Exploring Human Error Handling Strategies: Implications for Spoken Dialogue Systems

Gabriel Skantze

Centre for Speech Technology
KTH, Sweden
gabriel@speech.kth.se

Abstract

In this study, the user experience and the consequences of different error handling strategies for spoken dialogue are examined. A modification of the Wizard of Oz method is used, where a speech recogniser is included in the setting. This makes it possible to study how humans handle speech recognition errors before a dialogue system is actually built. The results show that wizards tend not to signal non-understanding when they face speech recognition problems, but instead ask task-related questions to confirm the wizard's hypothesis about the situation, rather than what has been said. This strategy leads to better understanding of subsequent utterances, whereas signalling non-understanding leads to decreased user experience of task success.

1. Introduction

An important bottleneck for many dialogue systems built today is the speech recogniser, which will inevitably introduce miscommunication in the human-computer dialogue. These kinds of error should be prevented, detected and handled by the dialogue system in a way that maximizes the user's satisfaction of using the system. In order to do this appropriately, data on human-computer dialogue must be collected along with the user's rating of her experience of using the system.

A common method for collecting such data before implementing an actual system has been the Wizard of Oz method, where an operator is simulating parts of the system, most often assuming a perfect speech recogniser. The problem with this method is that the collected corpus will not contain any data on how the speakers handle speech recognition errors.

In the experiment presented in this paper, a modified Wizard of Oz method was used. In order to expose typical speech recognition errors, the user spoke through a speech recogniser, and the wizard (henceforth referred to as operator) could read the results, but not hear the utterance. In this way, human error handling strategies could be explored. To get more varied data on different strategies, operators who were not experienced in designing dialogue systems and did not have an understanding of how errors traditionally are handled in dialogue systems were selected. For the same reason, the operators were allowed to speak freely and the users were openly informed that they interacted with a human operator, unlike the traditional Wizard of Oz setting.

Before motivating the modifications to the paradigm, the problem of error handling in spoken dialogue systems and the research questions for this study will be reviewed and discussed.

2. Background

2.1. Error handling in spoken dialogue systems

Miscommunication is often divided into misunderstanding and non-understanding [1]. *Misunderstanding* means that one participant obtains an interpretation that she believes is correct, but is not in line with the other speaker's intentions. If the addressee fails to obtain any interpretation at all, or obtains more than one interpretation, with no way to choose among them, a *non-understanding* has occurred. One important difference between non-understandings and misunderstandings is that non-understandings are recognized immediately by the addressee, while misunderstandings may not be identified until a later stage in the dialogue. Some misunderstandings might never be detected at all.

Given a non-understanding, the addressee must decide how to react. One option, often used in spoken dialogue system, is to signal this to the speaker of the non-understood utterance, by just saying "sorry, I didn't understand", or by a request for repeat ("please repeat"), which can result in a very tedious dialogue if there are a lot of errors. Despite the method's frequent use in public services, it is a widely held assumption among many dialogue designers that this signal of non-understanding should be avoided (cf. [2]). One question for this study is how humans will react when faced with the problem of non-understanding.

In a dialogue system, the problem is not just how to react to non-understandings, but also to detect (or decide) that a non-understanding has actually occurred. A robust parser should be able to return partial results with confidence scores. Given a noisy result from the parser, the system must decide if it should make an interpretation and thereby risk a misunderstanding or decide upon a non-understanding. How well human subjects are capable of such error detection is also a question for this study.

If the system decides upon an interpretation (and not upon a non-understanding), it should somehow signal how the utterance was interpreted. This feedback of understanding is called grounding [3]. The grounding can be more or less explicit, ranging from explicit confirmation to just an acknowledgement or a relevant next contribution. The choice of grounding strategy should depend on factors such as confidence of the interpretation, cost of task failure and the user's need for feedback in the specific situation. Given the system's response to the initial user utterance, the user will somehow react to it. If the user is satisfied with the interpretation, she will either confirm the interpretation or just continue with the topic at hand. If the interpretation was inadequate, a misunderstanding has occurred and the user will probably (but not always) signal this. This reaction needs to be analysed in order to detect that the original utterance was misunderstood. This (late) detection of an error that occurred several turns ago should not be confused with the (early) detection of a non-understanding.

