the user external events. These unexpected, and incongruent to the user's mental model, system events were pro-actively influencing the decisions and solutions the user made to solve the task (e.g., the amount of ordered food was changed pro-actively by the system). This means that without warning, the user was overruled by the system and either simply informed of this change, or was presented an additional justification or transparency explanation.

A. Results

139 starting participants were distributed among the three test groups (no explanation, transparency only, justifications only). 98 completed round 2, reaching the point

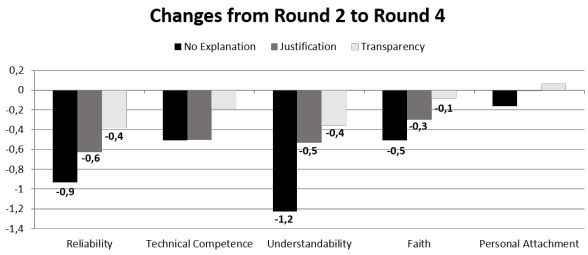

Figure 2. This figure shows the average changes of HCT bases from round 2 to round 4. The scale was a 5 point Likert scale with e. g., 1 the system being not understandable at all and 5 the opposite.

until the external events were induced and 59 participants completed the experiment. The first main result was that 47% from the group receiving no explanations quit during the critical rounds 3 and 4. However, if explanations were presented only 33% (justifications) and 35% (transparency) quit. This means that the use of explanations in those critial situations can help to keep the HCI running. The main results from the HCT-questionnaires can be seen in figure 2. The data states that providing *no explanations* in rounds three and four resulted in a decrease in several bases of trust. Therefore we can conclude that the external events did indeed result in our planned negative change in trust. *Perceived understandability* diminished on average over the people questioned by 1.2 on a Likert scale with a range from 1 to 5 when providing no explanation at all compared to only 0.4 when providing *transparency* explanations (no explanation vs. transparency $t(34)=-3.557$ $p<0.001$), and on average by 0.6 with *justifications* (no explanation vs. justifications $t(36)=-2.023$ $p<0.045$). Omitting explanations resulted in an average decrease of 0.9 for the *perceived reliability*, with transparency explanations in a decrease of 0.4 and for *justifications* in a decrease of 0.5 (no explanation vs. transparency $t(34)=-2.55$ $p<0.015$). These results support our hypotheses that transparency explanations can help to reduce the negative effects of loss of trust due to unexpected situations. Especially for the base of *understandability*, meaning the prediction of future outcomes, transparency explanations fulfill their purpose in a good way. Additionally, they seem to help with the users' perception of a system as reliable and consistent. The results show that it is worthwhile to augment ongoing dialogs with explanations in order to maintain HCT. In the following, we will describe how this is used in our developed *explanation augmentation* architecture.

V. Implementation

In order to decide when to give additional explanations, on one hand critical situations in HCI have to be recognized and on the other hand, if necessary the appropriate type of explanation has to be given. Obviously, recognizing those situations cannot be done solely by only using information coming from the interaction and its history. Multimodal input such as for example the accuracy of the speech recognition hypothesis, facial expressions or any other sensor information can help to improve the accuracy of recognizing critical moments in HCI (e.g., by detecting when the person is surprised, puzzled or simply not engaged). However, mapping sensor input to semantic information is usually done by classifiers and those classifiers convey a certain amount of probabilistic inaccuracy which has to be handled. Therefore, a decision model has to be able to handle probabilistic information in a suitable and appropriate manner.

A. Probabilistic Decision Model

For the problem representation of when and how to react a so-called partially observable Markov decision process (POMDP) was chosen and formalized in the Relational Dynamic Influence Diagram Language (RDDL) [10]. Formally, a POMDP consists of a set S of state variables, a set A of system actions, and a set O of all possible observations of the system. Furthermore, transition probabilities $P(s'|s,a)$ and observation probabilities $P(o'|s',a)$ are included. As the state of the underlying process cannot be determined exactly, a probability distribution over all possible states, called the *belief state* $b(s)$, is used instead. (For more information on POMDPs, see [11].)

The RDDL describes the probabilistic model of the domain, which determines when and how to augment the dialog at run-time. Observations o are the duration of interaction for each dialog step as well as the semantic information of the input (i.e. which action in the interface was triggered by speech, touch or point-and-click interaction). Those types of interaction can each bring along probabilistic distributions (e.g., speech recognition accuracy). The state s in terms of HCT is modelled by its respective bases. In other words, the belief state $b(s)$ of the probabilistic model contains the components of HCT. Namely, understandability, technical-competence, reliability, faith and personal attachment.

The system actions A, which are chosen by the policies depend on the current state of belief of the probabilistic model, are the dialog steps presented to the user. These are the different goals of explanations (justification, transparency, conceptualization, relevance and learning) as well as the task-oriented part of the dialog represented by a so-called *communicative function(c)*, with c from set C (e.g., question, inform, answer, offer, request, instruct) (see e.g. [12]). The transition probabilities and observation probabilities are represented by *conditional probability functions* (cpfs). They transfer the current state s into a new one (s'), according to the last action a and the observations o. Now, the aim is to define the cpfs in a way, that they together with the reward function $r(s,a)$ generate an optimal flow of the dialog. In order to maximise our reward, the bases of trust have to be intact and the costs of executing the actions (each action has a defined cost) should be kept low. For example, we defined that the understanding in s' will be high if a transparency explanation was the last system action m, the observations o were that the user clicked *OK* and viewing time was around his average time of viewing a dialog step like this before continuing. These cpfs are defined for all observations o and state components s.

