Interpretation and Generation of Dialogue with Multidimensional Context Models

Harry Bunt

Tilburg University
harry.bunt@uvt.nl

Abstract. This paper presents a context-based approach to the analysis and computational modeling of communicative behaviour in dialogue. This approach, known as Dynamic Interpretation Theory (DIT), claims that dialogue behaviour is multifunctional, i.e. functional segments of speech and nonverbal behaviour have more than one communicative function. A 10-dimensional taxonomy of communicative functions has been developed, which has been applied successfully by human annotators and by computer programs in the analysis of spoken and multimodal dialogue; which can be used for the functional markup of ECA behaviour; and which forms the basis of an ISO standard for dialogue act annotation. An analysis of the types of information involved in each of the dimensions leads to a design of compartmented, 'multidimensional' context models, which have been used for multimodal dialogue management and in a computational model of grounding.

Keywords: Multidimensional context modeling, dialogue acts, dialogue semantics, dialogue generation, dialogue act annotation.

1 Introduction

A context-aware dialogue system should base both the determination of its own actions and its interpretation of the user's behaviour on a model of the current context. Every dialogue system is context-aware to some extent, since its operation depends on the dialogue history. For instance, a simple frame-based system for providing public transport information may conduct a dialogue by systematically acquiring a number of parameter values such as the specification of a destination, of a departure place, of a travel date, and an approximate arrival or departure time, and subsequently providing information which results from a data base query with these parameter values. Such a system bases its actions on the current state of the frame, in particular on which parameter values have not yet been found, and interprets user inputs in terms of the parameter values that it is looking for. Such a system has a very limited context-awareness, due to the limited interpretation of ‘context' as a set of task-specific parameter values.

In general, the term ‘context' refers to the surroundings, circumstances, environment, background or settings of the activity of which the context is considered. In linguistics the term `context' has most often been interpreted as referring to the surrounding text. The context of a spoken dialogue utterance is also often understood in

A. Esposito et al. (Eds.): COST 2102 Int. Training School 2010, LNCS 6456, pp. 214–242, 2011.

this sense, in particular with reference to the preceding dialogue, but the broader interpretation in terms of circumstances, background and settings is important as well, and includes in particular the following:

(1) a. the type of interaction;
b. the task or activity that motivates the dialogue;
c. the domain of discourse;
d. the physical or perceptual conditions;
e. the information available to the dialogue participants.

Each of these notions of context occur in particular senses of the word `context' in English, as illustrated below:

(2) a. *in the context of a human-computer dialogue*
b. *in the context of a doctor-patient interview*
c. *in the context of philosophical discourse*
d. *in the context of telephone conversation*
e. *in the context of hardly any shared assumptions*

Each of these notions of context is relevant for the interpretation and generation of dialogue behaviour. Whether one is (a) participating in a dialogue with a computer or in a dialogue with another person may make a great difference, for example, for the interpretation and generation of ‘social' dialogue acts like greeting, thanking, apologizing, and saying goodbye. When a human dialogue partner wishes one a good day, at the end of a dialogue, it may be quite appropriate to return the wish, but it seems nonsense to wish a machine a good day[1]. Whether (b) one is being interviewed by a doctor, rather than having a chat with the neighbours, may make a great difference for how much one will say about one's health problems. Similar considerations apply with respect to the other notions of ‘context' in (1) and (2).

A definition of context which encompasses all these uses and which is appropriate for the study of dialogue and the design of dialogue systems, is the following:

(3) Context in dialogue is the totality of conditions which influence the interpretation or generation of utterances in dialogue [9].

This definition is rather too broad to be effectively useful. In order to arrive at a more manageable notion of context, we note that, according to this definition, a dialogue context has a proper part formed by those elements that can be influenced by dialogue. Whether one is talking to a computer or to a person (see 2a) is for example typically a permanent feature of a dialogue context, which is not changed by the dialogue. (Although this may happen: the occurrence of a persistent problem in communicating with a computer may cause the user to get connected to a human operator.) Or also, (2b) whether a dialogue takes place in the context of a patient seeing a doctor, or (2c), whether one participates in a philosophical discourse or in a dialogue aiming to know the departure time of a particular train, is not something that changes during a dialogue.

[1] We have witnessed inexperienced users of a spoken dialogue system do so, and subsequently be extremely annoyed at their own behaviour.

The consideration of which context information may be changed by a dialogue and which information cannot, leads to the distinction between *local* and *global* context. The global context is formed by the information that is not changed by the dialogue, and is often important in a global sense, having an influence on overall speaking style and use of interaction strategies. The local context contains the information which is changed when an utterance is understood, and is therefore crucial for the semantic interpretation of dialogue utterances.

(4) **Local context** is the totality of conditions which may be changed through the interpretation of dialogue utterances.

Local context information is typically more complex and fine-grained than global context information and requires an articulate form of representation.

In this paper we analyse the question which kinds of information need to be represented in the context model of a sophisticated natural language based dialogue system, taking a context-based approach to the understanding and generation of utterances in spoken and multimodal dialogue. More specifically, we view the meaning of a dialogue utterance in terms of how the context model of a dialogue participant is changed when he understands the utterance as encoding multiple dialogue acts. We summarize the theoretical framework of Dynamic Interpretation Theory (DIT), in which this view has been elaborated, and discuss its consequences for the content and structure of context models for dialogue interpretation and generation. We show how such a context model can be represented by means of typed feature structures, and show how such models can be applied in the study and computational modeling of a range of aspects of spoken and multimodal communication.

2 Context and Dialogue Interpretation in DIT

2.1 Dialogue Semantics

The framework of Dynamic Interpretation Theory owes its name to the observation that communicative behaviour is best understood in a dynamic way, in terms of communicative actions called dialogue acts, which are directed at one or more addressees and which describe what the speaker is trying to achieve as the result of his action being understood by the addressee(s). Upon understanding a dialogue act performed by a speaker (S), an addressee (A) forms certain beliefs about S's intentions and assumptions and other aspects of S's mental state. Consider example (5), showing the effects on A's knowledge about S that occur when A understands a *Check Question* addressed to him.

(5) S, addressing A: *This is the two-forty to Naples, right?*
(q = this is the two-forty to Naples)
1. A assumes that S wants to know whether q;
2. A assumes that S assumes that A knows whether q;
3. A assumes that S has a weak belief that q.

Note that the latter condition in (5) distinguishes a *Check Question* from a yes-no question. More generally, each type of dialogue act corresponds to a particular effect on an understander's state of information. In DIT, such states of information are called *contexts*, and the description of how context changes capture the understanding of communicative behaviour is called a *context-change semantics* [9]. This approach to utterance meaning is also known as the 'information state update' or `ISU' approach [48].

The application of a context-change approach to utterance meaning can only be successful if the context models that are used contain the kinds of information that can be changed by a dialogue act. For example, the semantic interpretation of a turn-grabbing action, whereby a dialogue participant A tries to take the speaker role from another participant S who currently occupies that role, should involve that S understands that A wants to occupy the speaker role; this can be accommodated in a context-change semantics only if a context model is used which contains information about the allocation of the sender role. (And by the same token, in the case of a multi-party conversation, the understanding of whom the current speaker is addressing requires the modeling of the distribution of the addressee role and other participant roles among the participants in the conversation.) Similarly, the semantic interpretation of a *Stalling* act (*Let me see...)* requires a context model which includes a representation of the estimated time needed by a speaker to construct his next contribution to the dialogue.

More generally, by inspecting the semantic interpretation of the kinds of things that participants in a dialogue say and signal (possibly nonverbally, or partly through language and partly nonverbally) in terms of changes in the context model of an understanding agent, we can obtain a catalogue of the kinds of information that a dialogue context model should include. A crucial step in such a process is constructing a catalogue of “kinds of things that participants in a dialogue say and signal". One of the products of the research in which the DIT framework has been developed is such a catalogue, through the definition of a comprehensive taxonomy of types of dialogue acts, called the DIT^{++} taxonomy[2] (This taxonomy has been designed to include in a systematic and consistent fashion besides the dialogue acts in the original DIT taxonomy [6] also a number of act types from other schemes, such as DAMSL, MRDA, and SLSA, with the aim to define a domain-independent schema for the functional annotation and semantic analysis of multimodal dialogue.

The DIT framework was developed to support the analysis of human dialogue, as well as the design of computer dialogue systems, in particular for system components which are usually called *Dialogue Managers*. Dialogue management is primarily a process of deciding what next to do in a dialogue. The question of what kinds of information should be included in a context model can be approached not only from the perspective of *understanding* dialogue behaviour, but equally from that of the *generation* of dialogue contributions. A Dialogue Manager's task of deciding how to continue a dialogue at a given point, can be formulated as making a choice from the possible things to say, which is a task that presupposes (again) some kind of catalogue of the kinds of things that could possibly and sensibly be said at that point. Or rather, that could be said *given the state of the context that has been brought about by the dialogue up to that point.*

[2] See `http://dit.uvt.nl`

For example, suppose the user has asked the system *Which flights from London are expected this evening after six, and which ones tomorrow morning before twelve?,* to which the answer needs to be looked up by going through a list with tonight's expected arrivals and through the flight schedule of the next morning, a dialogue system could decide that it needs a few seconds to collect this information and that it would therefore be appropriate to perform a *Stalling* act (like *Let me see.., London, this evening,...*) or a *Pausing* act (*Just a minute*). The Dialogue Manager will be able to generate such 'time management acts' only if has an awareness of the time needed by its information processing, in other words, if its context model includes information about the estimated time needed by this processing. More generally, a Dialogue Manager can only generate a given dialogue act if the conditions motivating the performance of that act are represented in its current context model; hence the specification of the functionality of a Dialogue Manager goes hand in hand with a specification of the kinds of information to be represented in its context model. A sophisticated dialogue system would be expected by its users to be able to generate the same classes of dialogue acts as the ones that it can understand, hence the generation perspective on context model requirements and the utterance understanding perspective are two sides of the same medal.

