ISO 24617-2: A semantically-based standard for dialogue annotation

Harry Bunt*, Jan Alexandersson, Jae-Woong Choe, Alex Chengyu Fang, Koiti Hasida, Volha Petukhova, Andrei Popescu-Belis and David Traum

*TiCC: Tilburg Center for Cognition and Communication, Tilburg University, The Netherlands
harry.bunt@uvt.nl

Abstract

This paper summarizes the latest, final version of ISO standard 24617-2 "Semantic annotation framework, Part 2: Dialogue acts". Compared to the preliminary version ISO DIS 24617-2:2010, described in Bunt et al. (2010), the final version additionally includes concepts for annotating rhetorical relations between dialogue units, defines a full-blown compositional semantics for the Dialogue Act Markup Language DiAML (resulting, as a side-effect, in a different treatment of functional dependence relations among dialogue acts and feedback dependence relations); and specifies an optimally transparent XML-based reference format for the representation of DiAML annotations, based on the systematic application of the notion of 'ideal concrete syntax'. We describe these differences and briefly discuss the design and implementation of an incremental method for dialogue act recognition, which proves the usability of the ISO standard for automatic dialogue annotation.

Keywords: semantic annotation; international standards; dialogue acts

1. Introduction

Semantic annotation schemes are intended for annotating primary data such as texts, speech transcripts, pictures, or recordings of multimodal or nonverbal communicative behaviour with semantic information. When the primary data originate from interactive use of language and/or other modalities, the semantic characterization of its segments almost certainly involves their interpretation in terms of dialogue acts, i.e. of how the speaker wants to influence the interactive situation.

Dialogue acts can be interpreted formally as update operations, to be applied to the information states of the interacting participants; this view is commonly known as the 'information-state update' or 'context-change' approach to the analysis of dialogue – see e.g. Traum & Larsson (2003); Bunt (2000; 2011). The two main components of a dialogue act are its communicative function and semantic content. The semantic content specifies the objects, propositions, events, etc. that the dialogue act is about; the communicative function specifies of the way an addressee should use the semantic content to update his information state. Dialogue act annotation is the activity of marking up stretches of dialogue with information about the dialogue acts which it contains, and is usually focused on marking up the communicative functions of utterances.

ISO standard 24617-2 has been developed in recent years in view of the need for an application-independent dialogue act annotation scheme that is both empirically and theoretically well founded, that can adequately deal with typed, spoken, and multimodal dialogue, and that can be effectively used both by human annotators and by automatic annotation methods. A preliminary, draft version of this standard ISO DIS 24617-2:2010) was outlined in Bunt et al. (2010); since then the standard has been more fully developed and formally approved by ISO.

This paper describes the changes that have been made to the preliminary version of the standard. The most important changes and developments, discussed in sections 3-5, concern the following points:

- addition of rhetorical relations between dialogue units;
- development of a full-blown compositional semantics for the Dialogue Act Markup Language DiAML (resulting, as a side-effect, in a different treatment of functional dependence relations among dialogue acts and feedback dependence relations);
- systematic application of the notion of 'ideal concrete syntax' to the design of an XML-based representation format for DiAML annotations;
- design and implementation of an incremental interpretation method for dialogue acts, proving its usability in automatic dialogue annotation.

First, Section 2 summarizes the main features of the ISO 24617-2 standard which have remained unchanged.

