in: 5th CompanionAble Workshop March 8 2011, Brussels, Belgium

Adaptative input interpretation for dialogue management of an autonomous robot

Steffen Müller, Christof Schröter, Horst-Michael Gross*

Neuroinformatics and Cognitive Robotics Lab,
Ilmenau Technical University, Ilmenau, Germany
steffen.mueller@tu-ilmenau.de,
WWW home page: http://tu-ilmenau.de/neurob

Abstract. This paper is giving an overview on multimodal dialogue technology and highlights some specifics of human machine dialogue on an autonomous companion robot especially for elderly, cognitively impaired people, to be developed in the CompanionAble [1] project. The central aspect is multimodality of inputs and outputs and adaptation to the user, which is essential for a natural and intuitive interaction. After introducing a prototypical dialogue system and pointing out possible mechanisms for user adaptation, the paper focuses on the aspect of adaptive input fusion and interpretation for multimodal dialogue input. This technique allows for probabilistic integration of different uncertain input modalities and for automatic learning of semantics for previously unknown inputs, demonstrated in the exemplary integration of speech, GUI, and head gesture input.

Keywords: multimodal human-machine-dialogue, input interpretation, user adaptation, fusion

1 Introduction

Within the CompanionAble consortium, a mobile companion robot for elderly people with mild cognitive impairments (MCI) is developed, which aims to assist them in their daily life, in cooperation with a "smart home" installation. Because of the target group of elderly, special emphasis is put on a natural communication allowing an intuitive interaction. Therefore, the dialogue management system plays a central role for the acceptance and usability of the robot. Figure 1 shows a generic example of a dialog system for a mobile robot. One important aspect is the multimodality: multiple channels are used for communication between the interaction partners, that is, from the robot's point of view, for input and output. Input modalities can be e.g. speech, head or hand gestures, facial expressions, but also a graphical user interface (GUI). For the output, speech and other audio can also be used, as well as again graphical screen output or e.g. robot gestures or face mimics. The advantage of multimodal interaction is the more

* This work is supported by EU-FP7-ICT Grant #216487 to CompanionAble.

natural communication, as intuitive signals can be used in both directions, like in human-human interaction. On the other hand, interpretation of inputs is much more complex than in a purely GUI based or speech-based (e.g. phone information) system.

Besides support for natural communication channels, adaptivity is a main aspect of an intuitive system. An adaptive system should be able to adjust itself to the capabilities and preferences of the user, in order to improve the communication. Within the dialogue system, there are several subsystems which have the ability of user adaptation:

The input fusion system can adapt to the users preferences (prefered communication channel) as well as specifics (personal way of expression). For the interpretation of the inputs, semantic meanings must be infered from the user's communication signals. This requires models of relations between input and meaning, which can be generic, but can also be adapted to reflect the specific user. A low-level example is the acoustic model of a user for speech recognition. Furthermore, when preferences of the user regarding usage of certain communication channels are learned and taken into account, this can help resolve ambiguous or potentially contradictory perception.

In the dialogue manager itself, different dialogue strategies can be used. Besides the course of the dialogue, for an proactive behaviour the time and situation must be chosen in which to approch the user, which can also be adapted to his preferences.

Finally, the output part offers many possibilities of adaptation, including the choice of output channels, but also parameters like voice and tone of speech output or the layout of a graphical output.

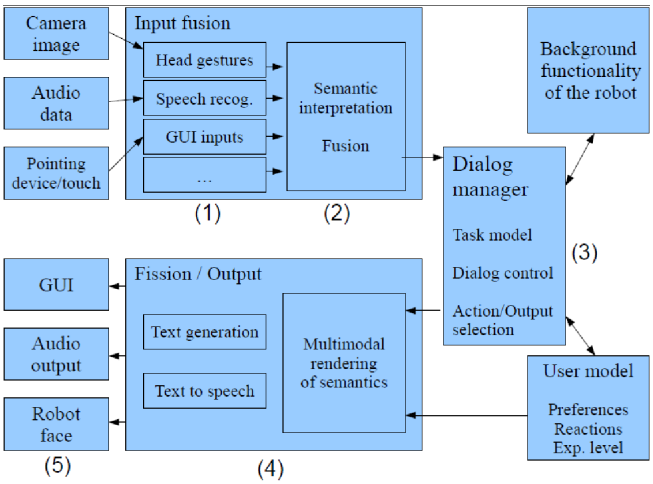

Fig. 1. Overview of a generic multimodal dialogue system for a companion robot.