The DIT framework seems especially fruitful for studying the contents of context models because of the detailed taxonomy of dialogue acts that it has defined, which supports a view on communication as consisting of multiple layers of activity, such as the layer of pursuing a particular task or goal, that of taking turns, that of providing feedback, and that of managing the use of time. This multidimensional view posits that dialogue participants are often simultaneously engaged in communicative activities in several of these layers, and that their communicative behaviour is therefore often *multifunctional*. In the rest of this section we summarize the DIT view on multifunctionality in dialogue.

2.2 Multifunctionality and Multidimensionality

Studies of human dialogue indicate that natural dialogue utterances are very often multifunctional. This is due to the fact that participation in a dialogue involves several activities beyond those strictly related to performing the task or activity for which the dialogue is instrumental. As noted by Allwood in [4], in natural conversation, among other things, a participant constantly *"evaluates whether and how he can (and/or wishes to) continue, perceive, understand and react to each other's intentions"*. They share information about the processing of each other's messages, elicit feedback, and manage the use of time and turn allocation, of contact and attention, and of various other aspects. Communication is thus a complex, multi-faceted activity, and for this reason dialogue utterances are often multifunctional. An analysis of this phenomenon [14], [15] shows that functional segments in spoken dialogue on average have 3.6 communicative functions, when functional segments are defined as follows:

(6) A **functional segment** is a minimal stretch of behaviour that has a communicative function (and possibly more than one).

Multidimensional taxonomies support dialogue utterances to be coded with multiple tags and have a relatively large tag set; see e.g. [3], [12], [13], [32], [45]. A large tag set may benefit in several respects from having some internal structure, in particular, a taxonomical structure based on semantic clustering can be searched more systematically and more 'semantically' than an unstructured one, and this can be advantageous for dialogue annotation, interpretation, and generation.

Bunt [12] suggests that a theoretically grounded multidimensional schema should be based on a theoretically grounded notion of dimension, and proposes to define a *set of dimensions* as follows.

(7) Each member of a set of dimensions is a cluster of communicative functions which all address a certain aspect of participating in dialogue, such that:

1. dialogue participants can address this aspect through linguistic and/or nonverbal behaviour;
2. this aspect of participating in a dialogue can be addressed independently of the other aspects corresponding to elements in the set of dimensions, i.e., an utterance can have a communicative function in one dimension, independent of its functions in other dimensions.

The first of these conditions means that only aspects of communication are considered that can be distinguished according to empirically observable behaviour in dialogue. The second condition requires dimensions to be independent, 'orthogonal'.

Petukhova and Bunt in [40], [41] present test results based on co-occurrence frequencies, phi-statistics, and vectorial distance measures to empirically determine to what extent the dimensions that are found in 18 existing annotation schemas are well-founded. A conclusion from this study is that the 10 dimensions of the DIT^{++} taxonomy, described below, form a well-founded set of dimensions.

2.3 Dimensions

The ten dimensions of DIT^{++} have emerged from an effort to provide a semantics for dialogue utterances across a range of dialogue corpora. Utterances have been identified whose purpose was to address the following aspects of participating in a dialogue:

(8) 1. advancing a task or activity motivating the dialogue;
2. monitoring of contact and attention;
3. feedback on understanding and other aspects of processing dialogue utterances;
4. the allocation of the speaker role;
5. the timing of contributing to the dialogue;
6. structuring the dialogue and monitoring the progression of topics;
7. editing of one's own and one's partner's contributions;
8. the management of social obligations.

Whether these aspects qualify as proper dimensions can be determined by checking them against definition (7). Take for instance the timing of contributions. Utterances that address this aspect of interacting include those where the speaker wants to gain a little time in order to determine how to continue the dialogue; this function is called

Stalling. Speakers indicate this function by slowing down in their speech and using fillers, as in *Ehm, well, you know,...* The observation that dialogue participants exhibit such behaviour means that the category of functions addressing the timing of contributions (which also includes the act of *Pausing*, realized by means of utterances like *Just a minute, Hold on a second*) satisfies criterion (7.1). Moreover, the devices used to indicate the *Stalling* function can be applied to virtually any kind of utterance, which may have any other function in any other dimension. Timing therefore satisfies criterion (7.2) as well, and hence qualifies as a proper dimension.

A similar analysis can be applied to the other aspects. Of these, the feedback category should be divided into two, depending on whether a speaker gives feedback on his own processing, or whether he gives or elicits feedback on the addressee's processing; we call these dimensions 'Auto-Feedback' and 'Allo-Feedback', respectively [7]. Examples of auto- and allo-feedback are:

(9) Auto-feedback: *Okay, right, m-hm; What?*, nodding, smiling; frowning
Allo-ffedback: *Okay? all right?; Nonono, Hoho, Wait a minute!*

Similarly, the category of dialogue acts concerned with editing one's own or one's partner's contributions, is better split into those concerned with editing one's own speech, called the Own Communication Management (OCM) dimension (using Allwood's terminology [5], and those concerned with the correction or completion of what the current speaker is saying, which by analogy we call the Partner Communication Management (PCM) dimension. Examples of OCM and PCM acts are (10) and (11), respectively. In (10) the speaker corrects himself; in (11) B corrects A's first utterance (PCM), which is subsequently acknowledged by A.

(10) A: *then we'e going to g-- turn straight back*

(11) A: *back to Avon, drop-*
B: *pick up the oranges*
A: *sorry, pick up the oranges*

Dialogue acts with a dimension-specific function are often performed partly or entirely nonverbally, such as positive feedback by nodding, negative feedback by frowning, or turn assignment by direction of gaze. A study by Petukhova [38], performed in the context of the EU project AMI[3], showed that all the communicative functions of the nonverbal behaviour of participants in AMI meetings could be described adequately in terms of the DIT^{++} functions, and produced a catalogue of nonverbal means (notably head gestures, facial expressions, and gaze behaviour) for expressing DIT^{++} communicative functions, either by themselves or in combination with verbal behaviour.

All in all, this had lead to the distinction of the following 10 dimensions in the DIT^{++} taxonomy:

[3] Augmented Multi-party Interaction; see `http://www.amiproject.org`

1. Task/Activity: dialogue acts whose performance contributes to advancing the task or activity underlying the dialogue;
2. Auto-Feedback: dialogue acts that provide information about the speaker's processing of previous utterances. Feedback is 'positive' if the processing was successful at the level that is addressed; negative feedback, by contrast, signals a processing problem;
3. Allo-Feedback: dialogue acts used by the speaker to express opinions about the addressee's processing of previous utterances, or to solicit information about that processing;
4. Contact Management: dialogue acts for establishing and maintaining contact, such as *Hello?*;
5. Turn Management: dialogue acts concerned with grabbing, keeping, giving, or accepting the sender role;
6. Time Management: dialogue acts which signal that the speaker needs a little time to formulate his contribution to the dialogue, or that the interaction has to be suspended for a while;
7. Discourse Structuring: dialogue acts for explicitly structuring the conversation, e.g. announcing the next type of dialogue act, or proposing a change of topic;
8. Own Communication Management: dialogue acts for editing the contribution to the dialogue that the speaker is currently producing;
9. Partner Communication Management: the agent who performs these dialogue acts does not have the speaker role, but assists or corrects the dialogue partner who does have the speaker role in his formulation of a contribution to the dialogue;
10. Social Obligations Management: dialogue acts that take care of social conventions such as welcome greetings, apologies in case of mistakes or inability to help the dialogue partner, and farewell greetings.

Positive and negative feedback acts do not necessarily express only success or problems encountered in processing previous utterances, but may additionally express an *attitude* such as happiness, surprise or regret. For dealing with this phenomenon, DIT makes use of a set of *qualifiers* which can be attached to communicative functions of a responsive nature, indicating a particular emotion or attitude towards the content that is discussed or towards the addressee (see [42]), as illustrated in example (13).

(13) 1. A: Could you please give me the details of that connection?
2. B: That's flight NY 607, departure eleven twenty a.m., arrival one forty p.m.
3. A: Perfect! Thanks!

In this example, A's utterance *Perfect!* signals not only that A has processed B's utterance successfully, but also that A is pleased with the information that he received. Using the qualifier attribute `sentiment` with the value `pleased`, this can be represented in the DiAML annotation language[4] as follows:

[4] DiAML: Dialogue Act Markup Language, as defined in ISO standard 24617-2 [29].