2. ISO DIS 24617-2:2010

The main features of ISO standard 24617-2 which have remained unchanged compared to the preliminary version DIS 24617-2:2010 are the following:

1. the notion of a semantic dimension in dialogue act analysis is explicitly defined, and nine semantic dimensions are defined, which are distinguished on empirical and theoretical grounds. This serves to optimally support multidimensional dialogue annotation, i.e. the annotation of units in dialogue with more than one functional tag;
2. two classes of communicative functions are distinguished: 'dimension-specific' ones (like Pause, Apology, and Take Turn), which can only be used in one specific dimension, and general-purpose ones, like Question, Answer, Offer, and Instruct, which can be combined with any kind of semantic content and form a dialogue act in the corresponding dimension (e.g. feedback question);

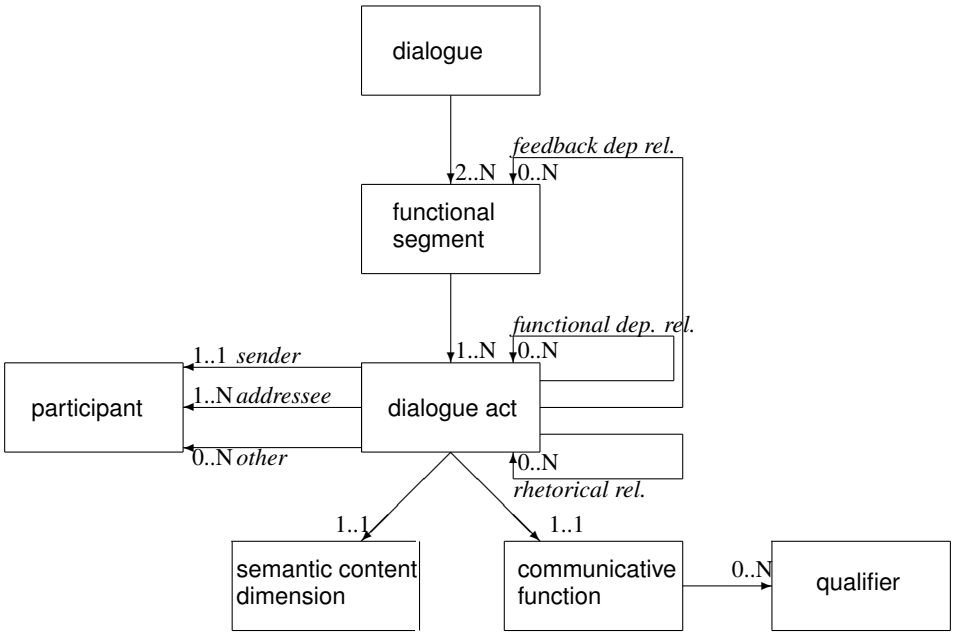

Figure 1: Metamodel for dialogue act annotation.

3. so-called 'function qualifiers' are defined for expressing that a dialogue act is performed (un-)conditionally, with (un-)certainty, or with a particular sentiment;

4. functional and feedback dependence relations are defined which relate a dialogue act to units earlier in a dialogue, e.g. for indicating which question is answered by a given answer, or which utterance the speaker is providing feedback about;

5. the notion of a 'functional segment' is used as the unit of dialogue act annotation; it is defined as a minimal stretch of behaviour which has one or more communicative functions;

6. multidimensional segmentation is applied, i.e. functional segments are distinguished for each dimension, e.g. a segment carrying a feedback function may overlap with a segment that carries a task-related function;

7. the Dialogue Act Markup Language (DiAML) is defined, with a 3-part definition: (1) an abstract syntax, which specifies the possible annotation structures in set-theoretical terms; (2) a semantics which specifies the interpretation of the structures defined by the abstract syntax; (3) a concrete syntax which defines an XML representation of annotation structures.

The ISO 24617-2 metamodel displayed in Figure 1 shows most of the concepts mentioned here and how they are interrelated.

3. ISO 24617-2

3.1. Rhetorical relations

Compared to the metamodel of ISO DIS 24617-2 (2010), the metamodel shown in Figure 1 additionally contains rhetorical relations as possibly[1] relating dialogue acts.

Rhetorical relations have been studied extensively in relation to written text. The parts of a coherent text are connected, explicitly or implicitly, by relations for which various terms have been used, such as 'rhetorical relations', 'coherence relations', or 'discourse relations' (e.g. (Hobbs, 1985; Mann and Thompson, 1988; Prasad et al., 2008). We will use the terms 'rhetorical relation' and 'discourse relation' interchangeably. Some of these relations, such as *Explanation, Justification*, and *Cause* are clearly semantic, whereas others, like *First, Second,... , Finally*; and *Summarizing* are more presentational in nature.

