Interactive Communication Management in an Issue-based Dialogue System

Staffan Larsson
Department of Linguistics
Göteborg University
sl@ling.gu.se

Abstract

We first give an overview of Interactive Communication Management (ICM), and especially feedback, in human-human communication. We then select a limited range of ICM behaviour useful for dialogue systems, and show how these utterances and responses to them are managed in a dialogue system implementing a theory of issue-based dialogue management.

1 Introduction

In this paper[1], we will give an account of the generation of feedback and sequencing in a dialogue system. Allwood (1995) uses the concept of "Interactive Communication Management" (ICM) to designate all communication dealing with the management of dialogue interaction. This includes feedback but also *sequencing* and *turn management*. Sequencing "concerns the mechanisms, whereby a dialogue is structured into sequences, subactivities, topics etc.".

Here, we will use the term ICM as a general term for coordination of the common ground, which in an information state update approach (Traum and Larsson, forth) comes to mean explicit signals (formalised as dialogue moves) enabling coordination of updates to the common ground. While feedback moves are associated with specific utterances (usually the utterance preceding the feedback move), ICM in general does not need to concern any specific utterance.

We start by briefly discussing ICM in current dialogue systems. We then proceed to give an overview of ICM, and especially feedback, in human-human communication. Finally, we select a limited range of ICM behaviour which we consider useful for dialogue systems, and show a brief example of how these utterances and responses to them are managed (selected, generated, and/or integrated) in GoDiS , a dialogue system implementing a theory of issue-based dialogue management (Larsson, 2002).

2 Verification

In the literature concerning practical dialogue systems, e.g. San-Segundo et al. (2001), grounding is often reduced to verification of the system's recognition of user utterances. Two common ways of handling verification are described as "explicit" and "implicit" verification, exemplified below (example from San-Segundo et al. (2001)).

- I understood you want to depart from Madrid. Is that correct? [explicit]
- You leave from Madrid. Where are you arriving at? [implicit]

Actually, both "explicit" and "implicit" feedback contain a verbatim repetition or a reformulation of the original utterance, and in this sense they are

[1]This paper is based on chapter 4 of (Larsson, 2002).

both explicit. The actual base for the distinction is that the first utterance (1) signals understanding but also a lack of confidence in the interpretation and (2) is trying to elicit a response regarding the correctness of the interpretation from the hearer, whereas the second utterance (1) signals confident understanding and (2) does not try to elicit a response.

In human-human dialogue, explicit confirmations are not very common. Presumably, they occur primarily in noisy environments and in situations where understanding is critical (e.g. when arranging a meeting in a busy airport). However, in dialogue systems, verification of user utterances are of central importance given the quality of current speaker-independent speech recognition. This explains to some extent why verification is often the only aspect of feedback handled by current systems - it is simply necessary. There is no reason, however, not to explore the possible uses of a wider range of feedback behaviour in dialogue systems. The obvious starting point for such exploration is human feedback behaviour.

3 Feedback and related behaviour in human-human dialogue

Clark and Schaefer (1989) describe *grounding* as the process of adding to the common ground. By *feedback* we mean behaviour whose *primary* function is to deal with grounding of utterances in dialogue[2]. This distinguishes feedback from behaviour whose primary function is related to the domain-level task at hand, e.g., getting price information. Non-feedback behaviour in this sense includes asking and answering task-level questions, giving instructions, etc. (cf. the "Core Speech Acts" of Poesio and Traum (1998)).

Answering a domain-level question (e.g., saying "Paris" in response to "What city do you want to go to?") certainly involves aspects of grounding and acceptance, since it shows that the question was understood and accepted. However, the primary function of a domain-level answer is to resolve the question, not to show that it was understood and accepted.

To get an overview of the range of explicit feedback behaviour that exists in human-human dialogue, we will classify feedback according to two basic criteria. (Below, we will assume a situation where a DP (Dialogue Participant) S has just uttered or is uttering u to a DP H, when the feedback utterance f, uttered by H to S, occurs.)