(14)
```
<dialogueAct xml_id="da1" target="#fs3.1"/>
    sender="#a" addressee="#b"
    communicativeFunction="autoPositive"
    dimension="autoFeedback"
    sentiment="pleased"
  </dialogueAct>
```

3 Communicative Functions

Some communicative functions are specific for a particular dimension; for instance *Turn Accept* and *Turn Release* are specific for turn management; and *Stalling* and *Pausing* are specific for time management. Other functions can be applied in any dimension; for instance a *Check Question* can be used with task-related semantic content in the Task dimension, but can also be used for checking correct understanding (feedback). In general, all types of question, statement, and answer can be used in any dimension, and the same is true for commissive and directive functions, such as *Offer, Suggest*, and *Request*. These communicative functions are therefore called *general-purpose* functions, as opposed to *dimension-specific* functions. The use of *Question* acts in different dimensions is illustrated in (15)

(15) 1. Are you open this Sunday. [*Task*]

2. Did you say Thursday? [*Auto-Feedback*]
3. Did you hear what I said. [*Allo-Feedback*]
4. Anyone have anything more to add? [*Discourse Structuring*]
5. Can you give me a moment to check this? [*Time Management*]
6. Peter, would you mind to continue? [*Turn Management*]

The DIT^{++} taxonomy of communicative functions therefore consists of two parts:

1. a set of clusters of **general-purpose functions**;
2. a set of clusters of **dimension-specific functions**.

Figure 1 shows the taxonomy of general-purpose functions. This taxonomy falls apart into four hierarchies:

- *information seeking* and *information providing* functions, which seek or provide information, respectively, and which together form the class of *Information Transfer* functions;
- *commissive and directive* functions, where the speaker commits himself to performing an action, or puts pressure on the addressee to perform or participate in an action; together, these form the class of *Action Discussion*

functions, which bring an action into the discussion that may or should be performed by the speaker, by the addressee, or jointly.

In each of these hierarchies, a mother - child relation between two communicative functions means that the child function is a special case of the mother function, while siblings are alternative, mutually exclusive specializations. The hierarchical structure of the set of general-purpose functions can effectively be used as a decision tree for deciding which communicative function applies to a given dialogue segment.

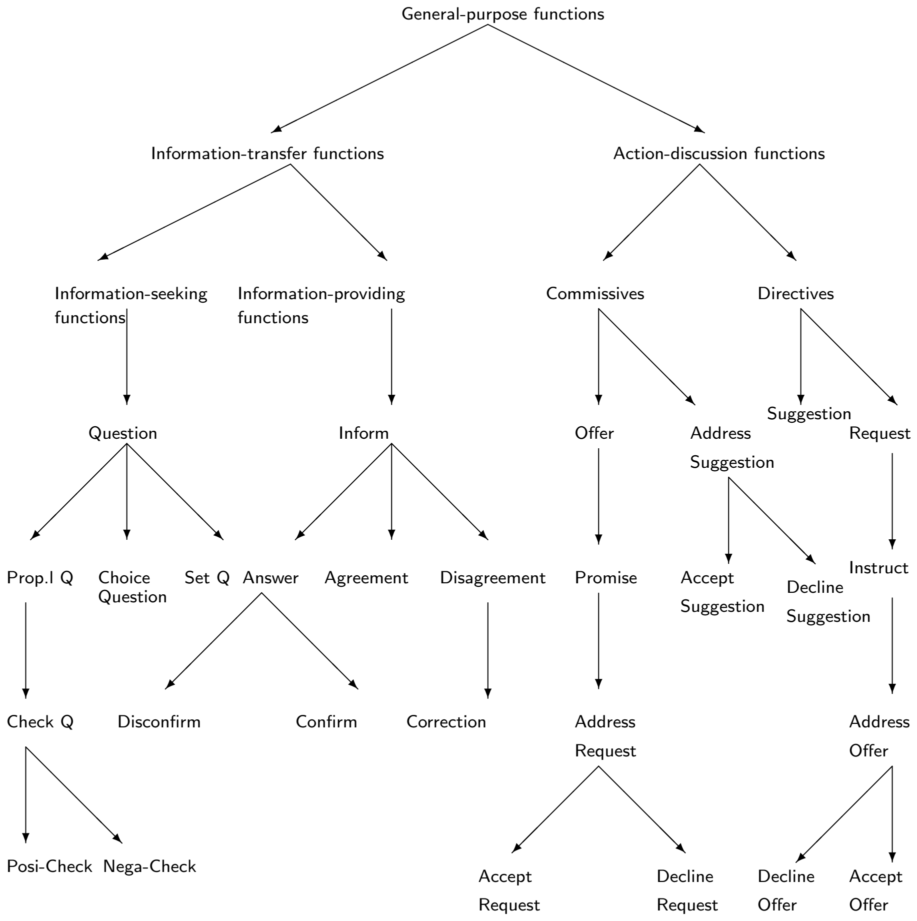

Fig. 1. DIT^{++} general-purpose functions

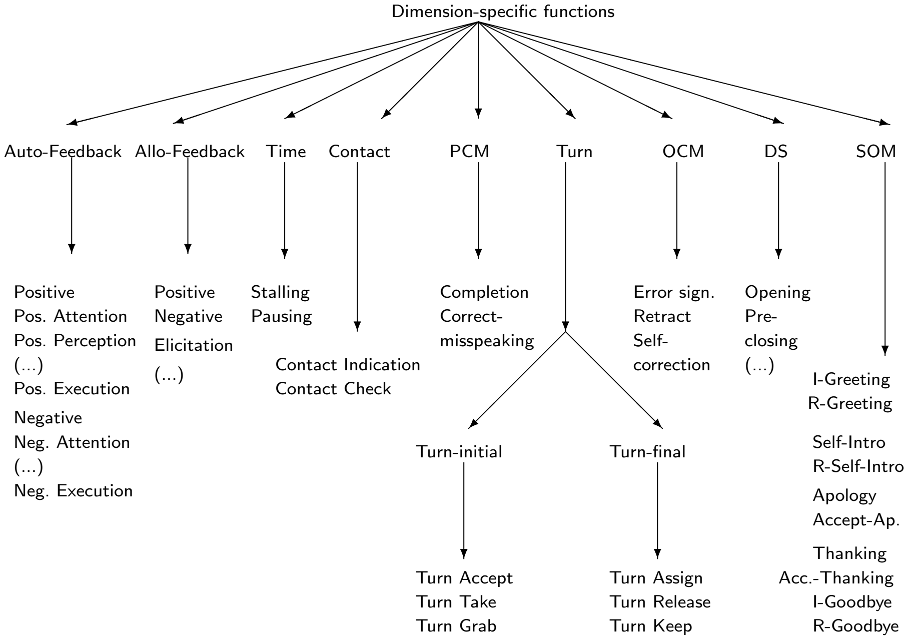

Fig. 2. DIT^{++} dimension-specific functions

4 Content and Structure of a Dialogue Context Model

Considering the content and structure of a context model, it should be kept in mind that a context model by definition is always the context model *of a particular dialogue participant*, since the generation and interpretation of dialogue acts is a separate process for each participant, and it can only be influenced by information about the context as viewed by that participant.

4.1 A Catalogue of Context Information

A well-defined set of dimensions, like those listed in (12), can help us to obtain a catalogue of kinds of information that an adequate dialogue context model should contain, since the communicative activities in each of these dimensions address a different type of information. In this section we examine each of these types of information.

a. Task/Actvity
The dialogue acts in the Task/Activity dimension have information concerning the underlying task or activity as their semantic content, and in order to provide a basis for interpreting and generating such dialogue acts, every context model needs to incorporate this kind of information. We have seen in Section 1 that a very simple, frame-based dialogue system employs a primitive form of this type of information, represented by slots in a frame where certain parameter values can be filled in. More sophisticated dialogue systems, that can accept and generate a variety of dialogue act

types, need a more sophisticated representation of task-related information; in particular, they need to take into account that some task-related information is available only to the participant whose context model is considered, some is not available to him but assumed to be available to one or more other participants, and still other information is assumed to be shared. These distinctions are of crucial importance if a dialogue system is to maintain a consistent context model while taking into account that a participant may contribute information that is incompatible with the information in the context model under consideration.

The importance of this point may be appreciated by considering the following example.

(16) 1. A: So the deadline for IWCS passed yesterday, but I didn't submit this time; I wasn't happy with what I'd written so far.
2. B: But the deadline has been extended, didn't you see?

In this example participant A performs an *Inform* act which provides participant B with certain information; B has reason to believe that A is partly wrong, and expresses that in the second utterance. This illustrates the fact that understanding an *Inform* act with semantic content *X,* performed by speaker A, does not mean for an addressee B that *X* is the case, which would be inconsistent with what B knows, but that *A believes that X*. And such a belief is obviously compatible with B believing that *X* is not the case. More generally, this shows that the representation of task-related information in a context model must be relative to a participant (or to be shared by multiple participants).

An important phenomenon in dialogue is that of establishing a *common ground*, information that each participant assumes to be shared with the other participants. This is for example relevant in the case of a disagreement, like (16); if the participants try to resolve the disagreement, and reach a stage where they believe they have succeeded in doing so, this should be represented in the context model; for this purpose a representation of *assumed common ground* is needed[5].

In sum, the representation of task-related information in the context model of a dialogue participant requires the representation of:

(17) a. information about the task
b. assumptions about the other participants' information about the task
c. assumptions about the task-related information that is shared with other participants.

b. Processing Information

b1. Own processing

The dialogue acts in the Auto-Feedback dimension have as their semantic content information about the speaker's own processing of previous dialogue utterances, most often of the last utterance by the previous speaker.

