# Finding Appropriate Interaction Strategies for Proactive Dialogue Systems—An Open Quest

**Florian Nothdurft, Stefan Ultes, Wolfgang Minker**
Institute of Communications Engineering
University of Ulm
Ulm, Germany
{florian.nothdurft,stefan.ultes,wolfgang.minker}@uni-ulm.de

## Abstract

In this paper we elucidate the challenges of proactiveness in dialogue systems and how these influence the effectiveness of turn-taking behaviour in multimodal as well as in unimodal dialogue systems. Effective turn-taking is essential for a natural and qualitatively high human-computer interaction. Especially in spoken dialogue systems, analysing whether the dialogue system should or could take the floor, seems to be an important process in the overall perceived quality of the interaction. Additionally, as technical systems get increasingly complex and evolve in the direction of intelligent assistants rather than simple problem solvers, proactive system behaviour may influence the perception of the ongoing dialogue between human and computer. Autonomously made decisions or triggered system actions may surprise or even disturb the user, which may result in a reduced transparency of the technical system. Therefore, the decision if, when and how to take the floor in a proactive system yields additional challenges. We discuss each layer of decision-making and explain how multimodal cognitive systems can help to control this decision-making in a valuable fashion.

## 1 Introduction

For spoken human-machine dialogues, the system decision of when to talk poses an important question. While this is usually an easy task for humans, a technical system is not yet able to analyse the complexity and nuances of a conversation. Hence, turn-taking strategies have been developed. State-of-the art interactive voice response (IVR) systems and spoken dialogue systems (SDS) usually use a predefined threshold to decide whether the user is willing to yield the floor. This simplistic approach leads to an unsatisfying and confusing user-experience, for example, because the user is interrupted by the system's re-prompting while thinking and trying to understand what the system expects (Ward et al., 2005; Raux et al., 2006). They also state that sometimes system time-outs are too long, leading to unusual and as awkward experienced waiting periods. Then both phenomena combine, this may lead to parallel attempts to take the floor. Hence, most recent research focuses on a more human-like approach to manage turn-taking behaviour in an SDS, for example by using automatically extractable features to inform efficient end-of-turn detection, and use this amongst other factors to train a turn-taking decision model based on decision theory (i.e., using statistical models), leading to significantly better results than fixed-threshold approaches (e.g., (Raux and Eskenazi, 2012)).

However, technical systems have evolved since the past decade from simple task-solving systems to technical companion systems (Honold et al., 2014) which solve tasks of increasing complexity cooperatively with the user. Hence, as the capabilities of such systems increase, it seems natural that technical systems will take over some of the responsibilities from the user and become an assistive system and life companion. To achieve this, these systems must also be able initiate interaction and not only react to the user. This will also lead to a more complex problem of turn-taking. While for conventional systems, only the question of when to take the floor is of interest, proactive agents also have to decide how to act

K. Jokinen and M. Vels. 2015. Proceedings of The 2nd European and the 5th Nordic Symposium on Multimodal Communication. 

and whether to act at all. Here, multimodal systems have a significant advantage over unimodal systems as they are able to exploit more cues about the interaction to make their decisions. This strategy also reflects human behaviour. It has notably been shown that human turn-taking not only depends on a various number of language cues but also on non-verbal cues like gesture or gaze (cf. (Duncan, 1972; Sacks et al., 1974; Gravano and Hirschberg, 2011)).

Hence, in this contribution, we describe and analyse the challenges of turn-taking for proactive agents in multimodal interaction and identify those key issues which have to be solved along the way to foster a healthy and sound human-computer interaction. In the next section we will elucidate the concept of proactive system behaviour followed by a description of our use-case at hand in Section 3. Section 4 will then discuss the resulting challenges for each layer of decision-making to give guidance for a future solution processes.

## 2 Proactive Behaviour

Proactivity in technical systems is an autonomous, anticipatory system-initiated behaviour, with the purpose to act in advance of a future situation, rather than only reacting to it. Therefore, for our research, we consider proactive behaviour as induced by implicit information and not by any kind of direct or explicit user interaction or user-made adaptation criteria. This means, for example, that user defined temperature values for a room, and the automatic adaptation to this preference when entering this room, do not count as proactive behaviour. Contrary to that the implicit sensing, e.g. by measuring body temperature using infrared sensors, that the user is feeling cold and the system's reaction to that by increasing the room temperature may be considered proactive. Respectively, the change of the user-interface modality to the user's characteristics may not be regarded as a proactive but an adaptive system behaviour. Therefore, only implicit reasons for proactive behaviour recognized by a cognitive system (Figure 1)—sensing a user's affective state for example—and the subsequent system actions may fulfil the requirements of proactive behaviour.