2 The Companionable robot dialogue system

Here, a short overview of the dialogue system developed for Companionable robot shall be given: The development of the dialog management model we use was inspired from Speech Act Theory [12], the turn taking model of a sequence of communication acts resulting from analysis of human-human interaction. Topics of conversation are described by frames, e.g. navigation commands, reminder, greeting, and information request, which can also be hierarchically dependent. Each frames consists of a couple of communicative acts, more precisely user acts and machine acts, which describe the semantic classes of possible communication. In order to allow disambiguation of inputs, the history of the dialog is represented implicitly by means of expectation values for the communicative acts.

Whenever a question is asked by the system, each possible input act representing an answer is assigned an expectation. That allows disambiguation of inputs which convey a clear semantic only in the context of the dialogue history. E.g. a simple "yes" can be an answer for different questions and thus has potentially many different meanings for the system. By considering the expectation values of the respective input acts, it is possible to choose the correct interpretation. The data communicated in the dialogue is stored in slots, having a value and a reliability which results from the inference process as described below. The reliability also indicates whether an input has been confirmed already or not. The output generation in our system is controlled by a handcrafted grammar, handling each machine output act. Grammar rules can be activated or suppressed by means of conditions, which will consider the reliability of slots to decide whether to confirm assumptions implicitly or explicitly. Furthermore, different possible personalities are realizable due to the activation of other grammar parts, as used for adaptation to user preferences.

3 Adaptive input fusion and interpretation

To allow for a natural communication with the robot and to have some additional input modalities besides the error prone speech recognition, the CompanionAble dialogue system is multimodal, incorporating speech recognition, GUI inputs via touch screen as well as head gestures. In contrast to purely speech based systems, a fusion of the different inputs is necessary. First a short introduction to the multimodal input fusion for dialogue systems is presented here. Since inputs are processed in parallel, time sensitivity is crucial in order to decide whether to interpret commands in parallel (e.g. a command and a pointing gesture) or sequentially. There exist formal models [4], [5] describing the combination of inputs. The CASE model e.g. categorises the modalities into sequential or parallel and the fusion into independent or combined case. Thus it finds four different ways of modality combination. Only when the modalities are independent (e.g. voice command for "come here" and doing respective gesture simultaneously) and are used in parallel, a synergy effect occurs, helping to reduce ambiguity and gain robustness, in contrast to the combined case (e.g. command "go there"

and pointing pose giving the target) where the correct interpretation of the semantic is dependent on two systems, reducing the overall robustness.

Fusion of multimodal inputs as central element of multimodality is subdivided into data-level, feature-level and decision-level fusion [3], where data-level fusion refers to combination of different e.g. audio channels or camera images before features are extracted. Feature-level fusion combines different features before the semantic classification takes place, but it only helps when applied for closely coupled and well synchronised modalities (e.g. speech and mouth movements prior to of speech recognition). The alternative is fusion at decision level. Here the different input channels each have extracted their own interpretation of the semantics in the user's expression, which are combined in the dialogue manager. Lalanne et al. [7] give an overview of the development of fusion techniques for multimodal dialogues and conclude that classical fusion in contrast to machine learning based approaches is well understood nowadays. Jaimes [8] noticed that machine learning based fusion at decision level, on the other hand, is still in its infancy and needs further research. So our approach intends to apply statistical analysis on the cooccurrence of inputs from different modalities. A problem for machine learning approaches in general is the amount of necessary training datasets, not to be underrated. For applying training during the operational phase, it is essential that an initial reasonable fusion is possible without the training data. Without an initial function, the system would not be used by people and we never would get interaction data for adaptation, the system thus could not improve the understanding at all. This initial function can be defined heuristically.

3.1 Adaptive semantic mapping of inputs

The system developed in the CompanionAble project combines speech commands, GUI interaction and head gestures, but until now the semantic has to be assigned to the inputs by hand, coming along with an inflexible interface. Semantic interpretation is done by means of a parser and a grammar describing the possible inputs. E.g. head gesture recognizers are trained in before to recognize the semantic classes "yes" and "no". Once these classes are recognized, the event is sent to the dialogue system which applies a grammar that fills the answer slots with the respective semantic label. In parallel the speech recognition extracts the speech semantics also by means of the grammar and GUI offers to get the input by means of touch interaction.