Two important aspects of different error handling strategies, which do not have to correlate, are how they are experienced and how efficient they are in resolving the problem. The PARADISE evaluation framework for dialogue systems offers a method for studying the interaction between these factors [4]. From a user-centred point of view, the experience of using the system should come first, and efficiency should be a means for improving the experience of using the system. To examine the user's experience of error handling is therefore important in the current study.

2.2. Using the Wizard of Oz method for studying error handling

In order to design dialogue systems that can handle the varieties of situations that occur in human-computer dialogue – such as error handling – data of such interaction should be collected. To do this, the Wizard of Oz method is traditionally used [5]. One problem is that it is hard to get an accurate account of what happens when speech recognition errors occur, since they are often ignored when the experiment is conducted. The optimistic assumption has been that these things could be added later on when the rest is solved. Others have tried to simulate errors, such as randomly substituting words in the input [5]. This is problematic for two reasons. First, the wizard is working under time pressure and it may be hard to do the right substitutions while controlling all other components. Second, the kind of errors that really do occur may be hard to simulate. Just substituting a word may be too simplistic a model. Often, two words can for example be substituted by three other words. Out-of-vocabulary words will often give rise to totally unexpected results, as the speech recogniser is trying to fit what has been said into the language model using in-vocabulary words.

The method can also be very costly. In order to deceive the subject, the wizard must work very fast and accurately. This does not only require a good design of the experimental setting and operator environment, but also much training of the operator. Another problem is that once a subject has been used in one Wizard of Oz study, she will be very suspicious when participating in similar studies. This makes it hard to reuse subjects, but also to do longitudinal studies, since it may be problematic or unethical to withhold the truth about the setting for a longer time.

It has also been shown that the behaviour of the dialogue system will have great impact on the user's behaviour (cf. [6]). Thus, how the wizard is supposed to act will influence the data that is collected. This can be a problem, since the collected data might be based on a priori assumptions about the user's behaviour and might not cover possible interaction patterns that were not anticipated.

3. Modifying the Wizard of Oz method

In this study, a speech recogniser was introduced in the Wizard of Oz setting, in order to get data on how the user and operator might react to speech recognition errors. The operator could not hear what the user said, but instead read the recognition result from the screen. The constraint that the user has to believe that she is talking to a computer was released. Instead, different naive operators were used, in order to get more varied data on error handling strategies. The goal of the study was not to test a specific system design, but instead to find suitable error handling strategies. Thus, the operator was treated as a subject as well. Using a speech recogniser in an ordinary Wizard of Oz setting, where the user is supposed to believe that she is talking to a computer, may be problematic, since it can be very hard to prescribe how the operator should behave depending on different levels of comprehensibility.

The assumption that the user must believe that she is interacting with a computer is common in Wizard of Oz studies (cf. [5]). The widespread belief is that if the user would believe that she was speaking to a human, her linguistic behaviour would be different. However, studies have shown that this assumption could be questioned. In an experiment conducted by Amalberti et al. [7], the effects of the user's conceptions about the other speaker were tested. Two groups of subjects were asked to obtain information about air travel via dialogue with a remote travel agent. One group was told that they were talking to a computer, while the other one was told that they were talking to a human operator. In both cases, the voice of the operator was distorted. The amount of distortion was carefully tuned, so that the human group could be told that they were testing communication through a noisy channel, while the other group believed that they were talking to a computer (as they were told). Thus, the experimental setting was exactly the same for the two groups, apart from their conceptions about the other speaker. The results showed that there were differences in the users' linguistic behaviour, but mostly in the beginning – a lot of differences tended to disappear after subsequent trials. The difference that could be found between the groups mainly concerned problem solving, where the users in the human group cooperated more with the operator. This suggests that the experience that the user has with the system also affects her way of interacting with it. It may be the case that if users are faced with more cooperative systems, they may start to take advantage of this. In order to advance in the development of dialogue systems, it could be dangerous to adapt to the users' current beliefs of the capabilities of such systems, especially to users that have very limited experience of such systems.

