CHAPTER 10

# Summary and discussion

## 10.1 Thesis summary

In spoken dialogue systems, uncertainty and errors are inevitable, mostly due to the error-prone speech recognition process. Even as speech recognition technology improves, users and developers of dialogue systems will likely try to make the interaction more efficient by taking risks and introducing more ambiguity and uncertainty, at least in systems targeted towards more human-like conversational dialogue. A dialogue system must therefore be aware of and react appropriately to these errors. It must have models and methods for detecting potential errors in its hypotheses of what the user is saying, for deciding what to do depending on its uncertainty of these hypotheses and the costs of different outcomes, for displaying its understanding to the user, for making clarification requests, and for detecting errors in propositions it has already accepted.

This thesis has presented experiments on these issues and suggested methods and models for handling them. We will here make a brief summary of these contributions.

### 10.1.1 Methods for exploring human error handling

As stated in the introduction, apparently satisfactory communication may often take place between humans without the listener arriving at a full interpretation of the words used. The question is how this seemingly smooth handling of uncertainty and miscommunication in human-human dialogue can be transferred to human-computer dialogue In Part II of this thesis, two experimental setups were presented, exploring how humans might deal with errors caused by imperfect models in the speech recognition process.

In a first experiment, pairs of subjects were given the task of guiding each other on a virtual campus by talking to each other. The person giving directions (the "operator") could not hear what the other speaker (the "user") said. Instead, the user's speech was recognised by a speech recogniser and the operator could read the results on a screen. This way, their reactions to speech recognition errors in a real dialogue setting could be studied.

In a second experiment, human judges were presented with data collected from the first experiment. Their task was to study the erroneous speech recognition results and try to determine which words were correct and which were not. By varying the amount of information given to them (such as dialogue context, ASR confidence scores, n-best lists), it was possible to study which factors the operators might have relied upon when detecting errors in the first experiment.

### 10.1.2 Non-understanding recovery

The first of the two experiments described above revealed that the operators did not routinely signal non-understanding, such as uttering "what did you say?", when faced with incomprehensible speech recognition results. Instead, they tried to ask task-related questions that confirmed their hypothesis about the user's position. This strategy led to fewer non-understandings of the subsequent user utterance, and thus to a faster recovery from the problem. When they did signal non-understanding, this had a negative effect on the user's experience of task success. Despite the numerous non-understandings, users reported that they were almost always understood.

This is very different from the behaviour of most current dialogue systems. When a system is faced with a non-understanding, it is often assumed that there is nothing left to do but to signal non-understanding and thereby encourage repetition. There are three possible reasons why this strategy failed more often than others in the experiment, and why they often fail in spoken dialogue systems. First, speakers tend to hyperarticulate when repeating themselves, and hyperarticulated speech is something that many speech recognisers do not have in their acoustic models. Second, non-understandings often occur because the utterance to recognise is poorly covered by the speech recognition models (it may even be out-of-vocabulary). If the models did not cover the utterance the first time, chances are that they will not do it a second time either. Third, signalling non-understanding may be perceived as frustrating by the users, who may experience the dialogue as dominated by explicit error handling. Thus, for the design of spoken dialogue systems, the results suggest that when non-understandings occur, a good domain model and robust parsing techniques should be used to, if possible, pose relevant task-related questions to the user, instead of signalling non-understanding.

### 10.1.3 Early error detection

In, the second experiment on human error handling, early error detection on word level was explored, in other words, the immediate detection of erroneous words in the speech recognition results. The results show that, in doing this task, humans benefit from both word confi-

dence scores and 5-best lists delivered by the speech recogniser. Immediate dialogue context (the previous operator/system utterance) was helpful (as long as the recognitions were not too poor), but longer context had no effect.

A machine learning experiment using two different learners for the task (memory-based and transformation-based) showed that word confidence scores were useful for automatic classification, and that other factors may contribute as well. Both lexical and contextual (from the utterance and from the discourse) features further improve performance, especially for content words.

### 10.1.4 Concept- and word-level error handling

A common approach to the decision between accepting and rejecting a hypothesis of what the user has said is to simply use the confidence score and compare it against a threshold. The results presented above, showing that other factors may contribute, is in line with previous research done in the area. However, most previous studies on early error detection have focussed on the detection of errors on the utterance level – the task has been to decide whether a hypothesis of a complete user utterance is correct or not. Such utterance-level error handling is often feasible in command-based dialogue systems where utterances are relatively short and predictable. However, in conversational dialogue systems, utterance-level error handling is often too simplistic. Humans engaging in conversation may often focus on parts of utterances, for example by posing fragmentary clarification requests, and thereby increase the efficiency of the dialogue. In dialogue systems that are targeted towards more human-like conversation, speech recognition results and the semantic interpretations of them may often be partly correct. This calls for error handling on a "sub-utterance" level.

