INTERACTIVE COMMUNICATION MANAGEMENT

Coding Manual V1.0

Joakim Nivre Jens Allwood Elisabeth Ahlsén

1 Introduction

The purpose of this document is to formulate principles for the coding of *interaction communication management* (ICM), especially *feedback*, but also aspects of *turn* and *sequence management*.

2 Transcription

The first step in the preparation of data for coding is to produce a transcription which is segmented into *words* and *utterances*, and where overlapping speech is consistently and unambiguously marked. The notion of *utterance* is defined in the following way:

Definition 1 *An utterance by a speaker α is a stretch of speech produced by α, bounded by silence or by the speech of another speaker.*[1]

[1] Note that in order to allow for pauses within an utterance, a distinction must be made between *silence* (which does not belong to an utterance) and *pause* (which is considered to be part of an utterance). For the moment, we have no strict operationalization of this distinction to offer. Note also that a pause within an utterance counts as such only if it is not filled by the speech of another speaker. In the latter case, it counts instead as an utterance boundary. We are aware that this leads to a certain arbitrariness in the segmentation of utterances, but we nevertheless feel that this is the best (semi-formal) operationalization that can be achieved at present.

3 Feedback

3.1 Feedback Units

The first step in the coding consists in identifying *feedback units* (FBUs) according to the following definition (cf. [Allwood 1988a, Allwood 1988b]):

Definition 2 *A feedback unit is a maximal continuous stretch of utterance (occurring on its own or as part of a larger utterance) the primary function of which is to give and/or elicit feedback.*

Let us consider a few examples:

(1)
```
A: kommer du
B: ja
A: kan du [1 ta me en ]1 penna
B: [1 va sa du ]1
B: okej // vill du ha en egen
A: ja de vill ja
```

In this (invented) dialogue, we can distinguish four FBUs: `ja`, `va sa du`, `okej` and `ja de vill ja`. The first and third of these consist of a single word, while the other two are larger phrases. The third one (`okej`) is part of a larger utterance, while the other three constitute utterances by themselves.

3.2 Structure

After the identification of FBUs, we proceed to a *structural* classification of these units. First of all, the units are coded with respect to *grammatical categories*. For one word units, this means assigning one of the *lexical categories* in Table 1 to the word in question.[2]

The first category (*feedback word*) corresponds to the category of *primary feedback words* in [Allwood 1988a] and is exemplified by words such as `ja`, `nej`, `mm`, etc. The remaining lexical categories (when used for feedback) are collectively referred to as *secondary feedback words* in [Allwood 1988a]. In example (1), the units `ja` and `okej` are both assigned the lexical category FB, while words such as `precis` and `bra` would be coded as adverbs (ADV).

[2]The list of lexical categories is still tentative and my have to be revised, in particular when other languages than Swedish are considered.

Table 1: Lexical Categories

Lexical category	**Code**
Feedback word	`fb`
Interjection	`interj`
Noun	`noun`
Adjective	`adj`
Verb	`verb`
Preposition	`prep`
Adverb	`adv`
Proper name	`pn`
Pronoun	`pron`
Conjunction	`conj`
Complementizer	`comp`
Determiner	`det`
Auxiliary	`aux`

In feedback units consisting of more than one word, each immediate consituent is assigned a grammatical category, either a lexical one (see above) or one of the *syntactical categories* in Table 2.

Thus, in example (1), the unit `va sa du` is assigned the syntactical category S, while the unit `ja de vill ja` is segmented into `ja`, which has the lexical category FB and `de vill ja`, which has the syntactical category S. A unit such as `ja precis` is segmented into `ja` (FB) and `precis` (ADV).[3]

In addition to the coding of grammatical categories, the structural classification also involves coding for *structural operations*, which may be subdivided into *phonological*, *morphological* and *contextual* operations.[4] If several feedback words occur together and fall into distinct groups, these should be distinguished. For example, the sentence `ja just de ja` should be analyzed

[3] Major syntactic phrases such as noun phrases and sentences can in principle be further analyzed into lexical categories, but in most cases the syntactic categorization will be sufficient. As for higher level combinations, such as 'feedback word + sentence' or 'feedback word + adverb', there is no need to code the category combinations separately, since these codes can be derived automatically from the coding of the constituents.

