Collaborating on Utterances with a Spoken Dialogue System Using an ISU-based Approach to Incremental Dialogue Management

Okko Buß, Timo Baumann, David Schlangen
Department of Linguistics
University of Potsdam, Germany
{okko|timo|das}@ling.uni-potsdam.de

Abstract

When dialogue systems, through the use of incremental processing, are not bounded anymore by strict, non-overlapping turn-taking, a whole range of additional interactional devices becomes available. We explore the use of one such device, trial intonation. We elaborate our approach to dialogue management in incremental systems, based on the Information-State-Update approach, and discuss an implementation in a micro-domain that lends itself to the use of immediate feedback, trial intonations and expansions. In an overhearer evaluation, the incremental system was judged as significantly more human-like and reactive than a non-incremental version.

1 Introduction

In human–human dialogue, most utterances have only one speaker.[1] However, the shape that an utterance ultimately takes on is often determined not just by the one speaker, but also by her addressees. A speaker intending to refer to something may start with a description, monitor while they go on whether the description appears to be understood sufficiently well, and if not, possibly extend it, rather than finishing the utterance in the form that was initially planned. This monitoring within the utterance is sometimes even made very explicit, as in the following example from (Clark, 1996):

(1) A: A man called Annegra? -
B: yeah, Allegra
A: Allegra, uh, replied and, uh, ...

In this example, A makes use of what Sacks and Schegloff (1979) called a *try marker*, a "questioning upward intonational contour, followed by a brief pause". As discussed by Clark (1996), this device is an efficient solution to the problem posed by uncertainty on the side of the speaker whether a reference is going to be understood, as it checks for understanding *in situ*, and lets the conversation partners collaborate on the utterance that is in production.

Spoken dialogue systems (SDS) typically cannot achieve the close coupling between production and interpretation that is needed for this to work, as normally the smallest unit on which they operate is the full utterance (or, more precisely, the turn). (For a discussion see e.g. (Skantze and Schlangen, 2009).) We present here an approach to managing dialogue in an incremental SDS that can handle this phenomenon, explaining how it is implemented in system (Section 4) that works in a micro-domain (which is described in Section 3). As we will discuss in the next section, this goes beyond earlier work on incremental SDS, combining the production of multimodal feedback (as in (Aist et al., 2007)) with fast interaction in a semantically more complex domain (compared to (Skantze and Schlangen, 2009)).

2 Related Work

Collaboration on utterances has not often been modelled in SDS, as it presupposes fully incremental processing, which itself is still something of a rarity in such systems. (There is work on collaborative reference (DeVault et al., 2005; Heeman and Hirst, 1995), but that focuses on written input, and on collaboration over several utterances and not within utterances.) There are two systems that are directly relevant here.

The system described in (Aist et al., 2007) is able to produce some of the phenomena that we are interested in here. The set-up is a simple reference game (as we will see, the domain we have chosen is very similar), where users can refer to objects shown on the screen, and the SDS gives continuous feedback about its understand-

[1] Though by far not all; see (Clark, 1996; Purver et al., 2009; Poesio and Rieser, 2010).

ing by performing on-screen actions. While we do produce similar non-linguistic behaviour in our system, we also go beyond this by producing verbal feedback that responds to the certainty of the speaker (expressed by the use of trial intonation). Unfortunately, very little technical details are given in that paper, so that we cannot compare the approaches more fully.

Even more closely related is some of our own previous work, (Skantze and Schlangen, 2009), where we modeled fast system reactions to delivery of information in installments in a number sequence dictation domain. In a small corpus study, we found a very pronounced use of trial or installment intonations, with the first installments of numbers being bounded by rising intonation, and the final installment of a sequence by falling intonation. We made use of this fact by letting the system distinguish these situations based on prosody, and giving it different reaction possibilities (backchannel feedback vs. explicit confirmation).