- level of action (basic communicative function): contact, perception, understanding or acceptance
- polarity: positive, negative or checking

These two criteria are loosely based on (Allwood et al., 1992) and (Allwood, 1995). Additional possible criteria, which will not be discussed further here (but see Larsson (2002)), are eliciting/non-eliciting, object-level/meta-level content, and surface form.

3.1 Levels of action in dialogue

Both Allwood (1995) and Clark (1996) distinguish four levels of action involved in communication [3].

- Acceptance (reaction, consideration): whether H has accepted and integrated (the content of) u
- Understanding (recognition): whether H understands u
- Perception (identification): whether H perceives u
- Contact (attention): whether H and S have contact, i.e., if they have established a channel of communication

These levels of action are involved in all dialogue, and to the extent that contact, perception, understanding and acceptance can be said to be negotiated, all human-human dialogue has a minimal element of negotiation built in.

[2] Since this paper is not concerned with multimodal dialogue, we will only discuss verbal feedback.

[3] Allwood and Clark use slightly different terminologies; here we use Allwood's terminology (and, for the acceptance level, also Ginzburgs) and add Clark's corresponding terms in parenthesis. The definitions are mainly derived from Allwood.

Given that grounding is concerned with all levels, it follows that feedback may concern any of these levels (and possibly several levels simultaneously).

3.2 Levels of understanding

We can make further distinctions between different levels of understanding, corresponding to three levels of meaning. These sublevels give a finer grading to the level of understanding[4].

- discourse-independent (but possibly domain-dependent) meaning (roughly corresponding to "meaning" in the terminology of Barwise and Perry (1983) and Kaplan (1979)), e.g., static word meanings
- domain-dependent and discourse-dependent meaning (roughly, "content" in the terminology of Barwise and Perry (1983) and Kaplan (1979))
  - referential meaning, e.g., referents of pronouns, temporal expressions
  - pragmatic: the relevance of u in the current context

By "discourse-independent" we mean "independent of the dynamic dialogue context." However, discourse-independent meaning may still be dependent on static aspects of the activity/domain. It is obvious that these levels of meaning are intertwined and do not have perfectly clear boundaries. Nevertheless, we believe they are useful as analytical approximations.

Since dialogue systems usually operate in limited domains, we will assume that we do not have to deal with ambiguities which are resolved by static knowledge related to the domain. For example, a dialogue system for accessing bank accounts does not have to know that "bank" may also refer to the bank of a river; it is simply very unlikely (though of course not impossible) that the word will be used with this meaning in the activity. It can be questioned whether this is always a good strategy, but for now we accept this as a reasonable simplification.

[4]A similar distinction is also used by Ginzburg (forth).

In this paper, we will not be dealing with referential meaning (simply because referent identification is currently not implemented in GoDiS). We will use the term *semantic meaning* to refer to discourse-independent meaning, and contrast this to pragmatic meaning.

3.3 Positive, negative, and checking feedback

Positive feedback indicates one or several of contact, perception, understanding, and acceptance, while negative feedback indicates lack thereof.

Negative feedback can be caused by failure to integrate U on any of the levels of action in dialogue:

- lack of contact - H and S have not established a channel of communication, or the channel has broken down
- lack of perception - H did not hear what S said
- lack of understanding on a semantic/pragmatic level - H recognized all the words, but could not extract a content
  - semantic meaning, e.g., word meanings
  - pragmatic meaning, i.e., the relevance of S's utterance in relation to the context
- rejection (lack of acceptance) of content

While there are clear cases of positive ("uhuh", "ok") and negative ("pardon?", "I don't understand") feedback, there are also some cases which are not so clear. For example, are check-questions (e.g., "To Paris?" in response to "I want to go to Paris") positive or negative? If positive feedback shows understanding, and negative feedback lack of understanding, then check-questions are somewhere in between; they indicate understanding but also that the lack of confidence in that understanding.

We assume a third category of *checking* feedback for check-questions and similar feedback types. If negative feedback indicates a lack of understanding, checking feedback indicates lack of confidence in one's understanding.