In a coherent dialogue the contributions are also connected by various relations. Two kinds of discourse relations which are specific to dialogue are (1) *functional dependence relations*, such as those between an answer and the question that it answers, or an accept apology and the apology which is accepted; and (2) *feedback dependence relations*, such as the relation between a feedback utterance like *"sure"* or a head nod, and the utterance that the feedback is about; or between *"you see?"* and the utterance that the speaker is eliciting information about. Rhetorical relations of the kind that have been studied extensively for written

[1]The specification '0..N' at the head of the arrow labelled 'rhetorical rel.' means that a dialogue act is related to zero or more other dialogue acts by a rhetorical relation.

texts also occur in spoken dialogue (see e.g. Asher and Lascarides, 2003; Tonelli et al., 2010; Petukhova and Bunt, 2011). Their occurrence in dialogue is illustrated in example (1) from the AMI corpus[2], where participant A talks about remote TV controls:

(1) 1. A: You keep losing them.
2. A: They easily slip behind or under the couch.

The events described in these sentences are semantically related by a *Cause* relation: losing remote controls is caused by them slipping behind or under the couch. In a way, the two utterances in (1), performed by the same speaker, together form a mini-discourse. Such relations can also connect utterances contributed by *different* speakers; for example, the second utterance in (1) could have been contributed by a different speaker. A slightly different case occurs in (2), where the causal connection is not between two events ('not-finding' events?), but rather between two propositions.

(2) 1. A: I can never find them.
2. B: That's because they don't have a fixed location.

Rhetorical relations between dialogue utterances do not necessarily relate the *semantic contents* of dialogue acts, but may also relate the *dialogue acts* as such, taking both their semantic contents and their communicative functions into account. The following example[3] illustrates this:

(3) 1. A: Where would you position the buttons?
2. A: I think that has some impact on many things

Utterance 2 in (3) expresses an Inform act which has a *Motivation* relation to the Question act in 1; it tells the addressees what motivated A to ask the question in 1.

In this paper we will use the term 'DA-rhetorical relation' for this type of relation, and the term 'inter-propositional relation' , which is sometimes used in the pragmatics literature, for rhetorical relations between the semantic contents of dialogue acts (irrespective of whether these contents are in fact propositions).

In the XML-based representation format of DiAML, rhetorical relations correspond to `rhetoricalLink` elements which contain two attributes whose values represent the two dialogue acts that are rhetorically related and one attribute (`@rhetoRel`) whose value contains the relation. The ISO 24617-2 annotation schema does not include a specific set of rhetorical relations, in view of the lack of a general consensus about the contents of such a set; values for the `@rhetoRel` attribute may be supplied for a specific annotation project as appropriate. In the representation (4), the `rhetoricalLink` element links the two `dialogueAct` elements representing the dialogue acts associated (through the value of the `@target` attribute) with the functional segments corresponding to the two utterances in (3). The semantic interpretation of such linking structures will be discussed in Section 4.

(4)
```
<diaml xmlns:"http://
  www.iso.org/diaml/"/>
<dialogueAct xml:id="da1" target="#fs1"
  sender="#p1" addressee="#p2"
  communicativeFunction="setQuestion"
  dimension="task"/>
<dialogueAct xml:id="da2" target="#fs2"
  sender="#p1" addressee="#p2"
  communicativeFunction="inform"
  dimension="task"
<rhetoricalLink dact="#da2"
  rhetoRelatum="#da1"
  rhetoRel="motivate"/>
</diaml>
```

3.2. Dependence relations

3.2.1. Functional dependences

In contrast to rhetorical relations, so-called functional dependence relations and feedback dependence relations, shown in Figure 1, were already part of ISO DIS 24617-2:2010. Functional dependence relations occur with dialogue acts that are responsive in nature, such as Answer, Confirmation, Agreement, Accept Apology, and Decline Offer. The semantic content of these types of dialogue act depend crucially on which previous dialogue act they respond to, and it's probably not a coincidence that they can be expressed by utterances that by themselves have no semantic content, such as *"Yes", "No thanks", "No problem",* and *"OK"*.