The work reported here is a direct scaling up of that work. For number sequences, the notion of utterance is somewhat vague, as there are no syntactic constraints that help demarcate its boundaries. Moreover, there is no semantics (beyond the individual number) that could pose problems – the main problem for the speaker in that domain is ensuring that the signal is correctly *identified* (as in, the string could be written down), and the trial intonation is meant to provide opportunities for grounding whether that is the fact. Here, we want to go beyond that and look at utterances where it is the intended meaning whose recognition the speaker is unsure about (grounding at level 3 rather than (just) at level 2 in terms of (Clark, 1996).) This difference leads to differences in the follow up potential: where in the numbers domain, typical repair follow-ups were *repetitions*, in semantically more complex domains we can expect *expansions* or *reformulations*.

3 The Puzzle Micro-Domain

To investigate these issues in a controlled setting, we chose a domain that makes complex and possibly underspecified references likely, and that also allows a combination of linguistic and non-linguistic feedback. In this domain, the user's goal is to instruct the system to pick up and manipulate Tetris-like puzzle pieces, which are shown on the screen. We recorded human–human as well as human–(simulated) machine interactions in this domain, and indeed found frequent use of "packaging" of instructions, and immediate feedback, as in (2) (arrow indicating intonation).

(2)
IG-1: The cross in the corner↗ ...
IF-2: erm
IG-3: the red one .. yeah
IF-4: [moves cursor]
IG-5: take that.

We chose these as our target phenomena for the implementation: intra-utterance hesitations, possibly with trial intonation (as in line 2);[2] immediate execution of actions (line 4), and their grounding role as display of understanding ("yeah" in line 3). The system controls the mouse cursor, e.g. moving it over pieces once it has a good hypothesis about a reference; other actions are visualised similarly.

4 Implementation

4.1 Overview

Our system is realised as a collection of incremental processing modules in the InproToolKit (Schlangen et al., 2010), a middle-ware package that implements some of the features of the model of incremental processing of (Schlangen and Skantze, 2009). The modules used in the implementation will be described briefly below.

4.2 ASR, Prosody, Floor Tracker & NLU

For speech recognition, we use Sphinx-4 (Walker et al., 2004), with our own extensions for incremental speech recognition (Baumann et al., 2009), and our own domain-specific acoustic model. For the experiments described here, we used a recognition grammar.

Another module performs online prosodic analysis, based on pitch change, which is measured in semi-tone per second over the turn-final word, using a modified YIN (de Cheveigné and Kawahara, 2002). Based on the slope of the f_0 curve, we classify pitch as rising or falling.

This information is used by the floor tracking module, which notifies the dialogue manager (DM) about changes in floor status. These status changes are classified by simple rules: silence following rising pitch leads to a timeout signal

[2] Although we chose to label this "intra-utterance" here, it doesn't matter much for our approach whether one considers this example to consist of one or several utterances; what matters is that differences in intonation and pragmatic completeness have an effect.

```
{< a ( 1 action=A=take; 2 prepare(A)   ; 3 U),
     ( 4 tile=T       ; 5 highlight(T) ; 6 U),
     ( 7              ; 8 execute(A,T) ; 9 U) >

 < b (10 action=A=del ;11 prepare(A)   ;12 U),
     (13 tile=T       ;14 highlight(T) ;15 U),
     (16              ;17 execute(A,T) ;18 U) >}
```

Figure 1: Example iQUD

sent to the DM faster (200ms) than silence after falling pitch (500ms). (Comparable to the rules in (Skantze and Schlangen, 2009).)

Natural language understanding finally is performed by a unification-based semantic composer, which builds simple semantic representations out of the lexical entries for the recognised words; and a resolver, which matches these representations against knowledge of the objects in the domain.

4.3 Dialogue Manager and Action Manager

The DM reacts to input from three sides: semantic material coming from the NLU, floor state signals from the floor tracker, and notifications about execution of actions from the action manager.