DIT distinguishes five levels of understanding that participants in a dialogue may reach in processing each other's utterances. The lowest level is that of paying *attention*. In a human-computer dialogue this hardly seems a relevant level to take into

[5] See [18] for a DIT-based computational model of establishing common ground.

consideration, but it is is quite relevant in human dialogue, especially in multi-party dialogue.

The next higher level, that of *perception*, is particularly important in spoken dialogue systems, where it is associated with problems in speech recognition. Because current ASR systems are far from perfect, there is a tendency in spoken dialogue systems to provide feedback all the time in order to let the user know what the system thinks the speaker said.

The next higher level, that of understanding in the sense of semantic and pragmatic *interpretation*, corresponds to identifying the speaker's intentions, i.e., to recognizing the communicative functions and semantic contents of his dialogue acts, and therefore determining the appropriate context update to perform.

The *evaluation* level is concerned with examining whether the context update, constructed by the successful understanding of a functional dialogue segment, is checked for leading to a consistent state of the context model. If the processing at this level is successful, then the participant's context model is indeed updated. For instance, evaluating a question is deciding whether it can be considered as a genuine request for information, which is worth answering; similarly, evaluating an answer is deciding whether the sender indeed may be assumed to provide information which he (the sender) assumes to be correct.

Reaching the *execution* level means being able to do something with the result of the evaluation. For example, in the case of a question, it consists of finding the requested information; in the case of an answer, successful 'execution' is integrating its content with the context model, leading to a new, consistent state of the model. If an inconsistent state would result, then the integration does not go through, and the execution of the answer fails. Similarly, unsuccessful execution of a question means that the requested information is not found (leading to responses like *I don't know*.)

At each of these levels of processing, a participant may encounter difficulties:

- At the level of attention the problem that may occur is that a participant did not pay attention and therefore did not hear what was said.
- At the level of perception, not only a machine but also a human participant may encounter speech recognition difficulties, e.g. because the speaker spoke unclearly or softly, or because some acoustic disturbance corrupted the communication channel.
- At understanding/interpretation level the participant may be unsuccessful in establishing the communicative function(s) of what was said (e.g. *Is this a question or a statement?)* or may have problems in determining the precise semantic content.
- At evaluation level a difficulty arises when the participant concludes that the speaker said something that is inconsistent with what he said before, or more generally with information that the participant assumed to be shared, for example when an utterance was interpreted as a question which the speaker believes has already been answered. The difficulty is that the context update, determined at the interpretation level, cannot be performed without making the participant's context model inconsistent.
- At execution level a difficulty arises when the participant is unable to perform a certain action which he wants to perform as a result of the other levels

of input processing being successful. For example, the participant has processed an instruction to perform a certain action successfully up to this level, but when actually executing the requested action he runs into an obstacle that prevents him from completing the action.

b2. Partner's processing

In allo-feedback acts the speaker signals his views on the success of the addressee's processing of previous utterances, or he elicits information about that, being uncertain whether the addressee paid attention, correctly heard or understood what was said, or successfully evalutated or executed that. The same levels of processing apply as in the case of auto-feedback.

c. Establishing and Checking Contact

Dialogue participants who cannot see each other have to make their presence and readiness to communicate explicit. Explicit acts for establishing, checking, and maintaining contact are therefore frequently found in dialogues over the telephone or via radio transmission. Especially when communicating over a distance, using communication channels whose reliability is not obvious and/or not instantaneous, a participant's effective presence and readiness are not always evident and are therefore checked (*Hello? Are you there?*).

In face-to-face communication contact is typically not the subject of verbalized dialogue acts, but is indicated and checked nonverbally by means of eye contact and facial expressions.

d. Taking Turns

For understanding and generating turn management acts, a context model needs to represent who currently occupies the speaker role; whether the participant whose context model is considered would like to have it; and if he already has the speaker role whether he wants to keep it, or wants someone else to take over.

e. Time

In order to be in the position to generate a *Stalling* or a *Pausing* act, a dialogue participant must have some information about his own processes involved in the production of a dialogue act. Common causes of *Stalling* are:

- the participant experiences a difficulty in finding or choosing a particular lexical item;
- the participant cannot decide immediately what form to give to the dialogue act that he intends to perform;
- the participant does not immediately have all the information available which is needed for the semantic content of the dialogue act under construction.

A participant's context model should therefore include information about lexical, syntactic, and semantic processes in utterance generation.

f. Discourse Plans

Especially in the case of dialogues with a well-structured underlying task, a participant may have a plan for how to organize the interaction. Professional agents in an information service, such as operators providing information about telephone numbers and services, or workers in call centers and help desks, often structure the interaction with customers in a fixed manner.

The observation that dialogue participants do not always say explicitly what they mean, using for example indirect questions and conversational implicatures as in example (18), has spawned a certain amount of work which considers as a crucially important task of a dialogue participant to recognize his partner's discourse plan, the so-called *plan-based approach* to dialogue (e.g. [21], [36]).

(18) A: Can you help me, I'm out of petrol.
B: There's a garage just around the corner.

Elaborate discourse plans include dialogue acts with a fixed form for opening and closing the dialogue, an order in which to address relevant topics, and strategies for providing feedback. A less elaborate discourse plan may consist of only the intention to answer a question, or the plan to interrupt the current speaker and return to a previous topic.

g. Own and Partner Speech Editing

A dialogue participant who edits his own speech, correcting himself or improving his formulation, makes use of information about what he is saying and about what he wants to say.

The performance and interpretation of dialogue acts where a participant edits the speech of the current speaker requires context models to contain essentially the same information relating to the partner's production processes as is needed for editing one's own speech.

h. Social 'Obligations' or 'Pressures'

Social obligations management (SOM) acts reflect the fact that participation in a dialogue is a form of interaction between people (at least originally), and therefore has to follow certain general conventions of human interaction, such as greeting, thanking, and apologizing in certain circumstances. In such circumstances, dialogue participants have to deal with ‘interactive pressures’ [7] to perform certain actions. For example, before starting a dialogue, in many situations there is a social convention to greet each other - which therefore can occur only at the beginning of a dialogue, and if the dialogue is with an unknown partner, it is often customary to introduce oneself. Such observations can be captured by means of '*interactive pressures*' on dialogue participants. For example, when one walks in the street and sees a familiar person approaching, there is a mounting pressure to perform a greeting as one gets closer, questions arising such as: *Does she notice me? Does she recognize me? Can she hear me?*, to which the answers become more certain as you get nearer, until a point is reached where the pressure has become `unbearable', and one performs that greeting – thereby resolving the pressure. The generation of a greeting thus requires context models to contain the representation of such interactive pressures. See also the next section, on the representation of ‘Social Context’.

4.2 Representation Structures

In designing an implementation of a context model, two main issues arise: (1) which representation formalisms could be appropriate, and (2) what overall structure should a context model have.

The various kinds of information listed in the previous subsection can be represented in an effective context model in many different ways. here we consider an implementation in terms of typed feature structures, as a computationally attractive an sufficiently expressive representation formalism[6]. Other representation formalisms that have been used for context modeling include discourse representation structures (DRSs, see [44]); the representation structures of Constructive Type Theory (so-called 'contexts', see [1],[9]); and 'modular partial models' [8].

For the overall structure of a context model we can take advantage of the orthogonality of the dimensions listed in (12). The choice of such a set of dimensions has the computationally attractive feature that a multifunctional dialogue segment, when interpreted as a set of dialogue acts in different dimensions, is semantically 'decomposed' into components in orthogonal dimensions, corresponding to independent context update operations. Moreover, in those dimensions where a given dialogue segment does not have a communicative function, we know that its interpretation will not affect the corresponding information in the context model. It therefore seems attractive to structure a context model in the same way as the set of dimensions, in order to maximally 'modularize' the context update processes. However, we will see that there are reasons for partitioning a context model in fewer than 10 compartments.

First, auto- and allo-feedback information are not only very similar in nature, but are also closely intertwined, since an allo-feedback act performed by participant A, and directed at participant B, provides or elicits information about B's processing of something said earlier in the dialogue, and when B responds to that he provides information about *his own processing* of that same something, hence B performs an auto-feedback act. Similarly, a response to an auto-feedback act is often an allo-feedback act. Examples (19) and (20) illustrate this.

(19) 1. A: You see what I mean? [allo-feedback]
2. B: I see what you mean. [auto-feedback]

(20) 1. A: This Saturday? [auto-feedback]
2. B: That's right. [allo-feedback]

Second, the information related to time management concerns the progress and time estimate of the speaker's processes involved in utterance interpretation or production, and as such forms an aspect of the information needed for the interpretation or generation of own communication management acts. Hence it seems best to consider time-related information not as a separate compartment in a context model, but rather as part of the information concerning understanding and production processes. This also provides an argument for not separating information about understanding

[6] Typed feature structures can also be used for the semantic representation of natural language, hence for the representation of the semantic content of a dialogue act – see [11].

and production processes into separate compartments, but instead to have a single compartment concerning a participant's information processing in general. We call this compartment the **Cognitive Context**; it contains the information needed for the interpretation and generation of auto- and allo-feedback acts, time management acts, and own- and partner communication management acts.