As part of the work for this thesis, the HIGGINS spoken dialogue system has been developed and evaluated. The initial domain for this system has been pedestrian navigation. The system has served as a test-bed for developing and evaluating techniques and models for concept-level error handling, such as robust interpretation, modelling grounding status in the discourse, displaying understanding, posing clarification requests, and late error detection.

### 10.1.5 Robust interpretation

The module for natural language understanding developed for HIGGINS, called PICKERING, is a robust interpreter, designed to parse results from a speech recogniser with n-gram language models in a conversational dialogue system. A context-free grammar (CFG) is used for parsing the input, but to add robustness, the interpreter applies a number of additional techniques to the standard CFG parsing algorithm. It allows unexpected words between and inside phrases, allows non-agreement in phrases, and computes concept-confidence scores.

An evaluation of PICKERING indicates that it generalises well when applied to unseen utterances, within the limited domain, produced by new speakers. As the WER of the ASR output increases, the set of robustness techniques utilised leads to graceful degradation of the interpretation results. The two techniques used to relax the CFG-constraints that were tested –

allowing non-agreement and insertions – both improved performance. Allowing an unlimited number of insertions into syntactical structures caused neither decline nor increase in accuracy.

### 10.1.6 Modelling grounding status

In HIGGINS, the dialogue management is divided into discourse modelling and action selection. The discourse modeller, called GALATEA, can be regarded as a final step in the interpretation processing chain, in which the dialogue context is considered. The task of GALATEA is to resolve ellipses and anaphora, but also to model the grounding status of concepts. This grounding status contains information about who has mentioned a given concept, when it has been mentioned, the surface realisation of it, and the system's confidence in it. As the same concept is mentioned several times, the grounding information gets unified and the grounding status is boosted. This way, the system may identify concepts in which it has low confidence, and then decide to display its understanding to the user or make a clarification request. However, since the confidence is represented in the model, the system may also postpone error handling and identify misunderstandings at a later stage, so-called late error detection. GALATEA does not only model utterances from the user, but also from the system. The system may therefore track how its own actions affect the grounding status of concepts. Since GALATEA also resolves ellipses and anaphora, the choice of utterance realisation (for example choice of referring expression) will affect how the grounding status gets updated.

An evaluation of the complete HIGGINS system showed that the performance of GALATEA and the rest of the system looks promising, not only when utterances are correctly recognised, but also when ASR errors are introduced.

### 10.1.7 Making grounding decisions

In the initial implementation of HIGGINS, the grounding decisions – that is, decisions about which recognition hypotheses to consider as correct and which grounding actions to take – were, as in most dialogue systems, based on hand-crafted confidence thresholds. In such an approach, a low confidence leads to rejection, a mid confidence to a clarification or display of understanding, and a high confidence to a silent acceptance. The problem with this approach is that the thresholds used are static, and not based on any empirical material or any theoretically sound model.

Based on the data collected for evaluating HIGGINS, a decision-theoretic approach was used to build a data-driven model for making grounding decisions. Ideally, the grounding decision should take into account the uncertainty of the hypothesis, the costs involved in taking the different grounding actions, and the costs of rejecting a correct hypothesis or falsely accepting an incorrect hypothesis. Based on task analysis of the HIGGINS navigation domain, cost functions that take these factors into account were derived. It was argued that efficiency– the number of syllables uttered by the user and system – is useful as a cost measure for the navigation domain. Dialogue data was then used to estimate parameters for these cost functions, so that the grounding decision may be based on both confidence and dialogue context.

For example, it was shown how concepts with high information gain should more often be clarified than concepts with low information gain, which are either simply rejected or accepted. To silently accept a concept which is associated with a very high cost of misunderstanding, a very high confidence in this concept is required.

### 10.1.8 Fragmentary grounding

Fragmentary utterances, like S.2 in the following example, are commonly used as evidence of understanding in human-human dialogue:

(69) U.1: I can see a red building.
S.2: *Red?*

By using fragmentary grounding, speakers may not only improve the efficiency of the dialogue, they may also pinpoint the source of the problem (in the example, the word "red").

In this thesis, the use of fragmentary grounding for concept-level error handling in spoken dialogue systems has been explored. It has been shown how such utterances are produced and interpreted in the HIGGINS system, that is, how the system may choose the right lexical realisation and how the grounding status gets updated after the user has responded. There have previously not been many studies on the use of fragmentary clarification requests in spoken dialogue systems interacting with real users. The results from the HIGGINS evaluation showed that users of spoken dialogue systems may have difficulties understanding fragmentary clarification, especially after incorrect speech recognition hypotheses, and that further research on when to use them, how to realise them, and how to understand the user's reaction to them is needed.

One of the difficulties with fragmentary grounding utterances is that they provide little syntactic guidance, and that their pragmatic meaning therefore is dependent on context and prosody to a large extent. In two experiments, the relation between prosodic realisation and interpretation of such utterances was explored. In the first experiment, subjects listened to synthesised fragmentary grounding utterances in which prosodic features were varied systematically, and the subjects were given the task of choosing between different paraphrases. The results showed that the position and height of the $F_0$ peak not only affects whether the grounding utterance is interpreted as positive or negative evidence, but also which level of action is concerned in the case of negative evidence. In a second experiment, a Wizard-of-Oz setting was used to show that users of spoken dialogue systems not only perceive the differences in prosody and the pragmatic meaning of grounding utterances, but that they also change their behaviour accordingly in a human-computer dialogue setting.