The new approach presented here tries to learn the user specific semantics of inputs based on the ongoing interaction. Here we can benefit from the different reliabilities of the input modalities which initially only have a basic semantic labelling. E.g. the GUI is considered to be very reliable and thus has a very clear semantic labelling. The other modalities like speech and head gestures have initially unspecific semantics and are learnt when occurring in parallel to reliable inputs or when the dialog could confirm a semantic for them. So we have two cases when semantic labelling is adapted: An unsupervised case when inputs are occurring in parallel and a more or less supervised case when misunderstanding

took place and the user has given the correct semantic interpretation afterwards.

Example 1:

1. Robot asks a question to the user: "... please select yes or no!"
2. User pushes GUI button "Yes" while nodding the head.
3. Head gesture classification notices a new head gesture sequence, but does not know the meaning.
4. Since the GUI modality has recognized a semantic class "yes" very reliable, the new head gesture sequence could be labelled with that class. However, the reliability is weak, since it is not clear if the two things really correlate causally or only have a random cooccurrence.
5. When again a yes/no question is in the dialog context, the user might say "yes" and make the head gesture similar to the former one.
6. The speech recognizer may recognize a "yes" with a very poor scoring. The head gesture is also recognized, which indicates the same semantic class "yes". Therefore, the probability accumulates and a reliable "yes" is understood, despite the weak speech recognition input.
7. When the dialog continues and the "yes" is confirmed due to missing interventions of the user, the weak labelling of the gesture class can be reinforced.

Example 2:

1. User gives a speech command "Go to the entrance".
2. This is misunderstood by the speech recognition as "go you and dance".
3. The dialogue does not understand since that sentence is not in its repertoire. The user is asked to clarify his commands by either using another utterance or a different modality (e.g. GUI).
4. The user may select the option from the "GUI" (which is very reliable) or he says "drive to entrance", which could be understand correctly.
5. When a command has been executed and confirmed, the dialog manager can assign the semantics to the originally false recognition result ("go you and dance" maps to "drive to target entrance" with a given reliability).
6. When the user again says "go to the entrance" and the speech recognition again recognizes "go you and dance", then the correct meaning is known by the dialog manager.

3.2 Probabilistic input semantic modelling

The domain of home robotics is characterized by large inter-individual, cultural and situation depending variances in expression of gestures (head gestures), activities and different quality of speech command recognition. Therefore, the focus of our research lies on the online learning of user specific probabilistic mappings from modality specific detections to semantic inputs for the dialogue management system. For a system like the robot companion a limited set of functionalities exists and therefore, in communication a limited set of commands needs to be recognized. Each command consists of various semantic facts, one for the command itself and some parameters optionally. In the model for input fusion, the dialog manager's repertoire consists of a set of semantic classes, which each has a set of exclusive values. E.g. one semantic class is yes/no answers, another is navigation commands and a third is locations (see Figure 3). In one user turn (one message from the user to the system), each semantic class can be present or absent and if it is included it can only represent one value exclusively. On the other dimension, there are the different input modalities, whose outputs can be broken down to a list of discrete observation events. This is easiest for GUI inputs, as GUI interaction already is based on events in almost every implementation. The head gesture recognition also can emit an event each time a former seen sequence of head movements appears. In conventional gesture recognition, these sequences are immediately classified according to a semantic label, but here we can just use abstract labels which only serve to distinguish different gestures. The more difficult case is speech recognition because there are no events

generated that allow a proper association to a semantic. Of course, there is much effort put in semantic understanding of natural language in speech or written text, but all these approaches are based on a proper text string as input. Our idea is a more low level approach. Similar to a pet which can execute commands it is conditioned to, the robot also may notice the meaning of commands without deep semantic understanding of grammar and syntax of the text inputs, only based on associations. The problem with speech is that with a vocal expression of several words, multiple semantic classes could be communicated. On the other hand, a long sentence only transmits one semantic class sometimes. Consider e.g. the input "remind me again at 5pm": here three semantic facts are contained - create a reminder entry, the reminder time is 5pm, and the topic of the reminder should be the same as the last one ("again"). On the other hand a sentence like "Hector good bye, I am going out" only has one useful semantic, which is that the user will leave the house. The solution for that problem is a subdivision of the input sentence into words and word groups. Each of the word groups can learn its own semantic labelling and by means of multiple observations the labelling for longer phrases will get weak, while the meaning of short word groups will become dominant.