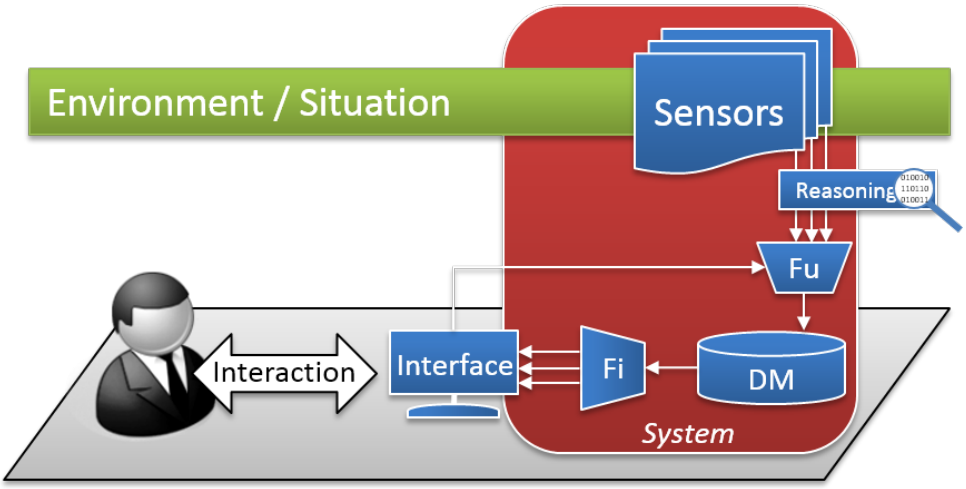


Figure 1: A typical architecture of a cognitive system. Only reasoned information coming from implicit interaction information (e.g., observations of the sensors) trigger proactive behaviour. *Fi* stands for Fission, which controls the modality arbitration. *DM* for Dialogue Management, which controls the flow of the dialogue and *Fu* for Fusion, which merges all input modalities to one consistent semantic representation.

## 3 Application

Proactive behaviour may occur in many different settings. In this work, we focus on proactive behaviour in a mixed-initiative system combining planning with dialogue. The research field of automated planning (e.g., (Biundo et al., 2011)) and scheduling deals with the development of methods and techniques to

automatically and autonomously create solutions, mostly action sequences, which will help a user or an autonomous system to achieve a predefined goal. The user proposes a goal to achieve and thereafter the system tries to come up with a solution. Such an autonomous process usually involves the risk of an unsatisfying or confusing user-experience. The user has no saying during the planning of the solution, and the proposed solution might not be the best in his mind.

Therefore, the application at hand rendering proactivity in dialogue systems is a cooperative planning system, which involves the user in the decision-making during the planning process (see Figure 2). Here, the interactive planning process is manifested in a fitness scenario. The user is guided through the process of selecting appropriate fitness exercises, to arrange an effective but also individual training plan. The automated planning will vary between four different variants: a fully-autonomous process, adding notifications to the user about the system's decisions, asking the user to confirm decisions, or leaving the decision completely to the user. For the latter, the users may decide about several options at times when the process is interrupted because of internal (e.g., planning heuristics) or external (e.g., affective user state) reasons. Hence, proactive behaviour is both the system-initiated integration of the user into the planning process due to planning heuristics and the proactive system reaction to implicit information like user behaviour observed by sensors. Therefore, this includes also proactive behaviour, which is triggered by the user's reaction to previous proactive behaviour. For example, proactive behaviour induced during planning may surprise the user and therefore lead again to proactive system behaviour.