4 Feedback in GoDiS

In this section, we first show how feedback dialogue moves in GoDiS are represented. We then review the full range of feedback moves, starting with system-generated feedback and then moving on to user feedback.

The general notation for feedback ICM moves used in GoDiS is the following:

- icm:$L * P\{{:}Arg\}$

where L is an action level, P is a polarity, and Arg is an argument.

For user utterances, GoDiS will be able to produce the following kinds of feedback utterances (for the examples, assume that the user just said "I want to go to Paris"):

- L: action level
  - con: contact ("Are you there?")
  - per: perception ("I didn't hear anything from you", "I heard you say 'to Paris'")
  - sem: semantic understanding ("I don't understand", "To Paris.")
  - und: pragmatic understanding ("I don't quite understand", "You want to know about price.")
  - acc: acceptance/reaction ("Sorry, I can't answer questions about connecting flights", "Okay.")
- P: polarity
  - neg: negative
  - chk: checking
  - pos: positive
- Arg: argument

The arguments are different aspects of the utterance or move which is being grounded, depending action level:

- for per-level: $String$, the recognized string
- for sem-level: $Move$, a move interpreted from the utterance, including (possibly underspecified) semantic content
- for und-level: C, where C : Proposition is the propositional (fully specified) content of the utterance
- for acc-level: C : Proposition, the content of the utterance

For example, the ICM move icm:und$*$pos:dest-city(paris) provides positive feedback regarding an utterance that has been understood as meaning that the destination city is Paris.

For user utterances, GoDiS will be able to produce e.g., the following kinds of feedback utterances (for the examples, assume that the user just said "I want to go to Paris"):

- contact
  - negative; icm:con$*$neg ("I didn't hear anything from you")
- perception
  - negative; icm:per$*$neg realized as fb-phrase ("Pardon?", "I didn't hear what you said.")
  - positive; icm:per$*$pos:$String$ realized as metalevel verbatim repetition ("I heard 'to paris' ")
- understanding (semantic)
  - negative; icm:sem$*$neg realized as fb-phrase ("I don't understand.")
  - positive; icm:sem$*$pos:$Content$ realized as repetition/reformulation of content ("To Paris."[5])
- understanding (pragmatic)
  - negative; icm:und$*$neg realized as fb-phrase ("I don't quite understand.")
  - positive; icm:und$*$pos:$Content$ realized as repetition/reformulation of content (object-level) ("To Paris.")
  - checking; icm:und$*$chk:$Content$ realized as ask about interpretation ("To Paris, is that correct?")

[5] For a more informative example, see utterance 9 in the dialogue on page 5.

- acceptance
  - negative: icm:acc$*$neg:$Content$ realized as explanation ("Sorry, Paris is not a valid destination city")
  - positive; icm:acc$*$pos realized as fb-word ("Okay")

As a subcase of negative acceptance (i.e., rejection), GoDiS is able to reject user questions using the icm:acc$*$neg:issue(Q) move, where Q is a question, as illustrated below (U is the user, S is the system)[6]:

```
U> What about connecting flights?
S> Sorry, I cannot answer questions about connecting flights.
```

We are not claiming that humans always make these distinctions explicitly or even consciously, nor that the link between surface form and feedback type is a simple one-to-one correspondence; for example, "mm" may be used as positive feedback on the perception, understanding, and acceptance levels. Feedback is, simply, often ambiguous. However, since GoDiS is making all these distinctions internally we might as well try to produce feedback which is not so ambiguous.

Of course, there is also a tradeoff in relation to brevity; extremely explicit feedback (e.g., "I understood that you referred to Paris, but I don't see how that is relevant right now.") could be irritating and might decrease the efficiency of the dialogue. We feel that the current choices of surface forms are fairly reasonable, but testing and evaluation on real users would be needed to find the best ways to formulate feedback on different levels. This is an area for future research.