## 10.2 Discussion and future work

### 10.2.1 Integration and improvements

As stated in the introduction, all aspects of error handling cannot be covered in the scope of this thesis, but the ambition has been to study how error handling may be performed in different parts of a complete spoken dialogue system. It should be noted, though, that not all of the models and methods explored in this thesis are currently integrated in the HIGGINS system. There are also many ways in which they may be improved and explored further. We will continue with a brief discussion on some of these issues.

The machine-learning methods for early error detection explored in Chapter 5 are not yet used by the HIGGINS system to sort out incorrect words. For these methods to be really useful in HIGGINS, methods for assigning probabilistic scores instead of binary decisions should be explored. A further improvement would be to do this confidence estimation on the concept-level (i.e., on the PICKERING results), resulting in concept confidence scores that may be used for modelling grounding status in GALATEA.

The action selection in HIGGINS was divided into a navigation action manager (NAM) and a grounding action manager (GAM), where only the NAM consults the domain database. The purpose of this division was to make the GAM more generic and responsive. However, if the grounding decision model proposed in Chapter 8 was to be implemented in HIGGINS, this division may not be viable. In order to make dynamic grounding decisions that take concept information gain into account, the grounding decision maker has to consult the domain database.

Much more work is needed to understand the relation between prosody and the pragmatic effects of fragmentary grounding utterances, before it results in a full model that may be used in a spoken dialogue system. However, the explorations in Chapter 9 may be regarded as a step towards such a model for Swedish.

In Chapter 6, it was shown how clarification on the perception level is handled in HIGGINS. An important next step is to model the clarification of ambiguous referring expressions or ellipses. As the system starts to make fragmentary clarification requests, users are likely to do this as well. Thus, the system must also be capable of recognising such requests.

It was also shown how the modelling of grounding status and confidence over time made it possible for the system to postpone error handling and detect errors later on (late error detection). However, while this information is stored for later use, a very simplistic model for detecting errors based on this information was used. Here, an empirical approach (such as machine-learning) would be interesting to explore, as discussed in 6.8. Another aspect of late error detection is to understand the users' reactions after display of understanding and identify the errors that they may reflect. As was evident in Chapter 8, display of understanding after incorrect recognitions very often failed, in the sense that the users did not react to them in a way that allowed the system to repair the misunderstanding.

### 10.2.2 Generalisation

Throughout this thesis, the guiding and navigation domain has been used in experiments and implementations. This has allowed us to reuse data and experiences between the different studies. As pointed out in previous chapters, the domain is similar to the Map Task setting used in many linguistic studies on human-human dialogue. There are several reasons why this domain is so frequently studied. Since the maps used are provided by the experimenter, it gives her control over the task and what the subjects will talk about. If we want to study conversational dialogue systems, this predictability is very useful, since the vocabulary may be restricted. The Map Task also allows the experimenter to make the maps different, in order to study how speakers deal with such problems. This has not been done in the work presented here, since the focus has been on problems that arise from speech recognition errors. Another feature of this domain is the frequent use of referring expressions and anaphora, which is interesting from a grounding perspective. As pointed out in 2.3.3.1, anaphora is not very often addressed in dialogue systems. One explanation for this is that anaphora is not typically needed in the extensively studied travel booking domain.

It remains to be investigated to what extent the results obtained in this thesis apply to other types of dialogue systems in other domains. The possibility to generalise the results on human non-understanding recovery presented in Chapter 4 was discussed in 4.4.2. The methods for early error detection investigated in Chapter 5 should be applicable to other systems and domains. However, the set of features that is found to be useful will likely vary. The generic modules in the HIGGINS dialogue system presented in Chapter 6 have been used to implement a few other domains, as mentioned in 6.8. The general model and parameters for making grounding decisions presented in Chapter 8 should be applicable to other domains. However, as discussed, the more specific estimation of the task-dependent parameters must be adapted for the domain. The tentative prosodic model presented in Chapter 9 should not be domain dependent.

As stated in the introduction, two general themes in this thesis have been to draw lessons from human error handling and to explore concept-level error handling. These are issues that may be more important for conversational dialogue systems than for command-based systems, in which more human-like error handling strategies may seem confusing and in which a typical utterance may not contain too many concepts. However, early error detection and grounding decisions are important in such systems as well, and the methods proposed here for these issues should be applicable. It is possible that other error handling issues than the ones addressed in this thesis may be more important in command-based systems. For example, the "speech graffiti" approach discussed in 2.1 may be used to provide the user with a pre-defined set of error correction commands. A much larger vocabulary and shorter utterances may also increase the usefulness of n-best lists, as discussed in 3.3.1.4.