Third, auto- and allo-feedback acts often refer to something that was said before, or to the interpretation of that, as in the following examples:

(21) a. S: Did you say “Thursday"?
b. S: Could you please repeat that?
c. S: I meant Saturday this week.

In order to deal with such cases, a dialogue participant needs to have a representation of what was said before, and of how that was perceived and interpreted. This kind of information is commonly called a *Dialogue History*, and next to task-related information, it is the kind of information that is most commonly represented in context models.

When a new contribution to an ongoing dialogue is processed, it is segmented into functional segments, corresponding to dialogue acts (according to definition (6)), which leads to an addition to the context model of the understanding partner that may look as in Figure 3, represented in terms of feature structures.

The segment is defined by a start- and end point, its observable verbal and/or nonverbal form, a sender who produced the segment, one or more addressees, and a set of dialogue acts, each characterized by a dimension, a communicative function, and a semantic content. In addition, if the segment under consideration is nonverbal or a backchannel, like *uh-huh*, it may be produced while another participant is occupying the speaker role, hence this needs to be represented separately.

Since the functional segment is the unit of dialogue act analysis, the representation of a record of what has happened in a dialogue is best structured as a chronologically ordered sequence of functional segments with their interpretation as shown in Fig. 3[7].

$$\left[\begin{array}{ll} \mathit{current\_segment}: & \left[\begin{array}{l} \mathit{start}: \langle \mathit{timepoint}\rangle \\ \mathit{end}: \langle \mathit{timepoint}\rangle \\ \mathit{expression}: \langle \mathit{verbatim/nonverbalform}\rangle \\ \mathit{dial\_acts}: \{\langle \left[\begin{array}{l} \mathit{dimension}: \langle \mathit{dim}\rangle \\ \mathit{comm\_function}: \langle \mathit{cf}\rangle \\ \mathit{sem\_content}: \langle \mathit{content}\rangle \end{array}\right] \rangle\} \\ \mathit{sender}: \langle \mathit{participant}\rangle \\ \mathit{addressees}: \{\langle \mathit{participant}\rangle\} \\ \mathit{speaker}: \langle \mathit{participant}\rangle \end{array}\right] \end{array}\right]$$

Fig. 3. Feature structure representation of Dialogue History element

[7] The exact chronological organization of the Dialogue History is a nontrivial matter, in view of the occurrence of overlapping and discontinuous functional segments contributed by the same speaker, as well as overlapping segments of (linguistic and/or nonverbal) communicative behaviour by other participants.

Both auto- and allo-feedback need a record of the dialogue history; for own- and partner communication management the current functional segment is needed; and for the generation and interpretation of discourse structuring acts a representation of the planned 'dialogue future' is needed. Moreover, for representing that a participant wants or plans to obtain the speaker role, this should also be represented in the dialogue future component of his context model. The information who currently occupies the speaker role is included in the representation of the current functional segment - so together, *current segment* and *dialogue future* contain the information relevant for generating and interpreting turn management acts; no separate compartment in the context model is needed for this purpose.

All in all, the linguistic context of a stretch of communicative behaviour, i.e. the surrounding dialogue, can be represented in a participant's context model as consisting of three components:

1. the *dialogue history*, i.e. a record of past communicative events;
2. the *current segment,* recording what is currently being contributed to the dialogue;
3. the *dialogue future*, i.e. the discourse plan of the dialogue participant.

This representation of the traditional notion of context in linguistics is one of the compartments in a context model, and we will refer to this compartment as the **Linguistic Context**.

We thus see that the Linguistic Context compartment together with the Cognitive Context contains all the information needed for the generation and interpretation of dialogue acts in the following dimensions: (1) Turn Management; (2) Discourse Structuring; (3) Time Management; (4) Own Communication Management; (5) Partner Communication Management; (6) Auto-Feedback; and (7) Allo-Feedback, i.e., in 7 of the 10 dimensions.

In order to keep the representation structures as simple as possible, we will from now on consider only dialogues with two participants; the generalization to multiple participants is straightforward.

For the task- or activity-related information that needs to be represented in a context model we have seen in (16) that we have to represent a participant's own information about the task, his assumptions about the information that the dialogue partner possesses, and his assumptions concerning the sharing of information. Since task-related information is most often reflected in the semantic content of dialogue acts with general-purpose functions, we will refer to the context model compartment that contains this kind of information as the **Semantic Context**; in terms of feature structures, it can be organized as follows:

$$\begin{bmatrix} SemContext : \begin{bmatrix} own\_task\_model : \langle beliefs \rangle \\ partner\_task\_model : \langle beliefs \rangle \\ common\_ground : \langle mutual\_beliefs \rangle \end{bmatrix} \end{bmatrix}$$

Fig. 4. Feature structure representation of `Semantic Context'

The representation in a context model of the processing state of the participant who 'owns' the context model, and of his assumptions concerning the partner's processing state, can be represented in a similar way, as consisting of (a) a list of parameters describing the status of each of the processes that are distinguished in utterance understanding and production; (b) a similar list describing assumptions about the status of these processes on the part of the partner; and (c) a description of assumed shared information of this kind. Such shared information plays a crucial role in processes of grounding; in fact, grounding applies not only to information about the underlying task, but also to what was said and how it was understood, as well as to other types of information in the context model. We will therefore see similar three-part structures in all the context model compartments.

The information needed for generating and interpreting dialogue acts in the Contact Management dimension is quite simple and can be characterized with two features, one to indicate whether a participant is present, in the sense of being in a position to use the communicative channels available in the dialogue setting under consideration (such as the telephone, or the computer in an on-line chat); and another to indicate whether a participant is ready to communicate. The compartment containing this information is called the **Perceptual/Physical Context**.[8]

Finally, the context information needed for generating and understanding Social Obligations Management (SOM) acts is somewhat different from that for other dimensions, due to the highly conventional nature of these acts. SOM acts, moreover, come in adjacency pairs. A greeting puts a pressure on the addressee to perform a return greeting. Same for saying goodbye. Introducing oneself puts a pressure on the addressee to also introduce herself. Apologizing puts pressure on the addressee to accept the apology; thanking puts pressure on the addressee to mitigate the cause of the thanking, as in *de rien; pas de quoi* (French); *de nada* (Spanish, Portuguese); *niente* (Italian); *ingen ting at takke for* (Danish). The pressures created by an initial SOM act are called *reactive pressures*. They are similar in nature to interactive pressures (see above, Section 4.1.h); the difference is that interactive pressures are created by properties of the interactive situation, while reactive pressures are created directly by SOM acts.

An interactive pressure can be represented as specification of the dialogue act that one is pressured to perform. This is usually an incomplete specification, for instance only specifying a communicative function. A reactive pressure is typically more detailed, since it is conventional to 'align' with the previous speaker and respond to an initial SOM act with a responsive act of similar form.

All in all, this leads to a context model with five compartments: the Linguistic, Semantic, Cognitive, Perceptual/Physical, and Social Context, which can be implemented in terms of feature structures as shown in Figure 5 (from [43]).

[8] This component may also contain information about perceptual context aspects such as the availability of visual information about certain situations in the task domain.

$$\left[\begin{array}{ll}
LingContext: & \left[\begin{array}{ll}
system: & \left[\begin{array}{ll}
\multicolumn{2}{l}{dialogue\_history: \langle functional\_segments\rangle}\\
current\_segment: & \left[\begin{array}{l}
start: \langle timepoint\rangle\\
end: \langle timepoint\rangle\\
expression: \langle verbatim/nonverbalform\rangle\\
dial\_acts: \{\langle \left[\begin{array}{l} dimension: \langle dim\rangle\\ comm\_function: \langle cf\rangle\\ sem\_content: \langle content\rangle \end{array}\right] \rangle\}\\
sender: system \mid user
\end{array}\right]\\
\multicolumn{2}{l}{dialogue\_future: \langle discourse\_plan\rangle}
\end{array}\right]\\
\multicolumn{2}{l}{user: \langle user's\ linguistic\ context\ (according\ to\ system)\rangle}\\
\multicolumn{2}{l}{shared: \langle shared\ linguistic\ context\rangle}
\end{array}\right]\\
\\
SemContext: & \left[\begin{array}{l}
system: \langle task\_model: beliefs,\ goals\rangle\\
user: \langle user's\ semantic\ context\ (according\ to\ system)\rangle\\
shared: \langle task\ related\ mutual\ beliefs\ (assumed\ common\ ground)\rangle
\end{array}\right]\\
\\
CogContext: & \left[\begin{array}{ll}
own\_proc\_state: & \left[\begin{array}{l} proc\_level: att \mid perc \mid int \mid eval \mid exec\\ proc\_problem: yes \mid no\\ problem\_input: x\\ time\_need: negligible \mid small \mid substantial \end{array}\right]\\
user\_proc\_state: & \left[\begin{array}{l} proc\_level: att \mid perc \mid int \mid eval \mid exec\\ proc\_problem: yes \mid no\\ problem\_input: y\\ time\_need: small \end{array}\right]\\
\multicolumn{2}{l}{shared\_CogContext: [\,.....\,]}
\end{array}\right]\\
\\
PercPhysContext: & \left[\begin{array}{ll}
system: & \left[\begin{array}{l} presence: positive \mid negative\\ readiness: positive \mid negative \end{array}\right]\\
\multicolumn{2}{l}{user: \langle user's\ perceptual/physical\ context\ (according\ to\ system)\rangle}\\
\multicolumn{2}{l}{shared: \langle assumed\ mutual\ beliefs\ about\ perceptual/physical\ context\rangle}
\end{array}\right]\\
\\
SocContext: & \left[\begin{array}{ll}
system: & \left[\begin{array}{l} reactive\_pressure: \langle dial\_act\rangle\\ interactive\_pressure: \langle dial\_act\rangle \end{array}\right]\\
\multicolumn{2}{l}{user: \langle user'ssocial\ context\ (according\ to\ system\rangle}\\
shared: & \left[\begin{array}{l} reactive\_pressure: \langle dial\_act\rangle\\ interactive\_pressure: \langle dial\_act\rangle \end{array}\right]
\end{array}\right]
\end{array}\right]$$