A general strategy used by GoDiS in ICM selection is that if negative or checking feedback on some level is provided, the system should also provide positive feedback on the level below. For example, if the system produces negative feedback on the pragmatic understanding level, it should also produce positive feedback on the semantic understanding level:

```
S> To Paris. I don't quite understand.
```

For system utterances, GoDiS will react appropriately to the following types of user-initiated feedback:

- perception level
  - negative; fb-phrase ("Pardon?", "Excuse me?", "Sorry, I didn't hear you") interpreted as icm:per$*$neg
- acceptance level
  - positive; fb-phrase ("Okay.") interpreted as icm:acc$*$pos
  - negative; issue rejection fb-phrase ("I don't know", "Never mind", "It doesn't matter") interpreted as icm:acc$*$neg:issue

In addition, irrelevant followups to system ask-moves are regarded as implicit issue-rejections. The coverage of user feedback behaviour is thus more limited than the coverage for system behaviour. The main motivation for this is that system utterances are less likely to be problematic for the user to interpret than vice versa.

5 Issue-based grounding in GoDiS

Different kinds of feedback from the system carry with them different expectations on and options for the user. This is especially important in the case where the system is unsure whether it has correctly interpreted the user's utterance, or in cases where the system misinterprets the user. In the issue-based approach to feedback management, this is accounted for by bringing under discussion questions related to grounding.

On our analysis, checking feedback on the understanding level with content C introduces a meaning-question ?und(C), roughly paraphrasable as "Is C a correct interpretation the latest utterance?". This question is available for resolution of elliptical answers (for y/n-questions, "yes" and "no"), and must be resolved before the dialogue can proceed. If the system produces this kind of feedback, the user may respond "yes" or

[6] Note that this requires the system to recognize and understand common questions that it cannot answer; in current systems this is usually not (if ever) the case.

"no" depending on whether the interpretation was correct or not; only when a positive answer to the understanding-question is received does the system assume that C is a part of the shared dialogue state. If no answer is received, the system cautiously assumes that the interpretation was erroneous.

U> `To Lund please`

S> `To London?`
(understanding-question now raised and unresolved, and available for resolution of short answers)

U> `(silence)`
(since the understanding-question is still not resolved, neither is the issue of destination)

S> `So, what city do you want to go to?`

When producing positive feedback on the understanding level, the understanding-question is not raised explicitly as was the case with checking feedback. The interpretation is tentatively assumed to be correct and is added to the shared dialogue state. The understanding-question is thus (tentatively) assumed to be resolved. However, on our analysis, positive feedback still makes the understanding-issue available for elliptical responses, so that the user may respond to the positive feedback using "yes" or "no".

This is possible in GoDiS because of the distinction between a local ("QUD", see Ginzburg (1996)) and a global ("ISSUES") stack of questions under discussion, where the former represents questions available for ellipsis resolution and the latter contains questions which have been raised but not yet resolved. If a question q is on QUD but not on ISSUES, an answer to q will trigger an accommodation rule copying q from QUD to ISSUES, thus reraising q and retracting any previously stored answers to q. In the case of grounding, where q is an understanding-question, this amounts to retracting the assumption that the previous user move was correctly understood.

Thus, if a negative response to positive feedback on the understanding level is received, the system retracts its assumption about the correctness of its interpretation:

U> `To Lund please`

S> `OK, to London.`
(understanding-question now tentatively resolved but available for resolution of short answers)
[formally, ?und(dest-city(london)) $\in$ QUD[7]]

U> `No!`
(understanding-question addressed and accommodated to ISSUES)
[formally, ?und(dest-city(london)) $\in$ ISSUES]
(negative answer to understanding-question integrated; proposition dest-city(london) retracted)

S> `Not to London. So, what city do you want to go to?`
(the question of destination city is reraised)

If no response to the understanding-question is received, the system continues to assume that the interpretation was correct and eventually proceeds to the next item in its dialogue plan:

U> `To Lund please`

S> `OK, to London.`
(understanding-question now tentatively resolved but available for resolution of short answers)

U> `(silence)`
(understanding-question no longer available for resolution of short answers)

S> `What city do you want to go from?`
(system proceeds to next item on the plan)

A more detailed account of issue-based grounding in GoDiS can be found in Larsson (2002).