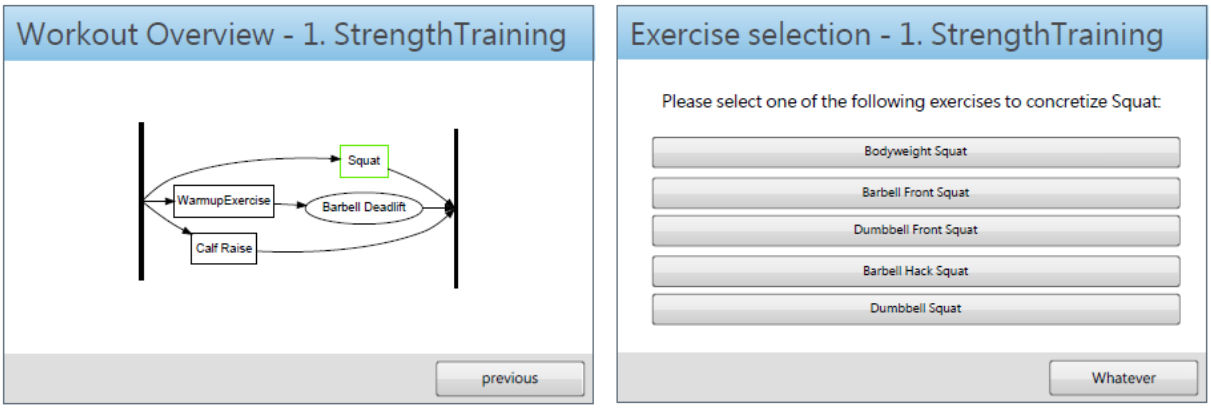


Figure 2: A screenshot of the interface of the prototypical mixed-initiative planning system, whose scenario is the interactive generation of an individual fitness training. The left image shows an overview of the current plan. The plan step *Squat* is still to be decomposed. In this case, the user can make the decision by selecting an appropriate refinement, the selection of a fitness exercise, shown on the right.

Taking a closer look at the user interaction in such a use-case, we encounter several questions regarding proactivity and turn-taking: *If* proactive behaviour is useful or necessary, *when* proactive behaviour should be integrated in the ongoing interaction (which is the one most related to classic turn-taking), and *how* this proactive behaviour should look like. In the next section, these questions will be discussed more thoroughly with regard to our application scenario.

## 4 Challenges for Proactive Systems

The three key questions for a proactive system during HCI are if, how, and when a proactive system behaviour is needed. Although those key questions will be discussed individually, we will see that they are nevertheless related to each other.

### If

Proactive behaviour is per definition anticipatory, with the idea to react to a future situation. Hence, proactive behaviour involves system actions apart from the expected task-oriented dialogue between human and computer. In our scenario this is either the user-integration into the planning process or

anticipatory system-actions dealing with an affective user state. Whether proactive behaviour is needed depends on several factors:

- How important is the proactive behaviour for the successful continuation of the dialogue, i.e., is it critical and required for short-term goals, but risks the cooperativity for interaction in the long run, or only beneficial in a longer perspective, to induce proactivity?
- Does the current user situation allow for additional system behaviour, e.g., additional system prompts?
- What is the classification probability for the cause of the proactive behaviour?

These main dimensions of whether proactive behaviour is adequate span the decision space depicted in Figure 3. If all three dimensions show significant values, proactive behaviour should be induced. It is

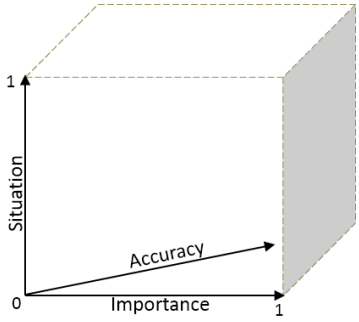


Figure 3: Decision Space: The *Situation* axis depicts if the external (e.g., environment) and internal user situations (i.e., user model) are adequate for proactive behaviour, the *Importance* axis whether the proactive behaviour addresses major or minor flaws in the interaction, and the *Accuracy* axis the recognition hypothesis classification results.

situation-adequate and triggered with a high probability based on a proactivity area within the decision space. Notwithstanding having the proactivity area usually originating with all axes at maximum value, its size and shape is highly dependent on the task at hand: while for non-critical tasks, the area may be quite big, critical tasks may have higher requirements to trigger proactive behaviour. Hence, a careful balance of all three dimensions is necessary.