The use of functional dependence relations in DiAML offers the possibility to mark up a functional segment not only as expressing an answer, but also to indicate which question is being answered, and similarly for the other responsive dialogue acts. Example (6) illlustrates this for the dialogue fragment (5)[4], using a link structure to represent the functional dependence between question and answer (where the answer in this example is expressed by the discontinuous functional segment *"No (..) there isn't"*).

(5) 1. C: Is there an earlier connection?
2. A: No, I'm sorry, there isn't.

In DiAML, an annotation structure is defined as a set of *'entity structures'*, which contain semantic information about a functional segment, and *'link structures'*, which describe semantic relations between functional segments. This is illustrated by (6) (which was a valid DiAML representation according to the preliminary ISO DIS 24617-2), which consists of three entity structures, associating dialogue act information with the functional segments fs1 (*"Is there an earlier connection?"*), fs2 (*"No there isn't"*) and fs3 ("I'm sorry"), respectively, and a link structure relating A's answer to C's question.

In ISO DIS 24617-2:2010, the semantic relation represented by a link structure is either a functional dependence relation or a feedback dependence relation.

[2]See `http://www.amiproject.org/`

[3]From the AMI meeting corpus - ES2002a.

[4]From the OVIS corpus, see `http://www.let.rug.nl/~vannoord/Ovis`.

(6)

```
<diaml xmlns:
  "http://www.iso.org/diaml/"/>
<dialogueAct xml:id="e1"
  target="#fs1"
  sender="#c" addressee="#a"
  communicativeFunction=
    "propositionalQuestion"
  dimension="task"/>
<dialogueAct xml:id="e2"
  target="#fs2"
  sender="#a" addressee="#c"
  communicativeFunction="answer"
  dimension="task"/>
<functionalLink
  dact="#e2"
  functionalAntecedent="#e1"/>
<dialogueAct xml:id="e3"
  target="#fs3"
  sender="#a" addressee="#c"
  communicativeFunction="apology"
  dimension="social obligations"/>
</diaml>
```

The information-state update view of dialogue acts which underlies the standard is explicit in the specification of the semantics of DiAML in terms of update operations, to be applied to information states. Each type of dialogue act corresponds to a particular type of update operation. The semantics of an annotation structure $\alpha = \{e_1, .., e_n, L_1, .., L_k\}$, consisting of the entity structures $\{e_1, .., e_n\}$ and the link structures $\{L_1, .., L_k\}$, is defined as the successive application of the update operations corresponding to each of the entity and link structures, ordered by their functional segments (see (20) in Section 4). In example (5) - (6) the annotation structure consists of the three entity structures e_1, e_2, and e_3, corresponding to the question, the answer, and the apology, respectively, and one link structure L_1 corresponding to the functional link. Successive application of the information state update operations for each of these structures runs into the problem, however, that the update operation for e_2 cannnot be defined independently of the link structure L_1, since the link to the answer's question is needed for determining the semantic content that is negated by A saying *"No it isn't"*.

For this reason, in the final version of ISO 24617-2 the annotation of functional dependences has been changed. Functional dependences are no longer conceived as separate link structures, but as parts of entity structures. This does justice to the intuition that a functional dependence is an inherent part of a responsive dialogue act: an Answer cannot exist without a Question; an Accept Apology cannot exist without an Apology, and so on. As a result, the XML representation (6) changes into (7), in which the entity structure e_2 contains the attribute `@functionalDependence` whose value specifies the question to which e_2 contains the answer.