Using automatic speech recognition in this setting gives further reasons for questioning the dependency on the user's conceptions about the other speaker. It may be the case that the user's specific linguistic behaviour in human-computer dialogue is more dependent on the limited understanding of the other speaker, than whether it is a human or a computer. In the current study, the user was told about the speech recognition and was therefore aware of the fact that complex utterances might not get through.

One thing that does differ in the conversation with humans and machines, even if there is a noisy channel in both cases, is the amount of common ground that we have before engaging in the conversation [3]. To make this difference as small as possible, the operator and the user should not be able to see or get to know each other before or during the experiment, so that they will not form any assumptions about each other, and have as little common ground as possible. However, both subjects should be fully informed about the experimental setting. This puts some constraints on how the operator should reply. One possibility could be to let her type a message, synthesize it and play it back to the user, using a text to speech system. However, pilot studies showed that this would be too slow, and that the operator might behave in a "lazy" way, not typing the whole message that she actually wanted to send. Another solution could be to let the operator choose or compose the answer from a set of templates. The problem with that approach is that it would restrict the operator's output, and unexpected behaviour may not be captured. The proposed solution is instead to distort the operator's speech through a vocoder, and let her speak freely. The final setting is illustrated in Figure 1:

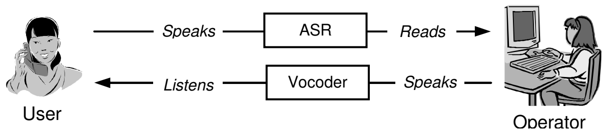


Figure 1: The setting used in the experiment.

It should be noted that this experimental setting lacks the control that the consistent behaviour of a trained operator would give. This method may be good for explorative studies, which aim for new ideas on dialogue behaviour, and especially on how error situations could be handled. It should be followed up by more controlled experiments, in order to test the derived hypotheses.

4. The domain for the experiment

A general distinction can be made between problem solving and information seeking dialogue. Most dialogue systems built today are designed for information seeking, such as travel information and stock quotes. The domain used in this experiment was not about information seeking, but direction-giving, which should be classified as problem solving. In such a system, the user has the goal to get to a specific location and uses the dialogue system (in this experiment simulated by an operator) to get route directions. The system does not know where the user is, so it must rely on the user's descriptions of the environment. One important difference from information seeking is that the dialogue system can establish the user's goal at an early stage in the dialogue and then work towards this goal. This is harder in information seeking (especially information browsing), since the system rarely knows the user's final goal of using the system. The type of dialogue can affect which types of error handling strategies that might be used by the speakers, which should be kept in mind when analysing the results.

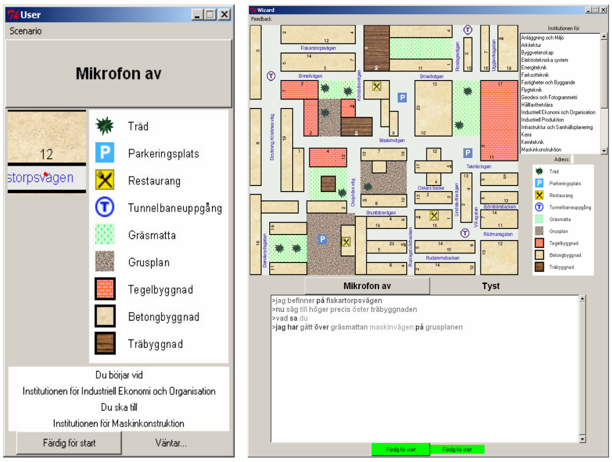
Dialogue about route descriptions have been studied extensively in so-called Map Task experiments [8]. The question is to what extent these data are applicable to dialogue systems for pedestrian navigation, since the "user" (called the "follower" in the Map Task) has access to the whole map and can talk about absolute directions (such as "north", "south", "up" and "down"). For this experiment, a simulation environment was built to prevent the user from using such information. This is described in the next section. To the author's knowledge, Map Task experiments have not been conducted using speech recognition previously.