6 Sequencing ICM in GoDiS

We will now briefly discuss sequencing ICM in GoDiS. We believe it is good practice to try to keep the user informed about what's going on inside the system, at least to a degree that facilitates a natural dialogue where system utterances "feel

[7] Although it lies outside the scope of this paper to present the formalism used in GoDiS for representing semantic content, it has been included here for those interested. We hope the formalism is sufficiently straightforward to be understood by the reader; if not, the formal details can be ignored.

natural". One way of doing this is to produce sequencing ICM indicating significant updates to the information state which are not directly related to specific user utterances but instead concerns subactivities, topics etc.[8].

For GoDiS, we have implemented three types of sequencing ICM. First, GoDiS will indicate when it is loading a dialogue plan[9]. Second, GoDiS will produce ICM to indicate when an issue is being reraised (in contrast to being raised for the first time). Third, GoDiS generates ICM moves for indicating accommodation of questions (which may involve plan recognition). This is meant to indicate that some inference has been performed, and that it has been successful.

The choice of actual wordings for these moves are currently only based on intuitions; both corpus studies and usability tests can serve to give a firmer ground to the exact formulation of sequencing moves.

- icm:reraise: indicate reraising implicitly ("So, ...")
- icm:reraise:Q: reraising an issue Q explicitly ("Returning to the issue of price.")
- icm:loadplan ("Let's see.")
- icm:accommodate ("Alright.")

7 A sample dialogue with GoDiS

The dialogue below gives an impression of what feedback and sequencing capabilities our system handles. The basic principle for move selection in GoDiS is that utterances may not raise more than one question. Surface form generation in GoDiS is template-based. The dialogue has been annotated with ICM dialogue moves.

1 S> `Welcome to the travel agency!`

2 U> `price information please`

3 S> `Okay.` [icm:acc∗pos]
`You asked about price.` [icm:und∗pos:issue(?x.price(x))]
`Let's see.` [icm:loadplan]
`How do you want to travel?`

4 U> `pardon` [icm:per∗neg]

5 S> `How do you want to travel?`

6 U> `by plane, london`

7 S> `By plane, is that correct?` [icm:und∗chk:transport(plane)]

8 U> `yes`

9 U> `London.` [icm:sem*pos:london]
`I don't quite understand.`[icm:und∗neg]
`Do you mean to London or from London?`

10 U> `to london`

11 S> `Okay,` [icm:acc∗pos]
`to London.` [icm:und*pos:dest-city(london)]
`What city do you want to go from?`

12 U> `lyons`

13 S> `Sorry, Lyons is not a valid destination city.` [icm:acc∗neg:dept-city(lyons)]
`So,` [icm:reraise]
`What city do you want to go from?`

14 U> `paris`

15 S> `Okay,`[icm:acc∗pos]
`to Paris.` [icm:und∗pos:dept-city(paris)]
`When do you want to travel?`

16 U> `Do I need a Visa?`

17 S> `Okay,` [icm:acc∗pos]
`you want to know about visa.` [icm:und∗pos:issue(?need-visa)]
`What country are you from?`

18U> `Sweden`

19S> `Okay, from Sweden. No, you do not need a visa. Returning to the issue of price.` [icm:reraise:?x.price(x)]
`So,` [icm:reraise]
`when do you want to travel?`

...

20U> `What about connecting flights?`

21S> `Sorry, I cannot answer questions about connecting flights.` [icm:acc*neg:issue(?x.connecting-flights(x))]

8 Conclusions and future research

Starting from a fairly broad categorization of various types of feedback and sequencing moves in human-human dialogue, we have implemented a subset of these moves in a dialogue system. We believe that a view of the full range of ICM behaviour can be useful for suggesting useful additions to the range of moves handled by dialogue

[8] While able to produce sequencing ICM, GoDiS is currently not able to understand sequencing ICM from the user.