In Section 4, where we briefly describe the formal semantics of DiAML annotation structures, we will see that this solves the problem of dealing with functional dependence relations in a way that is both conceptually and technically adequate.

(7)

```
<diaml xmlns:
  "http://www.iso.org/diaml/"/>
<dialogueAct xml:id="e1" target="#fs1"
  sender="#c" addressee="#a"
  communicativeFunction="prop.Question"
  dimension="task"/>
<dialogueAct xml:id="e2" target="#fs2"
  sender="#a" addressee="#c"
  communicativeFunction="answer"
  dimension="task"
  functionalDependence="#e1"/>
<dialogueAct xml:id="e3" target="#fs3"
  sender="#a" addressee="#c"
  communicativeFunction="apology"
  dimension="social obligations"/>
</diaml>
```

3.2.2. Feedback dependences

For feedback dependence relations the situation is similar in that the semantic content of a feedback act necessarily depends on the utterance(s) that the feedback is about. The difference is that the semantic content of a feedback act may be determined by what was *said* before, rather than by the semantic content of a previous dialogue act.[5] The following examples illustrate this.

(8) a.
1. A: I would like to come on Tursday.
2. B: On Thursday?

b.
1. C: That's at two-thirty.
2. S: I see.

In (8a), speaker B checks the correctness of his perception of what A said, which sounded a little strange (maybe A was hesitating between Tuesday and Thursday, or maybe A was a non-native speaker who has difficulty pronouncing *th*). In (8b), speaker S expresses that he heard and understood what C said. In both cases, the feedback act refers to the immediately preceding utterance, and this happens quite frequently, however, feedback acts can also refer to utterances farther back, as in (9), or about more than utterance, as in (10).

(9)
1. A: So I can be there at 10:30.
2. A: I don't know about Peterson.
3. B: 10:30, okay.
4. B: We'll start at 10:15 with the formalities.

In (9), B's utterance 3 provides 'distant' ' feedback about A's utterance 1. In example (**??**) from the AMI corpus) participant D's utterance D1 gives 'non-local' positive feedback about the sequence of utterances B1-B4, and his utterance D2 gives negative feedback about B1-B4 plus B5.

(10) B1: We're gonna be selling this remote control for twenty five euro
B2: and we're aiming to make fifty million euro

[5]The DIT^{++} taxonomy (see Bunt, 2009 and `http://dit.uvt.nl`), on which the ISO 24617-2 scheme is largely based, distinguishes several level of processing that an feedback act may address. The ISO scheme does not make such a distinction, and considers feedback acts as underspecified for the level of processing at which they apply.

B3: so we're gonna be selling this on an international scale
B4: and we don't want it to cost more than twelve fifty euros
D1: Okay [PositiveFB:B1-B4]
B5: So fifty percent of the selling price
D2: Can we go over that again [NegativeFB:B1-B5]

The existence of distant and non-local feedback necessitates the marking up of feedback dependence relations. The fact that most feedback utterances refer to the immediately preceding utterances could conceivably be exploited by using the immediately preceding utterance as the default value in annotations, but the fine-grained segmentation approach followed in ISO 24617-2 makes it preferable to mark up feedback dependences by an attribute that has the relevant *functional segment* as its value (cf. Figure 1), which is typically a part of the last utterance contributed by the previous speaker, and which is more accurate than 'previous utterance'. Feedback dependence relations are annotated in DiAML in a similar way as functional dependence relations, as is illustrated by the representation (11) of the first three utterances in (9).