Proactive behaviour itself, however, may have its own pitfalls. Apart from the usual reactive system behaviour where the reaction is anticipated by the user, autonomous decision-making by a proactive system involves the risk of creating incomprehensible and unexpected situations for the user. In the decision space, this maps to the dimension of *Situation*. Those situations usually occur due to incongruent models of the system: during interaction, the user builds a mental model of the system and its underlying processes determining system actions and output. If this perceived *mental model* and the actual system model do not match, the situation will be perceived as inconsistent - the user will not understand it.

In the present application scenario autonomous behaviour by the planner may lead to such situations. For example, the system's automatic preselection between a set of available options may cause user's confusion. The proactive system behaviour—adapting to the user's history of interaction—was not expected by the user, and might therefore be incomprehensible. These unexpected or incomprehensible situations have shown to reduce the user's trust in the system (Muir, 1992) and may ultimately result in reduced frequency or complexity of use (Nothdurft and Minker, 2014). The recognition of improper mental models appears not to be an easy task and requires the recognition of the "symptoms rather than the disease". This means that affective user states like *confusion* have to be recognized and compared to, e.g., the dialogue history to infer whether the mental models were incongruent.

The recognition of emotions and affective user states is one of the much studied research questions at the moment. Apart from basic emotion recognition, the most used affective user states to be recognized

via vision-based, audio-based, and audio-visual recognition are *interest*, *frustration*, *boredom*, and *confusion* (Zeng et al., 2009). In a meta-analysis on unimodal and multimodal affect detection, D'Mello and Kory (2012) stated that multimodal recognition accuracies yield performance improvements compared to unimodal affect recognition accuracies. However, in a naturalistic or semi natural (induced) context the improvements are minimal compared to classifiers trained on acted data. They found that contemporary affect detection mostly concentrates on bimodal or trimodal approaches. The most commonly used modalities are acoustic-prosodic cues and facial expressions (77% of all classifiers), followed by gestures, body movement and postures (30% of all classifiers). In general, recognition results based on non-acted data lead to accuracies ranging from 55% to 89%, with an average of 66%. Although there is promising work on this topic, spontaneous affective behaviour analysis in real settings, also commonly called "in the wild", still got a long way to go.

**How**

The next step in proactive system behaviour is the decision on how the system behaviour should be rendered, i.e., what kind of intervention is the most adequate. If we take a look at the prototypical application at hand, the interactive planner, several open questions arise. For example, even if the planning system decides that the user should be integrated in the next decision, still the question remains at which level the integration should be done (e.g., implicit vs. explicit, pruned vs. original). On the one hand an implicit confirmation of a system-preselected option may be possible where the user is only notified about the decision. On the other hand an explicit selection from a list of choices where the user's choice is unconstrained. For the former, the user also has the option to discard the system choice. Though these issues are related to proactive behaviour in our application, it is part of previous research. The most prominent work dealing with pruning (i.e., removing options) when presenting alternatives as lists was conducted by Sears and Shneiderman (Sears and Shneiderman, 1994). They stated that lists pruned to frequent selection options were faster and subjectively preferred to alphabetic lists.

In our work, the focus lies on how the systems' behaviour should be shaped when recognizing incomprehensible situations. As mentioned before, this may occur due to non-matching models. The user's mental model is a perceived representation of the reality, in this case of the system and it's underlying processes. However, the mental and the actual system model do not necessarily align, which may cause incomprehensibility. In (Nothdurft et al., 2014), we showed that incomprehensible proactive behaviour indeed will significantly reduce the user's perceived understandability and reliability of the system. This was done by training the user on a specific system and then confronting the participants with proactive, not yet experienced system behaviour, where the system did change the user's decision. In order to find out, *how* those situations should be handled by a technical system we took a closer look at human-human interaction. Here, misunderstandings or incomprehension are taken care of by providing explanations. In general, explanations are given to clarify, to change , or to impart knowledge. In these situations, the implicit idea consists of aligning the mental models and to establish a common ground between the participating parties.

Following that, we conclude that a technical system should attempt to clarify its actual model to the user in incomprehensible HCI situations. This means, that explanations should be given, to align the perceived mental model to the actual system model. However, there exists a variety of explanations which pursue different goals (Sørmo and Cassens, 2004):

**Conceptualisation** usually has the goal to address the user's declarative knowledge (e.g. describing things).

**Learning** addresses procedural knowledge in the sense, that for example tutorials are provided in order to learn how to do new things.

**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 advices or actions.

**Transparency** increases the user's understanding in 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. Thereby, the user may build a better mental model of the system and its underlying reasoning processes.