The central element of the information state used in the dialogue manager is what we call the iQUD (for *incremental* Question under Discussion, as it's a variant of the QUD of (Ginzburg, 1996)). Figure 1 gives an example. The iQUD collects all relevant sub-questions into one structure, which also records what the relevant non-linguistic actions are (RNLAs; more on this in a second, but see also (Buß and Schlangen, 2010), where we've sketched this approach before), and what the grounding status is of that sub-question.

Let's go through example (2). The iQUD in Figure 1 represents the state after the system has asked "what shall I do now?". The system anticipates two alternative replies, a *take* request, or a *delete* request; this is what the specification of the slot value in 1 and 10 in the iQUD indicates. Now the user starts to speak and produces what is shown in line 1 in the example. The floor tracker reacts to the rising pitch and to the silence of appropriate length, and notifies the dialogue manager. In the meantime, the DM has received updates from the NLU module, has checked for each update whether it is *relevant* to a sub-question on the iQUD, and if so, whether it *resolves* it. In this situation, the material was relevant to both 4 and 13, but did not resolve it. This is a precondition for the *continuer-questioning* rule, which is triggered by the signal from the floor tracker. The system then back-channels as in the example, indicating acoustic understanding (Clark's level 2), but failure to operate on the understanding (level 3). (As an aside, we found that it is far from trivial to find the right wording for this prompt. We settled on an "erm" with level pitch.)

The user then indeed produces more material, which together with the previously given information resolves the question. This is where the RNLAs come in: when a sub-question is resolved, the DM looks into the field for RNLAs, and if there are any, puts them up for execution to the action manager. In our case, slots 4 and 13 are both applicable, but as they have compatible RNLAs, this does not cause a conflict. When the action has been performed, a new question is accommodated (not shown here), which can be paraphrased as "was the understanding displayed through this action correct?". This is what allows the user reply in line 3 to be integrated, which otherwise would need to be ignored, or even worse, would confuse a dialogue system. A relevant continuation, on the other hand, would also have resolved the question. We consider this modelling of grounding effects of *actions* an important feature of our approach.

Similar rules handle other floor tracker events; not elaborated here for reasons of space. In our current prototype the rules are hard-coded, but we are preparing a version where rules and information-states can be specified externally and are read in by a rule-engine.

4.4 Overhearer Evaluation

Evaluating the contribution of one of the many modules in an SDS is notoriously difficult (Walker et al., 1998). To be able to focus on evaluation of the incremental dialogue strategies and avoid interference from ASR problems (and more technical problems; our system is still somewhat fragile), we opted for an overhearer evaluation. (Such a setting was also used for the test of the incremental system of (Aist et al., 2007).)

We implemented a non-incremental version of the system that does not give non-linguistic feedback during user utterances and has only one, fixed, timeout of 800ms (comparable to typical settings in commercial dialogue systems). Two of the authors then recorded 30 minutes of interactions with the two versions of the system.We then identified and discarded "outlier" interactions, i.e. those with technical problems, or where

recognition problems were so severe that a non-understanding state was entered repeatedly. These criteria were meant to be fair to both versions of the system, and indeed we excluded similar numbers of failed interactions from both versions (around 10 % of interactions in total).

We measured the length of interactions in the two sets, and found that the interactions in the incremental setting were significantly shorter (t-test, $p < 0.005$). This was to be expected, of course, as the incremental strategies allow faster reactions (execution time can be folded into the user utterance); other outcomes would have been possible, though, if the incremental version had systematically more understanding problems.

We then had 8 subjects (university students, not involved in the research) watch and directly judge (questionnaire, Likert-scale replies to questions about human-likeness, helpfulness, and reactivity) 34 randomly selected interactions from either condition. Human-likeness and reactivity were judged significantly higher for the incremental version (Wilcoxon rank-sum test; $p < 0.05$ and $p < 0.005$, respectively), while there was no effect for helpfulness ($p = 0.06$).