(11)
```
<diaml xmlns:
  "http://www.iso.org/diaml/"/>
<dialogueAct xml:id="e1" target="#fs1"
  sender="#a" addressee="#b"
  communicativeFunction="inform"
  dimension="task"/>
<dialogueAct xml:id="e2" target="#fs2"
  sender="#a" addressee="#b"
  communicativeFunction="inform"
  dimension="task"/>
<dialogueAct xml:id="e3" target="#fs3"
  sender="#b" addressee="#a"
  communicativeFunction="autoPositive"
  dimension="autoFeedback"
  feedbackDependence="#fs1"/>
</diaml>
```

3.3. Qualifiers

The communicative function qualifiers defined in the ISO 24617-2 standard are applicable to the general-purpose communicative functions (GPFs). Sentiment qualifiers are applicable to any GPF; conditionality qualifiers are applicable to the 'action-discussion functions' among the GPFs (such as Promise, Offer, Suggestion, Accept Offer, Decline Offer, Accept Suggestion); and certainty qualifiers are applicable to the 'information-providing functions' among the GPFs (Inform, Agreement, Disagreement, Correction, Answer, Confirmation, Disconfirmation). Dimension-specific communicative functions are typically expressed by formulaic expressions (*"Hello'', "Good morning", "Okay", "Yes", "Excuse me", "No problem", "Thank you", "Goodbye",...*) and do not admit qualifiers.

In example (12), the use of the phrasal tag *"if I'm not mistaken"* can be taken as a sign that the speaker, B, is not quite certain about the correctness of his answer. If B continues the dialogue, after A's question in 3, with the utterance 4a, then it can be decided that the answer in utterance 2 should be considered as uncertain, and should be marked up with the qualifier `uncertain`. If, by contrast, B continues the dialogue with utterance 4b, then the answer in 2 can be considered as certain after all, and marked up with the qualifier `certain`.

(12)
1. A: Do you know where the next meeting will be?
2. B: The next meeting will be in London, if I'm not mistaken.
3. A: You're not sure?
4a. B: No, not really.
4b. B: I'm pretty sure, in fact.

In (13), where the speaker directly signals his certainty or his uncertainty explicitly, the utterances should be annotated as having the communicative function (*Answer*) with the qualifiers `uncertain` and `certain`, respectively.

(13)
a. The next meeting will probably be in London.
b. The next meeting will definitely be in London.

Note that in (12.2) if B had said simply *"The next meeting will be in London"*, without the tag *"if I'm not mistaken"*, then the utterance would have been interpreted as an answer without any uncertainty. The `@certainty` attribute, used in DiAML to represent a speaker's (un)certainty (see Section 5), therefore has the default value `certain`.

Conditionality qualifiers, as illustrated in (14), behave similarly to certainty qualifiers.

(14)
1. A: Would you like to see other connections?
2a. U: Yes please.
2b. U: Only earlier connections, if you have.

With utterance 2a, participant U accepts A's offer (an Accept Offer act) unconditionally; in (14.2b) the offer is accepted conditionally. This difference can be represented in DiAML by the values *unconditional* and *conditional* of the qualifier attribute `@conditionality`. See Section 4.2 for the semantics of these qualifiers.

4. DiAML semantics

4.1. Annotation structures

The Dialogue Act Markup Language DiAML has as part of its definition an *abstract syntax*, which specifies the annotation structures defined in the standard as set-theoretical constructs, and a *concrete syntax* which defines an XML representation of these structures.

As mentioned above, an annotation structure in DiAML is a set of *'entity structures'*, which contain semantic information about a functional segment, and *'link structures'*, which describe semantic relations between functional segments. Formally, an entity structure is defined as follows:

(15) An entity structure is
1. a pair $\langle s, \alpha \rangle$ consisting of a functional segment s and a dialogue act structure α; or
2. a quadruple $\langle s, \alpha, E, \delta \rangle$ consisting of a functional segment s, a dialogue act structure α, a set of entity structures E, whose members α has a functional or feedback dependence relation to, and the element δ which specifies whether the dependence relation is functional or feedback.