Since input modalities are considered independent, each input event is modelled separately. This results in a rather complex set of probability tables as can be seen in Figure 2: For each semantic class S_i and for each observation event O_m there is a binary probability $P(S_i, O_m)$ indicating the cooccurrence of the semantics i and the input events j. Furthermore, for the possible values of each semantic class there is a probability distribution $P(V_i, O_m)$ for cooccurrence of value V_i and observation event O_m. By means of these probabilities it is possible to estimate the meaning of certain observations for the dialog input. The probability model will replace the interpretation based on grammars, as implemented for the first prototype of the dialogue system. This kind of modelling allows a unified handling of all input modalities and makes the fusion system easily extensible to further modalities. In the remaining subsections we will describe how certain inputs can be interpreted using this model, and how the model can be adapted based on observed dialogue inputs.

3.3 Probability model based input interpretation

One of the problems with parallel processing of different inputs is the alignment of the inputs. Two or more events never appear at the same moment. Therefore, a continuous input state is estimated over time, aggregating the inputs of a time window (the user's turn). This input state is described by a probability for each semantic class to be intended by the user, called the belief on the semantic class' occurrence $bel(S_i)$. Additionally, for each class the belief $bel(V_i)$ models the state of the values of the semantic classes in the input so far. The update of these beliefs is done by means of Bayesian filtering. On the other side the dialog is in a certain state, where only a subset of semantic class combinations can be possible inputs. By means of these expectations and the estimated occurrence of semantics in the input, with each new input event, the system can decide

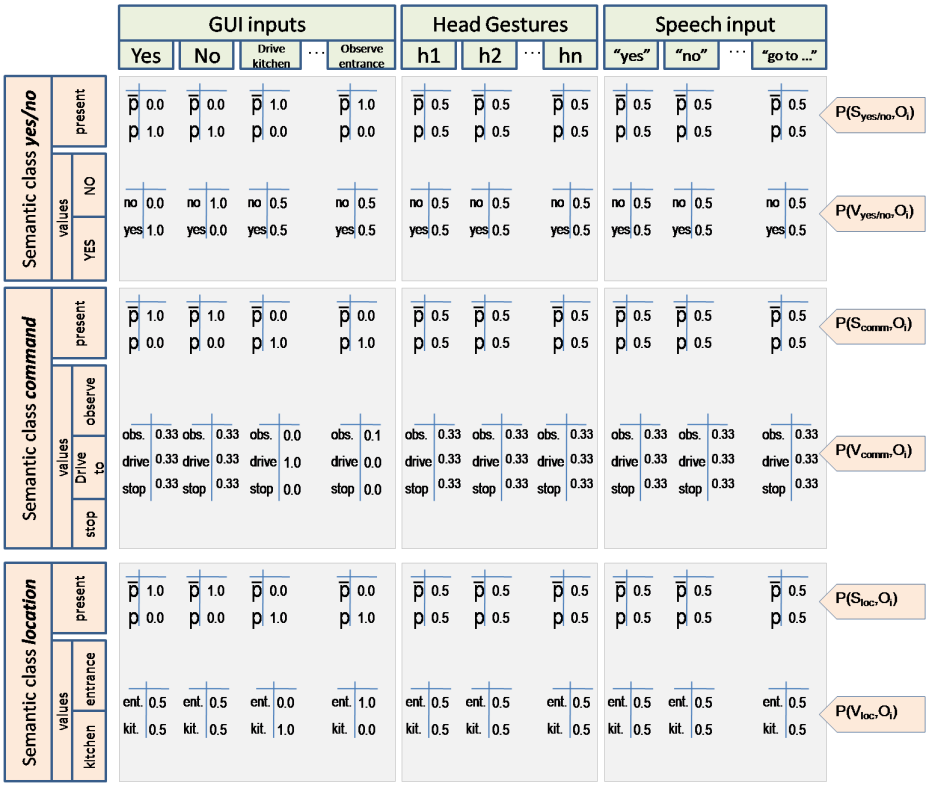

Fig. 2. Probabilistic model for input interpretation/fusion: only GUI provides information, other modalities are initially uniformely distributed and have to be learned in dialogue.

that either there is sufficient certainty about the input semantic or the system has to wait for further inputs. With each incoming input event, the $bel(S_i)$ and $bel(V_i)$ are propagated from the last event by means of a prediction model which reduces probability when not observed.

$$\widehat{bel}(S_i) = \frac{0.95\ bel(S_i)}{0.95\ bel(S_i) + bel(\overline{S_i})} \qquad \widehat{bel}(\overline{S_i}) = \frac{0.95\ bel(\overline{S_i})}{0.95\ bel(S_i) + bel(\overline{S_i})}$$