5 Conclusions

We described our incremental micro-domain dialogue system, which is capable of reacting to subtle signals from the user about expected feedback, and is able to produce overlapping non-linguistic actions, modelling their effect as displays of understanding. Interactions with the system were judged by overhearers to be more human-like and reactive than with a non-incremental variant. We are currently working on extending and generalising our approach to incremental dialogue management, porting it to other domains.

Acknowledgments Funded by an ENP grant from DFG.

References

Gregory Aist, James Allen, Ellen Campana, Carlos Gomez Gallo, Scott Stoness, Mary Swift, and Michael K. Tanenhaus. 2007. Incremental understanding in human-computer dialogue and experimental evidence for advantages over nonincremental methods. In *Proceedings of Decalog (Semdial 2007)*, Trento, Italy.

Timo Baumann, Michaela Atterer, and David Schlangen. 2009. Assessing and Improving the Performance of Speech Recognition for Incremental Systems. In *Proceedings of NAACL-HLT 2009*, Boulder, USA.

Okko Buß and David Schlangen. 2010. Modelling sub-utterance phenomena in spoken dialogue systems. In *Proceedings of Semdial 2010 ("Pozdial")*, pages 33–41, Poznan, Poland, June.

Herbert H. Clark. 1996. *Using Language*. Cambridge University Press, Cambridge.

Alain de Cheveigné and Hideki Kawahara. 2002. YIN, a fundamental frequency estimator for speech and music. *Journal of the Acoustical Society of America*, 111(4):1917–1930.

David DeVault, Natalia Kariaeva, Anubha Kothari, Iris Oved, and Matthew Stone. 2005. An information-state approach to collaborative reference. In *Short Papers, ACL 2005*, Michigan, USA, June.

Jonathan Ginzburg. 1996. Interrogatives: Questions, facts and dialogue. In Shalom Lappin, editor, *The Handbook of Contemporary Semantic Theory*. Blackwell, Oxford.

Peter A. Heeman and Graeme Hirst. 1995. Collaborating on referring expressions. *Computational Linguistics*, 21(3):351–382.

Massimo Poesio and Hannes Rieser. 2010. Completions, coordination, and alignment in dialogue. *Dialogue and Discourse*, 1(1):1–89.

Matthew Purver, Christine Howes, Eleni Gregoromichelaki, and Patrick Healey. 2009. Split utterances in dialogue: a corpus study. In *Proceedings of the SIGDIAL 2009*, pages 262–271, London, UK, September.

Harvey Sacks and Emanuel A. Schegloff. 1979. Two preferences in the organization of reference to persons in conversation and their interaction. In George Psathas, editor, *Everyday Language: Studies in Ethnomethodology*, pages 15–21. Irvington Publishers, Inc., New York, NY, USA.

David Schlangen and Gabriel Skantze. 2009. A general, abstract model of incremental dialogue processing. In *Proceedings of EACL 2009*, pages 710–718, Athens, Greece, March.

David Schlangen, Timo Baumann, Hendrik Buschmeier, Okko Buß, Stefan Kopp, Gabriel Skantze, and Ramin Yaghoubzadeh. 2010. Middleware for incremental processing in conversational agents. In *Proceedings of SIGDIAL 2010*, Tokyo, Japan.

Gabriel Skantze and David Schlangen. 2009. Incremental dialogue processing in a micro-domain. In *Proceedings of EACL 2009*, pages 745–753, Athens, Greece, March.

Marilyn A. Walker, Diane J. Litman, Candace A. Kamm, and Alicia Abella. 1998. Evaluating spoken dialogue agents with PARADISE: Two case studies. *Computer Speech and Language*, 12(3).

Willie Walker, Paul Lamere, Philip Kwok, Bhiksha Raj, Rita Singh, Evandro Gouvea, Peter Wolf, and Joe Woelfel. 2004. Sphinx-4: A flexible open source framework for speech recognition. Technical report, Sun Microsystems Inc.