[4] We use the term 'contextual operation' rather than 'syntactic operation', since this category includes operations, such as *repetition*, which can only be defined in relation to the preceding context.

Table 2: Syntactical Categories

Syntactical Categories	**Code**
Sentence	`s`
Noun phrase	`np`
Verb phrase	`vp`
Adjective phrase	`ap`
Adverb phrase	`advp`
Prepositional phrase	`pp`
Other	`other`

as `ja (fb)`, `just de (advp)` and `ja`.

- **Phonological operations**

  1. **Lengthening** is an operation that can be applied to any word or phrase. **Example:** `ja` $\Rightarrow$ `ja:`
  2. **Continuant reduplication** is an operation that is mainly used with primary feedback words. It comes in three varieties:
     (a) pure, e. g. `ja` $\Rightarrow$ `jaa`
     (b) with glottal stop e. g. `ja` $\Rightarrow$ `ja'a`
     (c) with glottal fricative e. g. `ja` $\Rightarrow$ `jaha`
  3. **Vowel addition** is mainly used with primary feedback units: `ja` $\Rightarrow$ `jae`
  4. **Truncation** is mainly used with primary feedback units: `ja` $\Rightarrow$ `a`
  5. **Ingressive** is mainly used with primary feedback words.
  6. **Prosodic modification** (other than lengthening) occurs with all kinds of words and phrases.

- **Morphological operations**

  1. **Reduplication** occurs mainly with primary feedback words: `ja` $\Rightarrow$ `jaja`

Table 3: Tags for Structural Operations

Tags	**Values**
`phon_op`	`lengthening`
	`cont_redupl(pure)`
	`cont_redupl(fricative)`
	`cont_redupl(stop)`
	`vowel_addition`
	`truncation(pure)`
	`ingressive`
	`prosody`
`morph_op`	`reduplication`
	`derivation`
	`compounding`
	`reduction`
`context_op`	`repetition`
	`reformulation`

2. **Inflection/derivation**:
   `hej` ⇒ `hejsan`
3. **Compounding**:
   `ja + då` ⇒ `jadå`
4. **Reduction**:
   `jaha` ⇒ `ha`

- **Contextual operations**

  1. **Repetition** (verbatim) of some part of the immediately preceding utterance:
     `A: har du en penna`
     `B:` <u>`penna`</u>
  2. **Reformulation**, i. e. reformulation by means of deictic and anaphoric expressions of some part of the immediately preceding utterance:
     `A: har du en penna`
     `B: ja` <u>`de har ja`</u>

Note that it is often necessary to assign several codes to the same segment. For example, in a feedback unit such as `jaa de gör ja`, the segment `jaa` is assigned the codes `lexcat=fb` and `phon_op=cont_redupl(pure)`, while the segment `de gör ja` is coded `syncat=s` and `context_op=reformulation`.

3.3 Position and Status

The coding of *position and status* concerns the position of an entire FBU within a larger utterance. Thus, the coding of *position* indicates whether the FBU is

- *single* (i. e. constitutes an entire utterance by itself),
- *initial* in an utterance,
- *medial* in an utterance,
- *final* in an utterance.

In example (1), the first two FBUs (`ja` and `va sa du`), as well as the last one (`ja de vill ja`), are *single*, while the third one (`okej`) is *initial.*

3.4 Function

Two aspects of the *function* of FBUs are coded:

1. Function type
2. Attitudes

The latter aspect applies mainly to *feedback givers* (see below).

Function type By *function type* is meant a broad classification of feedback functions into:

- giving feedback (`give`),
- eliciting feedback (`elic`),
- giving *and* eliciting feedback (`give_elic`).