**Relevance** explains why the task at hand is relevant to the user. In contrast to the previous two goals that focus on the solution, relevance tries to justify the system-pursued strategy.

Though all of these goals are important, justification and transparency explanations are the most promising ones for incomprehensible situations in HCI. Therefore, we conducted a study testing whether those two explanation goals differ in their effects between each other and to providing no explanations as well. Our hypothesis was that though both explanation goals will help remedy negative effects, transparency explanations will be more helpful. Indeed, we could show that when providing transparency explanations in incomprehensible situations the *perceived understandability*, which measures the ability to build a correct system model using questionnaires, diminished on average only by 0.4 when providing *transparency* explanations (no explanation vs. transparency $t(34)=-3.557$ $p<0.001$), and on average by 0.5 with *justifications* (no explanation vs. justifications $t(36)=-2.023$ $p<0.045$), compared to 1.2 on a Likert scale with a range from 1 to 5 when providing no explanation at all (see (Nothdurft et al., 2014) for more details).

This showed that providing explanations can help to build a better model, or at least to maintain a model by reducing the impairment, and by that reducing the negative risks of incomprehensible situations. The first part of our hypothesis could be confirmed, whereas the second part is still unclear. Currently we are not yet perfectly sure whether the not-significant difference between transparency and justification explanations was due to improper explanation design or whether those two indeed do not differ in their effects. However, in our opinion the former is more likely, because the complexity of transparency explanations was reduced in our experiment. This means, that in other systems consisting of more complex system processes, the difference between justification and transparency explanations will increase in terms of understanding and building a coherent mental model.

Regarding to our application scenario at hand, this means that incomprehensible situations have to be addressed by providing explanations about the system processes leading to the current system behaviour. For example, the automatic preselection of an action by the system could be motivated using the dialogue history. For example, by providing the explanation that the proactive system behaviour (i.e., the preselection the options) results from recognized user preferences using previous episodes of interaction.

These experimental results show that it seems to be worthwhile to use explanations to cope with incomprehensible situations in HCI. For the decision on *how* proactive behaviour should be shaped, we can state that explaining system processes or providing justifications help to deal with incomprehensible situations. Even if we can decide whether and how the proactive intervention should be shaped, we still need to determine an adequate point of time in the ongoing HCI to provide the proactive behaviour.

### When

The problem of *when* to initiate proactive behaviour for HCI means that appropriate turn-taking points in the ongoing interaction need to be found. Those must guarantee sound and effective proactive behaviour. This issue is mostly related to the classic turn-taking problem, which deals with organizing and structuring the conversation by deciding on the system side whether or not to take the floor. Classic ideas in Spoken Dialogue Systems include using pause durations, discourse structure, semantics, or prosodic information and timing features to detect appropriate turn-taking points (Raux and Eskenazi, 2012). While recognizing turn-taking cues in human-human interaction via multimodal signals has been covered in recent research (see (Mondada, 2007) for an overview), the use of multiple modalities to control turn-taking in HCI is only recently emerging as a hot topic. For multimodal systems, this includes analysing the user's verbal and non-verbal signals (e.g., gaze, gestures, body movement) to generate and display well-timed and natural multimodal system behaviour, including feedback and turn-taking signals. While turn-taking itself is already a difficult problem, proactive behaviour includes even more challenges regarding appropriate turn-taking points. For instance, behaving proactively in a given situation might

even be so important that it has to be initiated despite inappropriate discourse structure or semantics. Therefore, this issue can be related to the *Importance* axis of the *Proactivity Space* shown in Figure 3. When proactive behaviour is of utmost interest, inappropriate turn-taking has to be tolerated.

## 5 Conclusion

Future dialogue systems will have to solve increasingly complex tasks cooperatively with the user. As the task complexity as well as the capabilities of such systems increase, it seems natural that these systems will take over some of the responsibilities and help the user achieve the task by proactive system behaviour. Though this might relieve the user by reducing work and cognitive load, it nevertheless involves the risk of incomprehensible HCI situations. In this paper, we elucidated the challenges of turn-taking in proactive system behaviour and how multimodal approaches can help with this issue in the three different decision making layers *if*, *how*, and *when*. The described Decision Space is constructed by the dimensions $Importance$, $Accuracy$ and $Situation$, which are the most important ones to decide *if* proactive behaviour is necessary. In terms of *how* to intervene, providing explanations to foster the building of correct mental models was described in detail. The most promising explanations to foster coherent mental and actual system models seem to be transparency explanations. $When$ to initiate proactive behaviour is mostly related to the classic turn-taking problem. Here recent statistical approaches did lead to a more human-like turn-taking and increased user-experience. However, conclusively we can state that finding appropriate turn-taking strategies for proactive dialogue systems is still an open quest, involving many challenging as well as interesting research questions.