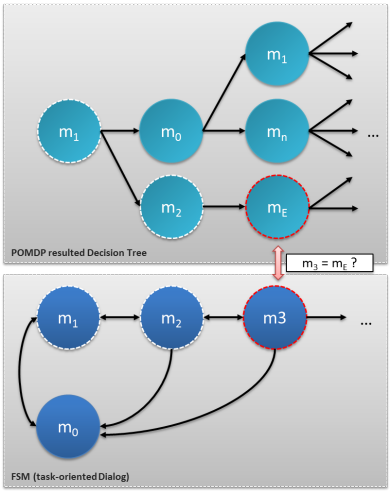

Figure 3. This figure shows the comparison of an FSM to the Decision Tree resulting from the POMDP. The next action m_3 in the FSM does not correspond to the one endorsed by the POMDP Decision Tree. Therefore, the dialog will be augmented by explanation action m_E.

Basically, conditional functions are defined using *if . . . else* for all wanted cases. The POMDP is defined in RDDL and then used by a planner [13] to search for an optimal policy π^*. This determines some kind of decision tree, to decide at each step which next action m' would be the best, dependent on the last action m, observations o and the previous belief state. This decision tree therefore represents some sort of guideline for the dialog flow.

B. Dialog Augmentation

The task-oriented dialog is modeled as a classic finite-state machine (FSM). Each dialog action has several interaction possibilities, each leading to another specified dialog action. Each of those dialog actions is represented as POMDP action m as part of C (*communicative function(c)*). At run-time, the next action in the FSM is compared to the one determined by the POMDP 3. This means that if the next action in the FSM is not the same as the one planned by the POMDP, the dialog flow is interrupted, and the ongoing dialog is augmented by the proposed explanation. For example, if the user is presented currently a communicative function of type *inform* and the decision tree recommends dependent on the current state of belief (here: understanding and reliability are both false) to provide a transparency explanation, the original next step in the FSM is postponed until after the explanation is first presented.

VI. Conclusion and Future Work

In this paper we showed the necessity to deal with critical situations in HCI using a probabilistic approach. The advantage of our approach is that the designer still can define a FSM-based task-oriented dialog. Usually most commercial systems are still based on such systems. However, expanding the dialog using a probabilistic decision model seems to be a valuable choice. Our experiment on the influence of explanations on HCT has clearly shown that it is worthwhile to augment the ongoing dialog by transparency or justification explanations to preserve an intact HCT relationship. In the future we will run experiments on how effective the hybrid FSM-POMDP approach is compared to classic as well as POMDP systems.

Acknowledgment

This work was supported by the Transregional Collaborative Research Centre SFB/TRR 62 “Companion-Technology for Cognitive Technical Systems” which is funded by the German Research Foundation (DFG).

References

[1] J. D. Lee and K. A. See, “Trust in automation: Designing for appropriate reliance.” *Human Factors*, vol. 46, no. 1, pp. 50–80, 2004.

[2] R. Parasuraman and V. Riley, “Humans and automation: Use, misuse, disuse, abuse,” *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 39, no. 2, pp. 230–253, June 1997.

[3] B. M. Muir, “Trust in automation: Part i. theoretical issues in the study of trust and human intervention in automated systems,” in *Ergonomics*, 1992, pp. 1905–1922.

[4] H. H. Clark and S. E. Brennan, “Grounding in communication,” in *Perspectives on Socially Shared Cognition*, L. Resnick, J. Levine, and S. Teasley, Eds. American Psychological Association, 1991, pp. 127–149.

[5] A. Glass, D. L. McGuinness, and M. Wolverton, “Toward establishing trust in adaptive agents,” in *IUI ’08: Proceedings of the 13th international conference on Intelligent user interfaces*. NY, USA: ACM, 2008, pp. 227–236.

[6] F. Sørmo and J. Cassens, “Explanation goals in case-based reasoning,” in *Proceedings of the 7th European Conference on Case-Based Reasoning*, 2004, pp. 165–174.

[7] M. Madsen and S. Gregor, “Measuring human-computer trust,” in *Proceedings of the 11 th Australasian Conference on Information Systems*, 2000, pp. 6–8.

[8] B. Y. Lim, A. K. Dey, and D. Avrahami, “Why and why not explanations improve the intelligibility of context-aware intelligent systems,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI ’09. NY, USA: ACM, 2009, pp. 2119–2128.

[9] R. C. Mayer, J. H. Davis, and F. D. Schoorman, “An Integrative Model of Organizational Trust,” *The Academy of Management Review*, vol. 20, no. 3, pp. 709–734, 1995.

[10] S. Sanner, “Relational dynamic influence diagram language (rddl): Language description,” 2010, http://users.cecs.anu.edu.au/ ssanner/IPPC2011/RDDL.pdf.

[11] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra, “Planning and acting in partially observable stochastic domains,” *Artificial Intelligence*, pp. 99–134, 1998.

[12] A. Stolcke, N. Coccaro, R. Bates, P. Taylor, C. Van Ess-Dykema, K. Ries, E. Shriberg, D. Jurafsky, R. Martin, and M. Meteer, “Dialogue act modeling for automatic tagging and recognition of conversational speech,” *Comput. Linguist.*, vol. 26, no. 3, pp. 339–373, Sep. 2000.

[13] F. Müller and S. Biundo, “Htn-style planning in relational pomdps using first-order fscs,” in *KI*, 2011, pp. 216–227.