5. Method

5.1. Experimental design

16 subjects were used, 8 users and 8 operators. All subjects were native speakers of Swedish. The subjects were paired in groups of operator/user. There were 8 women and 8 men, equally balanced as operators and users. Users were chosen to have low computer experience (to represent ordinary users), while the operators were chosen to have a somewhat higher computer experience and some experience of speech technology (but only limited experience of dialogue system design). This was assumed to make the learning of the operator interface faster.

The subjects were not allowed to see each other until the experiment was over, and were not given any information about each other before the experiment. Each subject was informed about the experiment and the setting, and the interfaces were explained to them. The user's and the operator's interfaces are shown in Figure 2.



Du börjar vid
Institutionen för Industriell Ekonomi och Organisation
Du ska till
Institutionen för Maskinkonstruktion

Anläggning och Miljö
Arkitektur
Byggvetenskap
Elektrotekniska system
Energiteknik
Farkostteknik
Fastigheter och Byggande
Flygteknik
Geodesi och Fotogrammetri
Hållfasthetslära
Industriell Ekonomi och Organisation
Industriell Produktion
Infrastruktur och Samhällsplanering
Kemi
Kemiteknik
Maskinkonstruktion

```
>jag befinner på fiskartorpsvägen
>nu säg till höger precis öster träbyggnaden
>vad sa du
>jag har gått över gräsmattan maskinvägen på grusplanen
```

Figure 2: The user's interface to the left and the operator's interface to the right.

At the bottom of the user's screen, a scenario was presented. According to the scenario, she was supposed to take herself from one department to another on a simulated campus, where only a small fraction of the map surrounding the current position was shown (seen from above). When the user changed direction, the whole map rotated, so that the user always was facing "up". This made it impossible for the subjects to talk about "up", "down", "north" or "south". Instead, they had to use landmarks and relative directions. In order to solve the task, the user had to tell the operator at which department she was and where she wanted to go.

The operator was given the task to guide the user, using a map showing the whole campus. The maps were identical, except for some street names that were missing on the user's map. The operator could easily look up where the departments were located. The user's position was not shown on the operator's map, so the operator had to rely on the user's descriptions of the environment. On each screen, there was a legend explaining the landmarks.

In order to get more concentrated utterances from the user, a push-to-talk mechanism was used. The speech was recognised by a speech recogniser and the recognised string was displayed on the operator's screen. An off-the-shelf speech recogniser was used with built-in acoustic models of Swedish. Tri-gram stochastic grammars were used, trained on a small corpus of invented dialogues and transcriptions from pilot studies with a vocabulary of about 350 words.

In order to make the confidence scores readable and let the operator easily get an overall understanding, the words were coloured in greyscale according to each word's confidence. Words that were coloured in darker tones had higher confidence scores, while lighter tones reflected lower confidence scores. Since the operator could not hear the user, an indicator on the screen showed when the user was speaking or not, in order to facilitate turn taking. The operator also had to push a button in order to say something to the user. However, the operator's speech was played back directly to the user, processed through a vocoder. The processed speech was fairly easy to understand, according to post-surveys. However, a lot of intonation was distorted, and the subjects could not hear whether it was a male or female voice.

Five scenarios were given to each pair of subjects, which resulted in 40 dialogues. The task was interrupted if they didn't complete it within ten minutes. After each scenario, both the operator and the user had to fill out a questionnaire about the interaction. The questionnaire consisted of a number of statements with which the subjects had to judge how much they agreed. They could choose from seven steps, ranging from "strongly disagree" to "strongly agree". After the whole experiment, both the user and the operator were interviewed.

5.2. Data analysis

The users' and operators' utterances were transcribed and manually annotated by one annotator for the different features that were to be analysed.