Similarly, the beliefs on the values are propagated:

$$\widehat{bel}(V_i) = \eta(0.1 + bel(V_i))$$

Where η is the normalization to realize a sum of one in the probability table. The aim of this prediction is to reduce certainty over time. If the time context gets longer, past observations should not be as relevant as the current ones. That way it is possible to correct inputs on the fly. Afterwards the predicted beliefs

are combined with the new observation, which should increase the probability on the possible semantic classes. Once the probability for one class S_i reaches a threshold, the semantic class is considered to be recognized and the associated dialogue action is triggered. After that the probability of that class is reset to zero because this input has been answered by the system. Other classes will stay unchanged and allow a faster recognition of these facts in further user turns.

$$bel(S_i) = \widehat{bel}(S_i)\frac{P(O_m = o_t, S_i)}{\sum_{O_m} P(O_m, S_i)} \qquad bel(V_i) = \widehat{bel}(V_i)\frac{P(O_m = o_t, V_i)}{\sum_{O_m} P(O_m, V_i)}$$

By means of the filtering process described above, the semantic fusion is performed implicitly. Figure 3 explains that in an example. Each time an event is received, the $bel(S_i)$ and $bel(V_i)$ are evaluated. E.g. "Go to the kitchen" triggers an update of $bel(S_i)$, but because the phrase is not known by the system ($P(O_{speech}, S_i)$ is uniform), the probability stays uniform. Then a GUI interaction is used to start the action and the system updates $bel(S_i)$ as well as $bel(V_i)$ again. Now the probability of semantic class "goto" and "target kitchen" exceeds the threshold causing the system to react. The robot might ask for a confirmation and now waits for a "yes" or "no". In the next user turn (blue bar) several head gestures (event 1 and 3) occur but none of them can raise the probability of "yes" or "no". The speech channel can raise the "yes", which is associated weakly with ok. The head gesture event 2 afterwards can finally raise the probability $bel(V_{yes/no} = "yes")$ and the action is executed.

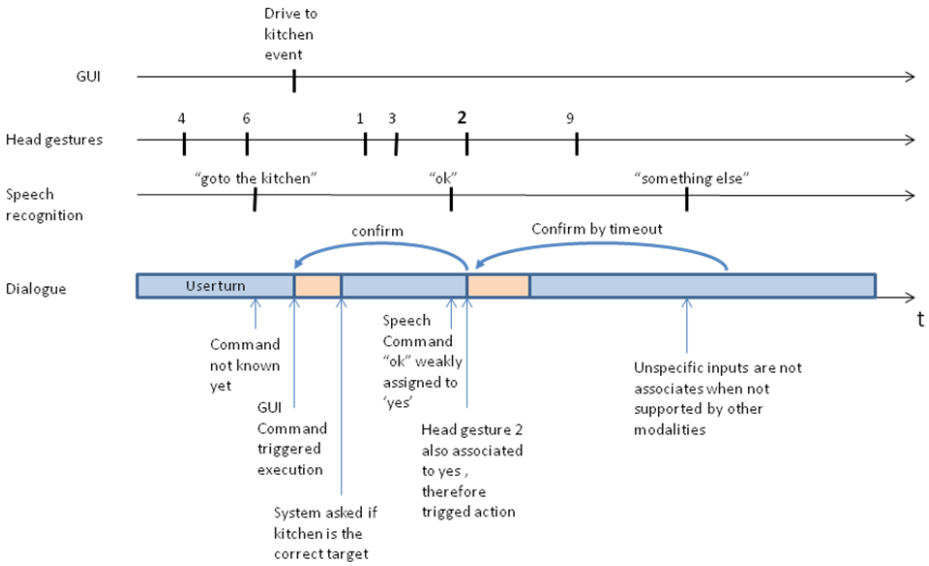

Fig. 3. Semantic fusion of input modalities GUI, head gestures, speech: In the top part events are plotted over time, while below the dialogue turns are visualized.