In example (1), `ja`, `okej` and `ja de vill ja` are givers, while `va sa du` is a giver-elicitor.

Attitudes The coding of attitudes can be broken down in three parts:

- CPU attitudes
- Acceptance attitudes
- Other attitudes

The term *CPU attitudes* is used to refer to the attitudes of *contact* (`cont`), *perception* (`perc`), and *understanding* (`und`) (cf. [Allwood 1988a]). These basic communicative functions are normally coded only when "marked", i. e. when they are *negative* and/or *explicit* (as opposed to the "unmarked" case which is *positive* and *implicit*). In our example, only the unit `va sa du` needs to be coded for negative perception (`perc(neg)`) and understanding (`und(neg)`).

By *acceptance attitudes* we mean the attitudes of acceptance or non-acceptance, occurring especially after communicative acts such as statements, questions, requests, and offers (cf. [Allwood, Nivre & Ahlsén 1992]). Primarily, these attitudes concern the main evocative intention of communicative acts, ie, accepting a statement as correct and worthy of belief, accepting to answer a question or accepting to carry out a request. Secondarily, the attitudes can concern acceptance of a turn and/or a communicative act. Since these two secondary functions closely correspond to the CPU attitudes *acceptance of (contact)* and *acceptance of communicative* (perception or understanding of communicative act), they are not separately coded. Acceptance attitudes can be coded as `positive` (acceptance), `negative` (non-acceptance) or `not relevant/applicable` . In addition to CPU attitudes and acceptance attitudes, such attitudes as *surprise*, *anger*, *happiness*, *sadness*, etc. are coded when relevant.

Finally, expressive features which go beyond CPU attitudes should be coded. Examples of such expressive features are epistemic, emotional and discursive attitudes, as well as some speech act like functions. The following list gives some examples: `hesitation, uncertainty, surprise, reminder, anger, happiness, sadness, contempt, friendliness, irony, support, polite, concession, admission, objection`.

`No Label` means that no specific expressive feature has been noted, ie, that the attitude is neutral or too difficult to decide on. No attempt is made to clearly distinguish attitude labels from speech act labels since many terms can be used in both senses, e.g *support* and *object*.

Table 4: Tags for Attitudes

Tag	**Value**
`contact`	`pos` `neg`
`perc`	`pos` `neg`
`und`	`pos` `neg`
`acc_evoc_function`	`pos` `neg`

`other_expr_features`	`surprise`
	`anger`
	`happiness`
	`sadness`
	`contempt`
	`prosodic features`
	`...`

3.5 Some problematic cases

Primary feedback words The following Swedish words count as primary feedback words:

`ja`
`jo`
`nej`
`nä`
`nja`
`m`
`okej`
`va`

Nja should also be regarded as a primary feedback word, where the speaker is doubtful about the content of the previous utterance. In such a case, `acceptance` should be marked as `negative` and `other expressive features: doubt` should also be coded. `Jaha` is counted as a primary derived feedback word.

Other feedback words For all the words, the other attitudes that have been suggested are default values, which can be invalidated by context.

`naä` occurs in contexts where the speaker is sceptical and not really agrees with the previous speaker. For example: `naä de menar ja inte`. `Naä` should be coded with `content negative, other attitude: hesitation, truncation`, and `vowel addition`.

`naäj` see `naä` above

`ne` variant of `nä` and should be coded the same way

`na` `acceptance:negative, other attitude: hesitation`

`tja` should not be coded as negative or positive concerning content. `Other attitude:hesitation` should be marked.

Pauses Whether pauses should be regarded as part of an utterance or not, is to be decided on the basis of the context. This is relevant in cases where feedback utterances end with a pause, as in the following example:

A: mm //

Here we have decided not to consider the pause as part of the utterance when coding the position and status of the feedback segment. `mm` will be coded as `position:single`.