Each user utterance was annotated based on how well it was understood by the operator. To estimate this, the speech recognition result and the operator's reaction to the utterance were considered. Of course, some cases were ambiguous, but most often this was fairly easy to determine. The degree of understanding was classified into four categories:

Full understanding — The full intention of the utterance was recognised.

Partial understanding — Only a fragment or a part of the full intention was recognised.

Non-understanding — No part or fragment of the intention was recognised. Single words were possibly understood.

Misunderstanding — The listener believed in a partial or full interpretation that was not in line with the speaker's intention.

Each utterance was also classified based on the dialogue acts that were intended. The common procedure in this kind of annotation is to let several persons annotate and then measure the degree of agreement. This has not been done in this study. However, the categories were selected specifically for the annotated corpus and there were few ambiguous utterances which were hard to classify. The most common types were:

- Operator describes/user requests route
- User describes/operator requests the current position
- User states/operator requests goal
- Request/statement of how the task is proceeding
- Greeting/Farewell/Thanks
- Acknowledgement
- Signal of non-understanding (including requests for repetition) and understanding

6. Results

6.1. General results

80% of the scenarios were solved within ten minutes. An example of a successful dialogue fragment is shown in Figure 3.

User:	I am at the department of industrial economy and organization
Operator:	Ok, where do you want to go?
User:	I want to go to the department of machine construction
Operator:	Ok, now I know where it is. If you walk on the street you are now, you can see that there is a number twelve
User:	Ok
Operator:	Walk past that until you have a wooden house in front of you
User:	I am standing in front of the wooden building
Operator:	Then you should take right
User:	Arrived at a lawn

Figure 3: An example dialogue translated from Swedish.

In this dialogue fragment, there was a 0% word error rate. However, this dialogue was not typical; there were a lot of errors in the recognition results, about 42% word error rate. This was partly due to the users' relatively free speech and partly due to the limited training of the language models (7,3% out-of-vocabulary per utterance). However, very few of the utterances resulted in misunderstandings. Instead, the operators were very good at deciding when a recognition result or parts of a result should be rejected (a non-understanding). The distribution of the different types of understanding is shown in Figure 4.

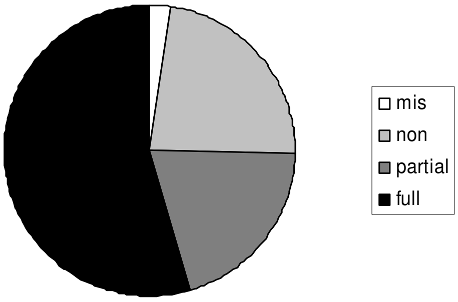


Figure 4: The distribution of the operators' understanding of the users' utterances.

Despite of the numerous non-understandings, post interviews revealed that the users in general experienced that they were almost always understood. One possible reason for this was that the operators most often did not explicitly signal non-understanding, unlike the behaviour of most dialogue systems. Instead, they often asked a task-related question to the user or continued with the route direction, using the context and possibly some part of the non-understood recognition result that seemed correct.

6.2. Strategies after non-understanding

To investigate the effect of the operators' reactions to non-understandings, the operator dialogue acts following a non-understanding were classified into three distinguishable (approximately equally distributed) groups: continuation of route description, signal of non-understanding, and task-related questions (about current position) to the user. Examples of these strategies are shown in Table 1.

Table 1: The different operator strategies after non-understandings. All examples translated from Swedish. Spoken utterances in italics, the output from the ASR in bold:

Strategy 1: Continuation of route description	
O:	Continue a little bit forward.
U:	**street that there house** *(Past the wooden house?)*
O:	Now, walk around the wooden house. Take left and then right.
Strategy 2: Signal of non-understanding	
U:	**west with** *(That's right.)*
O:	Please repeat what you said.
U:	**that there with** *(That's right.)*
Strategy 3: Task-related question about position	
O:	Do you see a wooden house in front of you?
U:	**yes crossing address now** *(I pass the wooden house now.)*
O:	Can you see a restaurant sign?