Fig. 5. Feature structure representation of context model in dialogue system

5 Applications

The DIT framework and dialogue act taxonomy have been used successfully in a variety of applications:

- for empirical, theoretical and computational modelling of semantic and pragmatic phenomena in spoken and multimodal dialogue;
- for dialogue annotation, and as the starting point for an ISO effort to define a standard for dialogue act annotation;
- for the design of dialogue system components, in particular for multimodal input interpretation, dialogue management, and the generation of multifunctional utterances in spoken dialogue systems;

In this section we briefly consider each of these applications.

5.1 Semantic and Pragmatic Dialogue Analysis

a. Information flow and grounding. Every communicative function in the DIT^{++} taxonomy is formally defined as a particular type of update operation on an addressee's context model. Depending on its dimension, a dialogue act updates a particular context component; a multifunctional utterance leads to the update of several components. This approach provides good instruments for studying and modeling the flow of information between the participants in a dialogue. Fine-grained models of information flow through the understanding of dialogue behaviour in terms of dialogue acts have been developed and analysed in [37], and have resulted in a empirically-based computational model of grounding in dialogue [18].

b. Semantics of discourse markers. Another semantic study based on the multidimensional approach of DIT is that of the semantics of discourse markers; words or phrases that connect the pieces in a dialogue (or in a monologue), like *but, and, so,* and *well.* It was shown that such expressions often perform multiple semantic functions, which are well explained in terms of the dimensions in the DIT^{++} taxonomy (see [39]).

c. Multifunctionality and co-occurrence patterns. To generate multifunctional dialogue behaviour in a sensible way, it is important to have qualitative and quantitative knowledge of this phenomenon, and to know which kinds of multifunctional utterances occur in natural dialogue. It has been shown (see [14], [15]) that, when a fine-grained segmentation is applied to dialogue, with possibly overlapping and interleaved functional segments, the average multifunctionality of a segment in spoken dialogue without visual contact amounts to 1.6 when only explicitly expressed and conversationally implicated functions are taken into account, and 3.6 when entailed functions are also counted.

d. The interpretation of nonverbal dialogue behaviour. An investigation into the applicability of the DIT^{++} taxonomy to nonverbal behaviour in dialogues in the AMI corpus showed that the DIT^{++} functions provided fulll coverage for interpreting the nonverbal activity [38]. The nonverbal behaviour may serve five purposes: (1) emphasizing or articulating the semantic content of dialogue acts; (2) emphasizing or supporting the communicative functions of synchronous verbal behaviour; (3) performing separate dialogue acts in parallel to what is contributed by the current speaker (without turn shifting); (4) expressing an emotion or attitude; or (5) expressing a separate communicative function in parallel to what the same speaker is expressing verbally. The latter occurs relatively rarely, as witnessed by the fact that the multifunctionality of dialogue segments shows only a small increase when synchronous nonverbal behaviour is taken into account.

It may be noted, on the other hand, that with visual contact there is an increase of more than 25% of the number of functional segments, mostly due to participants not in the speaker role providing nonverbal feedback.

5.2 Annotation

a. DIT^{++} annotation. The DIT^{++} taxonomy has been applied in manual annotation of dialogues from various corpora: the DIAMOND corpus of two-party instructional human-human Dutch dialogues (1,408 utterances)[9]; the AMI corpus of task-oriented human-human multi-party English dialogues (3,897 utterances)[10]; the OVIS corpus of task-oriented human-computer Dutch dialogues (3,942 utterances); TRAINS dialogues[11] (in English); and Map Task dialogues[12] both in English and in Dutch. Geertzen et al. [28] report on the consistency with which naive annotators as well as expert annotators were able to annotate, and compares the results. Expert annotators achieve agreement scores of more than 90%; naive annotators achieve scores in the order of 60%.

b. The LIRICS project. In the EU project LIRICS[13] a taxonomy of communicative functions was defined which is a slightly simplified version of the DIT^{++} taxonomy, retaining its dimensions but eliminating the distinction of levels of feedback as well as uncertain variants of information-providing functions, informs with rhetorical functions, and some of the low-frequency functions [34]. The resulting taxonomy has 23 general-purpose functions (where DIT^{++} has 31) and 30 dimension-specific functions (where DIT^{++} has 57, of which 20 fine-grained feedback functions).

The usability of this taxonomy was tested in manual annotation of the LIRICS test suites in Dutch, English, and Italian by three expert annotators. Remarkably high, agreement was found between the annotators; for the general-purpose functions an average κ score was found of 0.98; for the other categories scores ranged from 0.94 for SOM acts to 0.99 for auto-feedback acts (see [35]).

c. Towards an ISO standard for functional dialogue markup

In 2008 the International Organization for Standards started up the project Semantic Annotation Framework, Part 2: Dialogue acts, which aims at developing an international standard for the markup of communicative functions in dialogue. This project builds on the results of an ISO study group on interoperability in linguistic annotation, of which the European project LIRICS was a spin-off.

The ISO project takes the DIT^{++} and LIRICS taxonomies as point of departure for defining a comprehensive open standard for functional dialogue markup. The latest version of the ISO proposal has the status of Draft International Standard (ISO 24617-2:2010). It includes the definition of 26 general-purpose functions and 30 dimension-specific functions, plus three binary-valued qualification attributes (*certainty*, *conditionality*, and *partiality*) and one attribute with an open class of values *sentiment*). These concepts are all defined according to ISO standard 12620 for data category definitions, and will eventually be included in the on-line ISOcat data category registry (http://www.isocat.org)[14].

[9] For more information see Geertzen, J., Y. Girard, and R. Morante (2004) The DIAMOND project. Poster at the 8th Workshop on the Semantics and Pragmatics of Dialogue (CATALOG), Barcelona.

[10] `http://www.amiproject.org`

[11] See [2].

[12] See [22].

[13] Linguistic InfRastructure for Interoperable Resources and Systems; for more information see `http://lirics.loria.fr`

[14] The data categories are temporarily available at `http://semantic-annotation.uvt.nl`

In addition to these data category definitions, the standard provides the XML-based annotation language DiAML (Dialogue Act Markup Language), with annotation guidelines and examples of annotated dialogues.[15] See [29] and the summary description in [20].

The DIT^{++} definitions of communicative functions have been adapted in release 5 (April 2010) so as to be identical to those in the ISO standard for functions which are shared by the two schemes; as a result, the ISO set of concepts is a proper subset of the DIT^{++} concepts, and DIT^{++} annotations can make use of the DiAML representation format.

5.3 Dialogue System Design

a. Dialogue management. The DIT^{++} taxonomy has been used in the design and implementation of the PARADIME dialogue manager, which forms part of the IMIX system for multimodal information extraction (see [30], [31]). The multidimensional dialogue manager generates sets of dialogue acts (in formal representation) that are appropriate in the current dialogue context, and delivers these to a module for expressing a set of dialogue acts in a multifunctional utterance. It makes use of autonomous software agents for each dimension, that generate candidate dialogue acts in that dimension. An evaluation agent examines the set of candidate dialogue acts that is generated in this fashion and decides on sets of candidate acts that can be turned into multifunctional utterances. Figure 6 shows this architecture.

This approach to dialogue utterance generation opens the ppossibiliity to generate multifunctional utterances in a deliberate and controlled fashion.

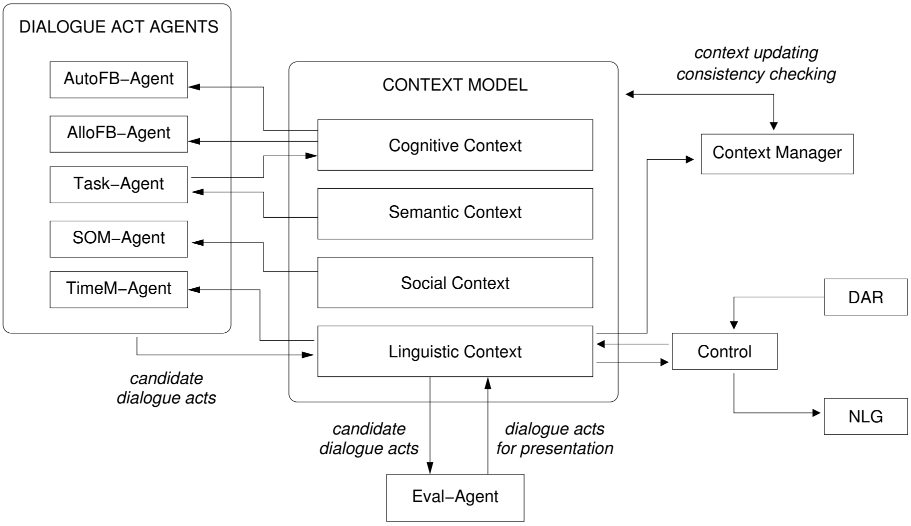

Fig. 6. Multidimensional dialogue manager architecture

[15] See also example (14) above.