Hesitation sounds When coding feedback, don't pay any notice to hesitation sounds like *e1* etc:
A: just e1 precis
This entire phrase should just be coded as an advp.

Words and phrases of Greeting

- ***Hej* and *Hej då***

  `Hej` should be coded as feedback.[5] This word is regarded as feedback giving as well as -eliciting, depending on the situation. If there are two

[5]One could possibly think of some case where it doesn't work as a greeting and is not feedback, but we haven't seen such cases yet.

persons greeting each other, the first `hej` should be coded as `elicit`, and the response-`hej` as `give`. `Hej` belongs to the lexical category `Interjektion` and `Hej då` should be coded as `Interjp`. Other than that, `hej då` should be coded like `hej`.

- **Välkommen** should be considered as feedback-eliciting, and should thus be coded as elicit. The word is an interjection.
- **Var så god** - should be coded as Interjp.

4 Turn and Sequence Management

Turn and sequence management are coded in three different categories:

1. Turn characteristics (opening, holding, closing)
2. Overlap functions
3. Sequential marking

It can be noted that these categories are partly overlapping in the sense that the same phenomena may be coded in more than one category though from different perspectives.

4.1 Turn Characteristics

Coding turn characteristics presupposes that utterances are divided into smaller parts exemplifying the characteristic in question.

1. Turn opening
   - Explicit turn accceptance (if any). This usually occurs in the beginning of an utterance.
2. Turn holding
   - Turn holding mechanisms (if any), such as *pause, phrase, morpheme, lengthening, repetition, reduplication.* This usually occurs medially in an utterance but can also occur initially or finally.
3. Turn closing

  - Explicit turn allocation + addressee (if any).
  - Final maximal phrase (syntactic category).

   Turn closings usually appear at the end of an utterance.

4. Turn taking

   - The utterance serves to take over the turn.

4.2 Overlap Function

Coding *turn characteristics* like coding *overlap* presupposes dividing utterances into smaller parts, since very often a part of an utterance is overlapped. Instances of overlap are coded with the following two functional categories:

- Interruption. Here we distinguished the interrupted utterance from the interrupting utterance.

- Other function. Overlaps can occur for many other reasons than being part of an interruption. Some common other functions are the following: *Giving attention, affirmation, acceptance, reaffirmation, reminder, excuse, continuation, hesitation, disagreement.*

4.3 Sequential Marking

Sequential marking in dialog occurs on several levels. Three important levels are
(i) the interaction as a whole
(ii) a particular subactivity, and
(iii) a particular exchange.
In our coding we distinguish two cases:
(i) Opening, continuing and closing the interaction as a whole or a subactivity from
(ii) opening, continuing and closing a particular exchange type like a question-answer sequence. Again codes may apply only to different parts of an utterance. The codes are the following:

- Opening

  - Interaction as a whole, subactivity

  - exchange

- Continuing
  - Interaction as a whole, subactivity
  - exchange
- Closing
  - Interaction as a whole, subactivity
  - exchange

References

[Allwood 1988a] Allwood, J. (ed) (1988) Feedback in Adult Language Acquisition. Final Report II. Ecology of Adult Language Acquisition (ESF).

[Allwood 1988b] Allwood, J. (1988) Om det svenska systemet för språklig återkoppling. In Linell, P., Adelswärd, V., Nilsson, T. and Pettersson, P. A. (eds) *Svenskans beskrivning* 16. SIC 21a. University of Linköping: Department of Communication Studies.

[Allwood, Nivre & Ahlsén 1992] Allwood, J., Nivre, J. and Ahlsén, E. (1992) On the Semantics and Pragmatics of Linguistic Feedback. *Journal of Semantics* 9, 1–26.

[Lager 1992] Lager, T. (1992) TagLog—the manual. University of Göteborg: Department of Linguistics.

[Hagman & Nivre 1993] Hagman, J. and Nivre, J. (1993) Transcription Standards. University of Göteborg: Department of Linguistics.