3.4 Probability model adaptation

When a user's turn is finished and a semantic class has been recognized, the events in the near past can be used for learning associations. This is based on the assumption, that all the signals from the user within the turn intend to communicate the same semantic meaning. However, in order to prevent false associations, the learning process cant be triggered before the semantic class has been confirmed to be correctly understood. For that the dialogue is used itself. If the next user turn did not correct the input, which is an implicit confirmation, or explicitly confirms the action upon request, we can assume the interpretation of the former user turn to be correct. Also in a case where no further inputs are expected and no intervention of the user takes place, the assumption is taken as confirmed when a timeout triggers. This can happen e.g. when a "drive to" command is given and the robot simply starts driving without any further dialogues. The actual update of the probability tables is done as follows: For each modality m in the association table $P(O_m, S_i)$, these values can be updated where $P(S_i)$ was above the threshold (all semantic classes being part of the reached interpretation). The closer the event was to the execution time, the more likely it is associated to the semantics. Thus in the update, the cell's value has to be increased depending on the time distance. For realizing a real Maximum a Posterior estimation (MAP) of the probability tables, where each observation has the same influence, the actual table will not be normalized to sum 1, but absolute frequencies of the cooccurences are counted instead. However, before the table is used for further computation, a temporary normalization needs to be performed, yielding a valid probability distribution.

$$P(O_m = o_t, S_i = s_t) = P(O_m = o_t, S_i = s_t) + e^{-\Delta t/\tau}$$

$$P(O_m = o_t, V_i = v_t) = P(O_m = o_t, V_i = v_t) + e^{-\Delta t/\tau}$$

In the example above the semantic mapping of "Goto the kitchen" as well as head gestures 4 and 6 will be adapted. For the head gestures this is straight forward, while the speech input first has to be separated into the word groups.

4 Conclusion

This paper has presented a novel approach for input fusion and interpretation in a multimodal dialogue system, which is based on probabilistic modelling of relations between input events and conveyed semantics. The presented method allows for aggregation of uncertain and weakly synchronized inputs and for learning of previously unknown inputs or reinforcement of uncertain knowledge of input semantics. Tests with simulated uncertain inputs have proven the validity of the approach, which will be implemented on a real robot to be tested in actual user interaction next.

References

1. EU FP7-funded project "CompanionAble": http://www.companionable.net
2. S. Oviatt, "Multimodal interactive maps: Designing for human performance," *Human-Computer Interaction*, vol. 12, no. 1, pp. 93–129, 1997.
3. H. Trung, "Multimodal dialogue management-state of the art," *Human Media Interaction Department, University of Twente*, 2006.
4. B. Dumas, D. Lalanne, and S. Oviatt, "Multimodal Interfaces: A Survey of Principles, Models and Frameworks," *Human Machine Interaction*, pp. 3–26, 2009.
5. L. Nigay and J. Coutaz, "A design space for multimodal systems: concurrent processing and data fusion," in *Proceedings of the INTERACT'93 and CHI'93 Conference on Human factors in computing systems*. ACM, 1993, p. 178.
6. P.R. Cohen, M. Johnston, D. McGee, S. Oviatt, J. Pittman, I. Smith, L. Chen, and J. Clow, "QuickSet: Multimodal interaction for distributed applications," in *Proceedings of the fifth ACM Int. Conf. on Multimedia*. ACM, 1997, pp. 31–40.
7. D. Lalanne, L. Nigay, et al., "Fusion engines for multimodal input: a survey," in *Proc. of the 2009 Int. Conf. on Multimodal interfaces*. ACM, 2009, pp. 153–160.
8. A. Jaimes and N. Sebe, "Multimodal human-computer interaction: A survey," *Computer Vision and Image Understanding*, vol. 108, no. 1-2, pp. 116–134, 2007.
9. S. Oviatt, R. Coulston, S. Tomko, B. Xiao, R. Lunsford, M. Wesson, and L. Carmichael, "Toward a theory of organized multimodal integration patterns during human-computer interaction," in *Proceedings of the 5th Int. Conf. on Multimodal interfaces*. ACM, 2003, p. 51.
10. E. Reiter and R. Dale, "Building applied natural language generation systems," *Natural Language Engineering*, vol. 3, no. 1, pp. 57–87, 1997.
11. O. Lemon, "Adaptive natural language generation in dialogue using Reinforcement Learning," *Proceedings of SEMdial*, 2008.
12. W.P. Alston, *Illocutionary acts and sentence meaning*, Cornell Univ Pr, 2000.
13. A. Jameson, "Adaptive interfaces and agents," *Human-Computer Interaction: Design Issues, Solutions, and Applications*, p. 105, 2009.
14. V. Rieser and O. Lemon, "Learning effective multimodal dialogue strategies from Wizard-of-Oz data: Bootstrapping and evaluation," in *Proceedings of ACL*, 2008.
15. M. Spitters, M. de Boni, J. Zavrel, and R. Bonnema, "Learning effective and engaging strategies for advice-giving human-machine dialogue," *Natural Language Engineering*, vol. 15, no. 03, pp. 355–378, 2008.