A more complete implementation of the context model described in Figure 5 has been developed by Petukhova et al. [43] for experimentation with constraints on the generation of multifunctional dialogue utterances.

b. Machine recognition of DIT^{++} functions. A prerequisite for using dialogue acts in a dialogue manager is that the dialogue system is able to recognize dialogue acts with acceptable precision. The automatic recognition of the DIT^{++} dialogue acts (as well as in other taxonomies, such as DAMSL) was investigated for the corpora mentioned above, as well as for dialogues from the Monroe and MRDA corpora. For the various DIT dimensions, F_1 scores were found ranging from 62.6% to 96.6%, without any tweaking of the features used in the machine learning algorithms. This suggests that the recognition of (multiple) functions in the taxonomy is a realistic enterprise. For more information see [27].

5.4 Functional Markup of Embodied Conversational Agents

The DIT^{++} taxonomy has been applied in the analysis of dialogues recorded in a range of different settings involving various combinations of modalities (face-to-face two-person and multiparty conversation, human-computer over the telephone, helpdesk dialogues with visual information on a display,...), and has shown that nonverbal communicative behaviour does not express other types of dialogue acts than those that have been identified in spoken language. The 88 communicative functions in the DIT^{++} taxonomy[16] are sufficient to capture the types of dialogue acts performed not only by verbal communicative behaviour but equally by nonverbal and multimodal behaviour.

This is not to say that nonverbal behaviour does not add anything to the verbal behaviour in multimodal dialogue. Facial expressions, head movements, and hand and shoulder gestures are known to often express certain emotions and attitudes which are added to the speaker's words without changing the communicative function of the verbally expressed dialogue act. Gaze direction, head movements, facial expressions, hand gestures and body posture (in addition to prosody) has also been shown (see [42]) to be used by speakers for expressing their certainty of the correctness of the information that they offer in information-providing acts, or of their commitment in a commissive act; see Table 1 for the main features of nonverbal behaviour that are used to signal certainty.

A variety of taxonomies has been proposed for the classification of emotions and attitudes that dialogue participants may exhibit, from the 6 basic emotions in Ekman's pioneering work [25] to the corpus-based 14 sentiments distinguished by Reidsma et al. [46]; other proposals include those of Craggs and McGee-Wood [24]; Laskowski and Burger [33]; and Ekman's extended taxonomy [26]. For the nonverbal expression of Ekman's basic 6 emotions Petukhova and Bunt [42] found in the AMI multiparty dialogue corpus the features of facial expressions shown in Table 2.

[16] DIT^{++} release 5; see `http://dit.uvt.nl`

Table 1. Nonverbal expressions of certainty

Certainty	Gaze direction	Head movement	Facial expression	Gesture	Posture orientation
Uncertain	aversion redirection involuntary eye movements	waggles	lip-compression; lip-pout; biting/liking; lowering eyebrows; constricting forehead muscles	adaptors, e.g. self-touching; shoulder shrug	posture shift
Certain	direct eye contact;	head nod (for emphasis)	thin lips; pushing up the chin boss; widely open eyes;	beat gestures	leaning forward /to addressee

Phenomena such as the certainty and emotions of a speaker associated with a dialogue act can be modelled with the help of the concept of a *function qualifier* applied to a communicative functions, as mentioned above (see (13) above), and the use of attributes like the sentiment qualifier attribute, used in the corresponding DiAML representation (14)). The findings of Petukhova and Bunt, reported in [42], have been incorporated both in the latest version of the DIT^{++} annotation schema and in the ISO Draft International Standard ISO 24617-2[17].

Table 2. Facial expressions corresponding with Ekman's six basic emotions

Emotion	Facial expression					
	Forehead	Eyebrows	Eyes	Cheeks	Lips	Chin
Anger	wrinkled	lower lids; pulled together	lower eyelids tensed and straightened		tensed; pressed together	chin bossf pushed up
Disgust		pulled down down	lower lids tensed upper lids raised; narrowed		upper lip; drawn up; pressed together; mouth open	
Fear		raised straight up	lids raised up		corners pulled; lips stretched; mouth open	jaw dropped
Happy			lids narrowed; eye corners wrinkled	outer, upper area raised	corners raised	
Sad	wrinkled	pulled together raised in center of forehead	narrowed	raised	stretched; corners; turned downt	chin boss pushed up
Surprise	wrinkled	raised straight up	upper lids raised		mouth open; tensed or relaxed	jaw drop

[17] For the most recent documentation of this draft ISO standard see `http://semantic-annotation.uvt.nl`

These observations have given rise to the idea that the DIT^{++} taxonomy and the DiAML language may be useful for the functional markup of the behaviour of embodied conversational agents (ECAs). The use of ECAs in user interface is motivated by the idea that the addition of a face, and possibly more of a body, can help to make the dialogue between a computer and a user more natural, in particular because it may bring emotions and attitudes to the interaction by smiling, bowing, looking happy, sorry, embarrassed, surprised, and so on. But this can only work well if the nonverbal behaviour of the ECA is indeed ‘natural' in the sense of expressing the same functions and qualities as the human behaviour that it resembles, and if the behaviour is displayed in the same contexts where a human would display it.

In the ECA community an effort has started to design a standard for the functional markup of ECA behaviour, including a functional markup language. The DiAML language can be the basis of such a markup language, with the elements of the DIT^{++} taxonomy as values of DiAML attributes, and the qualifier attributes and values defined in ISO DIS 24617-2. The latter can be replaced by or supplemented with other or more fine-grained sets of qualifier attributes and values, as appropriate for specific ECA applications.

6 Perspectives and Future Work

In this paper we have presented the DIT framework for context-based dialogue analysis, annotation, interpretation, and generation. We have used the 10 dimensions of the DIT^{++} taxonomy of communicative functions for identifying the types of information that a context model should contain. An analysis of these information types has lead us to design a dialogue participant's context model as consisting of five compartments, called Linguistic, Semantic, Cognitive, Perceptual/Physical, and Social Context, and we have made this design concrete by showing an implementation based on typed feature structures.

We have illustrated the advantages of an articulate context-based approach by its use in a variety of applications, including the detailed modeling of information flow and grounding in dialogue, the semantics of discourse markers, the annotation of spoken and multimodal dialogue (two- and multiparty; human-human and human-computer); the definition of an ISO standard for dialogue annotation; the design of components of a computer dialogue system; and the functional markup of the behaviour of embodied conversational agents.

The DIT^{++} taxonomy of communicative functions has been made fully ISO-compatible, in the sense that:

- the dimensions and the set of communicative functions defined in the ISO standard are proper subsets of those defined in DIT^{++};
- the set of qualifiers defined in the ISO (draft) standard has been adopted;
- the categories of entities and relations underlying the standard is shared with the DIT model, hence the DiAML markup language defined as part of the ISO standard can equally well use the more extended set of concepts defined in DIT^{++}. In particular, for the 9 dimensions shared by the two annotation schemes (Contact Management being the only one not shared), every communicative function

defined in DIT^{++} either coincides with a function defined in the ISO standard, or is a specialization of such a function.

Any ISO 24617-2 annotation is therefore also a DIT^{++} annotation, and any DIT^{++} annotation can be converted into a compatible ISO 24617-2 annotation (except for annotations in the Contact Management dimension) by replacing fine-grained DIT^{++}-functions by less fine-grained functions where necessary.

The context model described in this paper has been fully implemented for experimentation with dialogue management strategies, but has not yet been incorporated in a dialogue system, where it would offer the possibility to deliberately generate multifunctional utterances; this is an item on the agenda for future work.

Acknowledgements

I thank the members of the Tilburg Dialogue Club for discussions and experiments during which the DIT framework was shaped over the years. This includes Ielka van der Sluis, Simon Keizer, Hans van Dam, Jeroen Geertzen, Yann Girard, Roser Morante, Rintse van der Werf, and especially Volha Petukhova. I thank the members of the ISO 24617-2 project team: Jan Alexandersson, Jean Carletta, Jae-Woong Choe, Alex Fang, Koiti Hasida, Volha Petukhova, Andrei Popescu-Belis, Claudia Soria, and David Traum for discussions on dialogue analysis and dialogue act annotation. The same goes for the members of the Expert Consultancy Group of the ISO project, in particular Jens Allwood, Thierry Declerck, Dirk Heylen, Kristiina Jokinen, Maciej Karpinski, Kiyong Lee, David Novick, Laurent Romary, and Nicla Rossini.