Notice that the operator's reaction in strategy 1 and 3 doesn't reveal that the intention of the user's utterance wasn't in fact understood. However, a few words may be understood, which is possibly forming a hypothesis about the user's position. This is confirmed, not by requesting or verifying what the user actually said, but by signalling the operator's hypothesis about the user's geographical position, either through a route description or by a task-related question.

Miscommunication can often lead to error spirals, where the user just repeats the non-understood utterance or starts to hyper articulate (cf. [9]). A good error recovery strategy should therefore aim at coming to understanding as quickly as possible after a non-understanding has occurred. The distribution of the operator's understanding of the user utterance following the different strategies was therefore counted and compared to the expected general distribution of understanding. The result is shown in Figure 5. The expected distribution, shown in the rightmost bar, is the same as in Figure 4, i.e. the general distribution for all user utterances.

There was no deviation from the expected distribution after strategy 1 and 2, but after strategy 3 there were significantly less non-understandings, and an increased number of partial understandings (X^2=18.45; dF=3; p<0.001; goodness-of-fit test). This shows that strategy 3 gives a faster recovery from the problem.

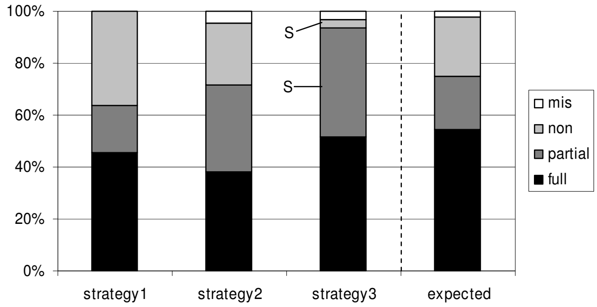


Figure 5: The understanding of the user's utterance that follows the operator's reaction to a non-understanding. "S" marks significant deviation from the expected value.

6.3. User experience of task success

To investigate how different factors contributed to the user's experience, a multiple regression analysis was used in a way similar to the PARADISE evaluation framework for dialogue systems [4]. The input to the regression analysis is an independent factor (in this case the user experience) and a set of dependent factors. The output is a set of coefficients for the dependent factors that describe the relative contribution of each factor for the variation in the independent factor. Unlike PARADISE, cost and task success were not separated and normalised, but just treated as factors in the analysis. Since the user's task was given beforehand and was quite artificial, it was hard to get a measure of the "user satisfaction". Instead, the user's experience of task success was used. The question "how well do you think that you did in solving the task?" from the questionnaire was used as the dependent factor, which was a rating from 0 to 6. The independent factors were: time to solve the task, the length of the path that the user went, the mean word error rate, the number of non-understandings, and the number of uses of strategy 1, 2 and 3. The regression analysis resulted in a significant correlation between the dependent factor and two of the independent factors (R^2=0.56; p < 0.0001). The contribution of the different factors to the user's experience of task success is shown in Table 2.

Table 2: Results from the regression analysis.

Contributing factors	Coeff	SE	T Stat	P-value
Task completion time	-0,456	0,083	-5,499	< 0,000
Strategy 2	-0,560	0,262	-2,142	0,039
Non-contributing factors				
Total path				
Word error rate				
Non-understanding				
Strategy 1				
Strategy 3				

As can be seen in the table, the only factors that were contributing were time for task completion and the number of non-understandings that the operator had *signalled* (which both had a negative effect). It is interesting to note that the number of non-understandings per se had no effect on the user's experience, but only the cases where the user was made aware of the non-understanding.

7. Conclusions and Discussion

In the experiments, the high word error rate caused only a few misunderstandings, but many non-understandings. This suggests that different knowledge sources (such as confidence score, syntactic structure and context) can be used (at least by humans) for early detection of errors in the recognition result, and for deciding upon appropriate reactions to them. Despite the numerous non-understandings, users reported that they were almost always understood. Unlike most dialogue systems, the operators did not often signal non-understanding. If they did display non-understanding, this had a negative effect on the user's experience of task success. Non-understandings per se had no such effect.