## Acknowledgements

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

Susanne Biundo, Pascal Bercher, Thomas Geier, Felix Mller, and Bernd Schattenberg. 2011. Advanced user assistance based on ai planning. *Cognitive Systems Research*, 12(34):219 – 236. ¡ce:title¿Special Issue on Complex Cognition¡/ce:title¿.

Sidney D'Mello and Jacqueline Kory. 2012. Consistent but modest: a meta-analysis on unimodal and multimodal affect detection accuracies from 30 studies. In *Proceedings of the 14th ACM international conference on Multimodal interaction*, pages 31–38. ACM.

Starkey Duncan. 1972. Some signals and rules for taking speaking turns in conversations. *Journal of Personality and Social Psychology*, 23:283–292.

Agustín Gravano and Julia Hirschberg. 2011. Turn-taking cues in task-oriented dialogue. *Comput. Speech Lang.*, 25(3):601–634, July.

Frank Honold, Pascal Bercher, Felix Richter, Florian Nothdurft, Thomas Geier, Roland Barth, Thilo Hoernle, Felix Schüssel, Stephan Reuter, Matthias Rau, Gregor Bertrand, Bastian Seegebarth, Peter Kurzok, Bernd Schattenberg, Wolfgang Minker, Michael Weber, and Susanne Biundo. 2014. Companion-technology: Towards user- and situation-adaptive functionality of technical systems. In *10th International Conference on Intelligent Environments (IE 2014)*, pages 378–381. IEEE. SFB-TRR-62,Planning,KnowledgeModeling.

Lorenza Mondada. 2007. Multimodal resources for turn-taking pointing and the emergence of possible next speakers. *Discourse Studies*, 9(2):194–225.

B M Muir. 1992. Trust in automation: Part i. theoretical issues in the study of trust and human intervention in automated systems. In *Ergonomics*, pages 1905–1922.

Florian Nothdurft and Wolfgang Minker. 2014. Justification and transparency explanations in dialogue systems to maintain human-computer trust. In *Proceedings of the 4th International Workshop On Spoken Dialogue Systems (IWSDS)*. Springer, January.

Florian Nothdurft, Felix Richter, and Wolfgang Minker. 2014. Probabilistic human-computer trust handling. In *Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, pages 51–59, Philadelphia, PA, U.S.A., June. Association for Computational Linguistics.

Antoine Raux and Maxine Eskenazi. 2012. Optimizing the turn-taking behavior of task-oriented spoken dialog systems. *ACM Trans. Speech Lang. Process.*, 9(1):1:1–1:23, May.

Antoine Raux, Dan Bohus, Brian Langner, Alan W Black, and Maxine Eskenazi. 2006. Doing research on a deployed spoken dialogue system: One year of lets go! experience. In *in Proc. INTERSPEECH, 2006*, pages 65–68.

H. Sacks, E.A. Schegloff, and G. Jefferson. 1974. A simplest systematics for the organization of turn-taking for conversation. *Language*, 50(4, Part 1):696–735, December.

Andrew Sears and Ben Shneiderman. 1994. Split menus: effectively using selection frequency to organize menus. *ACM Transaction Computer-Human Interaction*, 1:27–51.

F. Sørmo and J. Cassens. 2004. Explanation goals in case-based reasoning. In *Proceedings of the 7th European Conference on Case-Based Reasoning*, pages 165–174.

Nigel G. Ward, Anais G. Rivera, Karen Ward, and David G. Novick. 2005. Root causes of lost time and user stress in a simple dialog system. In *INTERSPEECH*, pages 1565–1568. ISCA.

Zhihong Zeng, Maja Pantic, Glenn I Roisman, and Thomas S Huang. 2009. A survey of affect recognition methods: Audio, visual, and spontaneous expressions. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(1):39–58.