References

1. Ahn, R.: Agents, Objects and Events. A computational approach to knowledge, observation and communication. PhD. Thesis, Eindhoven University of Technology (2001)
2. Allen, J., Schubert, L., Ferguson, G., Heeman, P., Hwang, C.J., Kato, T., Light, M., Martin, N., Miller, B., Poesio, M., Traum, D.: The TRAINS Project: A case study in building a conversational planning agent. Technical Note 94-3, University of Rochester (1994)
3. Allen, J., Core, M.: DAMSL: Dialogue Act Markup in Several Layers (Draft 2.1). Technical Report, Discourse Resource Initiative (1997)
4. Allwood, J.: An activity-based approach to pragmatics. In: Bunt, H., Black, W. (eds.) Abduction, Belief and Context in Dialogue. Studies in Computational Pragmatics, pp. 47–80. Benjamins, Amsterdam (2000)
5. Allwood, J., Nivre, J., Ahlsén, E.: Speech Management or the Non-written Life of Speech. Nordic Journal of Linguistics 13, 1–48 (1990)
6. Bunt, H.: Context and Dialogue Control. Think Quarterly 3(1), 19–31 (1994)
7. Bunt, H.: Dynamic Interpretation and Dialogue Theory. In: Taylor, M., Bouwhuis, D., Néel, F. (eds.) The Structure of Multimodal Dialogue, vol. 2, pp. 139–166. Benjamins, Amsterdam (1995)
8. Bunt, H.: Context Representation for Dialogue Management. In: Bouquet, P., Serafini, L., Brézillon, P., Benerecetti, M., Castellani, F. (eds.) CONTEXT 1999. LNCS (LNAI), vol. 1688, pp. 77–90. Springer, Heidelberg (1999)
9. Bunt, H.: Dialogue pragmatics and context specification. In: Bunt, H., Black, W. (eds.) Abduction, Belief and Context in Dialogue. Studies in Computational Pragmatics, pp. 81–150. Benjamins, Amsterdam (2000)

10. Bunt, H.: A Framework for Dialogue Act Specification. In: 4th Joint ISO-SIGSEM Workshop on the Representation of Multimodal Semantic Information, Tilburg (January 2005), `http://let.uvt.nl/research/ti/sigsem/wg`
11. Bunt, H.: Quantification and modification represented as Feature Structures. In: 6th International Workshop on Computational Semantics (IWCS-6), pp. 54–65 (2005)
12. Bunt, H.: Dimensions in dialogue annotation. In: LREC 2006, 5th International Conference on Language Resources and Evaluation, pp. 919–924 (2006)
13. Bunt, H.: The DIT $^{++}$ taxonomy for functional dialogue markup. In: AAMAS 2009 Workshop, Towards a Standard Markup Language for Embodied Dialogue Acts, pp. 13–24 (2009)
14. Bunt, H.: Multifunctionality and multidimensional dialogue semantics. In: DiaHolmia, 13th Workshop on the Semantics and Pragmatics of Dialogue, pp. 3–14 (2009)
15. Bunt, H.: Multifunctionality in Dialogue. Computer Speech and Language 25(2011), 222–245 (2010), `http://dx.doi.org/10.1016/j.csl.2010.04.006`
16. Bunt, H., Girard, Y.: Designing an open, multidimensional dialogue act taxonomy. In: Gardent, C., Gaiffe, B. (eds.) DIALOR 2005, Proceedings of the Ninth Workshop on the Semantics and Pragmatics of Dialogue, pp. 37–44 (2005)
17. Bunt, H., Keizer, S.: Dialogue semantics links annotation for context representation. In: Joint TALK/AMI Workshop on Standards for Multimodal Dialogue Context (2005), `http://homepages.inf.ed.ac.uk/olemon/standcon-SO1.html`
18. Bunt, H., Keizer, S., Morante, R.: An empirically-based computational model of grounding in dialogue. In: 8th Workshop on Discourse and Dialogue (SIGDIAL 2007), pp. 283–290 (2007)
19. Bunt, H., Romary, L.: Standardization in Multimodal Content Representation: Some Methodological Issues. In: 3rd International Conference on Language resources and Evaluation (LREC 2004), pp. 2219–2222 (2004)
20. Bunt, H., Alexandersson, J., Carletta, J., Choe, J.-W., Fang, A., Hasida, K., Lee, K., Petukhova, V., Popescu-Belis, A., Romary, L., Soria, C., Traum, D.: Towards an ISO standard for dialogue act annotation. In: LREC 2010, 8th International Conference on Language Resources and evaluation (2010)
21. Carberry, A.: Plan Recognition in Natural Language Dialogue. MIT Press, Cambridge (1990)
22. Carletta, J., Isard, A., Isard, S., Kowtko, J., Doherty-Sneddon, G.: HCRC dialogue structure coding manual. Technical Report HCRC/TR-82 (1996)
23. Clark, H.: Using Language. Cambridge University Press, Cambridge (1996)
24. Craggs, R., McGee Wood, M.: A categorical annotation scheme for emotion in the linguistic content of dialogue. In: André, E., Dybkjær, L., Minker, W., Heisterkamp, P. (eds.) ADS 2004. LNCS (LNAI), vol. 3068, pp. 89–100. Springer, Heidelberg (2004)
25. Ekman, P.: Universals and cultural differences in facial expressions of emotion. In: Cole, J. (ed.) Nebraska Symposium on Motivation. University of Nebraska Press, Lincoln (1972)
26. Ekman, P.: Basic Emotions. In: Dalgliesh, T., Power, M. (eds.) Handbook of Cognition and Emotion, pp. 207–283. John Wiley, Sussex (1999)
27. Geertzen, J.: The automatic recognition and prediction of dialogue acts. PhD Thesis, Tilburg University (2009)
28. Geertzen, J., Petukhova, V., Bunt, H.: A multidimensional approach to dialogue segmentation and dialogue act classification. In: 8th Workshop on Discourse and Dialogue (SIGDIAL 2007), pp. 140–147 (2007)
29. ISO DIS 24617-2: Language resource management - Semantic Annotation Framework - part 2: dialogue acts, ISO, Geneva (August 2010), `http://semantic-annotation.uvt.nl`

30. Keizer, S., Bunt, H.: Multidimensional dialogue management. In: 7th Workshop on Discourse and Dialogue (SIGDIAL 2006), pp. 37–45 (2006)
31. Keizer, S., Bunt, H.: Evaluating combinations of dialogue acts for generation. In: 8th Workshop on Discourse and Dialogue (SIGDIAL 2007), pp. 158–165 (2007)
32. Larsson, S.: Coding Schemas for Dialog Moves. Technical report from the S-DIME project (1998), http://www.ling.gu.se/sl
33. Laskowski, K., Burger, S.: Annotation and analysis of emotionally relevant behaviour in the ISL meeting corpus. In: LREC 2006, 5th International Conference on Language Resources and Evaluation (2006)
34. LIRICS D4.3: Documented Set of Semantic Data Categories. EU eContent Project LIRICS Deliverable D4.3. 3 (2007a), http://semantic-annotation.uvt.nl
35. LIRICS D4.4: Multilingual Test Suites for Semantically Annotated Data. EU eContent Project LIRICS Deliverable D4.3 (2007b), http://semantic-annotation.uvt.nl
36. Litman, D., Allen, J.: A Plan Recognition Model for Subdialogues in Conversation. Cognitive Science 11(2), 163–200 (1987)
37. Morante, R.: Computing meaning in interaction. PhD Thesis, Tilburg University (2007)
38. Petukhova, V.: Multidimensional interaction of multimodal dialogue acts in meetings. MA thesis, Tilburg University (2005)
39. Petukhova, V., Bunt, H.: Towards a multidimensional semantics of discourse markers in spoken dialogue. In: 8th International Workshop on Computational Semantics (IWCS-8), pp. 157–168 (2009)
40. Petukhova, V., Bunt, H.: Dimensions of Communication: a survey. TiCC Technical Report TR 09-003, Tilburg Center for Cognition and Communication, Tilburg University (2009)
41. Petukhova, V., Bunt, H.: The independence of dimensions in multidimensional dialogue act annotation. In: Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the ACL (NAACL 2009), pp. 197–200 (2009)
42. Petukhova, V., Bunt, H.: Introducing Communicative Function Qualifiers. In: Fang, A., Ide, N., Webster, J. (eds.) Language Resources and Global Interoperability. Proceedings of the Second International Conference on Global Interoperability for Language Resources (ICGL 2010), pp. 123–131. City University of Hong Kong (2010)
43. Petukhova, V., Bunt, H., Malchanau, A.: Empirical and theoretical constraints on dialogue act combinations. In: 14th Workshop on the Semantics and Pragmatics of Dialogue (PozDial), Poznan (2010)
44. Poesio, M., Traum, D.: Towards an Axiomatization of Dialogue Acts. In: Twente Workshop on the Semantics and Pragmatics of Dialogue, pp. 207–222 (1998)
45. Popescu-Belis, A.: Dialogue Acts: One or More Dimensions?. ISSCO Working Paper 62, ISSCO, Geneva (2005)
46. Reidsma, D., Heylen, D., Odelman, R.: Annotating emotion in meetings. In: LREC 2006, 5th International Conference on Language Resources and Evaluation (2006)
47. Traum, D.: A Computational Theory of Grounding in Natural Language Conversation. PhD Thesis, Department of Computer Science, University of Rochester (1994)
48. Traum, D., Larsson, S.: The Information State Approach to Dialogue Management. In: Smith, R., van Kuppevelt, J. (eds.) Current and New Directions in Discourse and Dialogue, pp. 325–353. Kluwer, Dordrecht (2003)