[9] Due to space limitations we will not here go into the internal workings which trigger sequencing ICM. See Larsson (2002) for detailed accounts of these.

systems, both by making systems more flexible in terms of dialogue management and by making dialogues more natural.

Our account of grounding and ICM is so far only partial in coverage; phenomena that remain to be accounted for and/or implemented include clarification ellipsis (Cooper and Ginzburg, 2001), semantic ambiguity resolution, collaborative completions and repair (Traum, 1994), and turntaking ICM. While we have included some rudimentary sequencing ICM, further investigations of the appropriateness and usefulness of these utterances are needed; here, research on discourse markers (Schiffrin, 1987) and cue phrases (Grosz and Sidner, 1986; Polanyi and Scha, 1983; Reichman-Adar, 1984) can be of use. We also want to explore turntaking in asynchronous dialogue management, and how this relates to turntaking ICM.

We are currently implementing several applications for GoDiS to enable evaluation and further development of the system. A VCR application will be reachable by phone and allow users to program and control a (computerized) VCR using spoken dialogue. We are also implementing a robot control application allowing users to interactively control and program a robot's movements using spoken dialogue.

References

Jens Allwood, Joakim Nivre, and Elisabeth Ahlsen. 1992. On the semantics and pragmatics of linguistic feedback. *Journal of Semantics*, 9:1–26.

Jens Allwood. 1995. An activity based approach to pragmatics. Technical Report (GPTL) 75, Gothenburg Papers in Theoretical Linguistics, University of Göteborg.

J. Barwise and J. Perry. 1983. *Situations and Attitudes*. The MIT Press.

H. H. Clark and E. F. Schaefer. 1989. Contributing to discourse. *Cognitive Science*, 13:259 – 94.

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

Robin Cooper and Jonathan Ginzburg. 2001. Resolving ellipsis in clarification. In *Proceedings of the 39th meeting of the Assocation for Computational Linguistics, Toulouse*, pages 236–243.

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

J. Ginzburg. forth. Questions and the semantics of dialogue. Forthcoming book, partly available from `http://www.dcs.kcl.ac.uk/staff/ginzburg/papers.html`.

B. J. Grosz and C. L. Sidner. 1986. Attention, intention, and the structure of discourse. *Computational Linguistics*, 12(3):175–204.

D. Kaplan. 1979. Dthat. In P. Cole, editor, *Syntax and Semantics v. 9, Pragmatics*, pages 221–243. Academic Press, New York.

Staffan Larsson. 2002. *Issue-based Dialogue Management*. Ph.D. thesis, Göteborg University.

Massimo Poesio and David R. Traum. 1998. Towards an axiomatization of dialogue acts. In *Proceedings of Twendial'98, 13th Twente Workshop on Language Technology: Formal Semantics and Pragmatics of Dialogue*, pages 207–222.

L. Polanyi and R. Scha. 1983. On the recursive structure of discourse. In K. Ehlich and H. van Riemsdijk, editors, *Connectedness in Sentence, Discourse and Text*, pages 141–178. Tilburg University.

R. Reichman-Adar. 1984. Extended man-machine interface. *Artificial Intelligence*, 22(2):157–218.

Ruben San-Segundo, Juan M. Montero, Juana M. Guitierrez, Ascension Gallardo, Jose D. Romeral, and Jose M. Pardo. 2001. A telephone-based railway information system for spanish: Development of a methodology for spoken dialogue design. In *Proceedings of the 2nd SIGdial Workshop on Discourse and Dialogue*, pages 140–148.

D. Schiffrin. 1987. *Discourse Markers*. Cambridge University Press, Cambridge.

David Traum and Staffan Larsson. forth. The information state approach to dialogue management. In Ronnie Smith and Jan Kuppevelt, editors, *Current and New Directions in Discourse & Dialogue*. Kluwer Academic Publishers.

D. R. Traum. 1994. *A Computational Theory of Grounding in Natural Language Conversation*. Ph.D. thesis, University of Rochester, Department of Computer Science, Rochester, NY, July.