An alternative reaction to signalling non-understanding was to instead ask task-related questions that were confirming the operator's hypothesis about the user's position, but not what the user actually said. This strategy led to fewer non-understandings of the subsequent user utterance, and thus to a faster recovery from the problem.

One question is whether the decreased user experience of task success was directly caused by the signal of non-understanding (which can be frustrating), or indirectly by the non-understanding of subsequent utterances. However, like strategy 2, strategy 1 did not lead to decreased non-understanding, but unlike strategy 2, it did not lead to decreased experience of task success. This suggests that it is the signalling of non-understanding per se that is frustrating and gives the user an experience of task failure. This shows that efficiency might not be the sole predictor for the user's experience of task success.

When designing graphical human-computer interfaces, a general guideline has been to always provide feedback on how the user's actions are "perceived" by the system (cf. [10]). The current study shows that such a principle may not hold for speech interfaces, at least not for systems in which a more natural conversational metaphor is adopted.

The results suggest that when non-understandings occur in spoken dialogue systems, a good domain model and robust parsing techniques should be used to pose relevant questions to the user (instead of signalling non-understanding), so that errors can be efficiently resolved without the user experiencing the dialogue as problematic and dominated by error handling. One obvious question is whether these error handling strategies are possible to implement in a dialogue system, since they require robust parsing techniques, good error detection capabilities, and a world model that facilitates the construction of relevant task-related questions. A new dialogue system for pedestrian navigation is under development at the Centre for Speech Technology, KTH, where we will try to incorporate these techniques. It will be interesting to see whether the users will behave differently or have different experiences of error handling when they are talking to a computer. Another interesting question is whether this kind of error handling can be used in information-seeking domains as well, when there is no clear goal of the dialogue. The task-related questions after non-understandings may require that the system has such a goal.

8. Acknowledgements

This research was carried out at the Centre for Speech Technology, a competence center at KTH, supported by VINNOVA (The Swedish Agency for Innovation Systems), KTH and participating Swedish companies and organizations.

9. References

[1] McRoy, S. W. (1998). Preface - Detecting, repairing and preventing human-machine miscommunication. *International Journal of Human-Computer Studies, 48*, 547-552.

[2] Balentine, B., Morgan, D. P., & Meisel, W. S. (2001). *How to Build a Speech Recognition Application: Second Edition: A Style Guide for Telephony Dialogues*. San Ramon CA: Enterprise Integration Group.

[3] Clark, H. H. (1996). *Using language*. Cambridge: Cambridge University Press.

[4] Walker, M. A., Litman, D., Kamm, C. & Abella, A. (1997). PARADISE: a general framework for evaluating spoken dialogue agents. *Proceedings of the 35th Annual General Meeting of the Association for Computational Linguistics*, 271-280.

[5] Fraser, N. M., & Gilbert, G. N. (1991). Simulating speech systems. *Computer Speech and Language, 5*, 81-99.

[6] Brennan, S. E. (1996). Lexical entrainment in spontaneous dialog. *Proceedings of ISSD*, 41-44.

[7] Amalberti, R., Carbonell, N. & Falzon, P. (1993). User representations of computer systems in human-computer speech interaction. *International Journal of Man-Machine Studies, 38*(4), 547-566.

[8] Anderson, A. H., Bader, M., Bard, E. G., Boyle, E. H., Doherty, G. M., Garrod, S. C., Isard, S. D., Kowtko, J. C., McAllister, J. M., Miller, J., Sotillo, C. F., Thompson, H. S., & Weinert, R. (1991). The HCRC map task corpus. *Language and Speech*, *34*(4), 351-366.

[9] Bell, L. (2000). *Linguistic adaptations in spoken and multimodal dialogue systems.* Licentiate Thesis, Department of Speech, Music and Hearing, KTH.

[10] Norman, D. (1988). *The Design of Everyday Things*. London: MIT Press.