A *dialogue act structure* contains the information that characterizes a single dialogue act. This includes a specification of the sender, the addressee(s), and the communicative function. For dialogue acts with a general-purpose function the dimension of the semantic content is another important component; for dialogue acts with a dimension-specific function the dimension need not be specified, since it is inherent in the definition of the function. General-purpose functions may additionally have one or more qualifiers. A dialogue act structure is therefore either a triple, consisting of a sender S, a (set of) addressee(s) A, and a dimension-specific communicative function f_d, or a quintuple containing additionally a dimension d, and one or more qualifiers q (when q is empty, we will for simplicity omit it):

(16) i. $\alpha = \langle S, A, f_d \rangle$
ii. $\alpha = \langle S, A, g, d, q \rangle$

A link structure is a triple

(17) $L = \langle \epsilon, E, \rho \rangle$

consisting of an entity structure ϵ, a non-empty set E of entity structures, and a rhetorical relation ρ, which relates the dialogue act α in ϵ to the entity structures in E.

A dialogue act structure captures the *functional* part of a dialogue act; it does not include the full semantic content but only the dimension which classifies the semantic content. The semantics of a dialogue act structure is therefore defined as a function that can be applied to a given semantic content to form the interpretation of a full-blown dialogue act. For a dialogue act without functional or feedback dependences this is expressed by (18), which defines the interpretation $I_a(\langle s, \alpha \rangle)$ of the entity structure $\langle$s,$\alpha\rangle$, that associates the dialogue act structure α with a functional segment s. This interpretation is a function applied to the semantic content $\kappa_1(s)$ of that segment.

(18) $I_a(\epsilon) = I_a(\langle s, \alpha \rangle) = I_a(\alpha)(\kappa_1(s))$

The interpretation of a dialogue act structure with a dimension-specific communicative function f_d, i.e. a triple $\langle S, A, f_d \rangle$, or with a general-purpose function g and an empty list of function qualifiers, i.e. a quintuple $\langle S, A, d, g, \langle\rangle \rangle$, is defined as the interpretation of its communicative function, applied to the interpretations of its arguments:

(19) i. $I_a(\langle S, A, f_d \rangle) = I_a(\text{f})(I_a(S), I_a(A), I_a(d))$
ii. $I_a(\langle S, A, d, f \rangle) = I_a(\text{f})(I_a(S), I_a(A), I_a(d))$

Note that of the arguments of I_a in the left-hand sides of (19), *S, A,* and d are non-recursive elements. I_a assigns to S and A certain individuals, identified in the metadata of an annotated dialogue, and to the dimension argument d, a component of an IS to be updated.

A link structure $L = \langle \epsilon, E, \rho \rangle$ is interpreted semantically as the set of updates that create rhetorical ρ-links between the representations in the participants' information states (ISs) of the dialogue act in ϵ and those in E. This assumes that the dialogue acts that occur in a dialogue are represented as such in an IS, an assumption that is shared by virtually all proposals for dialogue context modelling (see below). More specifically, it is commonly assumed that an IS has a part called the 'Dialogue History', where a record is kept of the communicative events in the dialogue, typically in the form of transcriptions of what each participant says (and does); to these representations, an interpretation is attached in terms of dialogue acts. The updates corresponding to link structures then come down to the addition of rhetorical links between these representations.

The semantics of an annotation structure $\{e_1, .., e_n, L_1, .., L_k\}$, consisting of the entity structures $\{e_1, .., e_n\}$ and the link structures $\{L_1, .., L_k\}$, is defined as the sequential application of the update functions corresponding to the constituent entity and link structures, ordered by the textual order $<_T$ of their functional segments, where the update operations corresponding to textually coinciding ('$=_T$') entity structures are unified rather than sequenced. The notation ; /$\sqcup$ is used to indicate this: formally, 'α ; /$\sqcup$ β' means that the operation α should be followed (;) by the operation β if $\alpha <_T \beta$; if $\alpha =_T \beta$ then the two operations should be unified ($\sqcup$). (See Bunt, 2011 for details and examples.)

(20) $I_a(\{e_1, .., e_n, L_1, .., L_k\}) = I_a(e_1)$;/$\sqcup$ ...;/$\sqcup$ $I_a(e_n)$;/$\sqcup$ $I_a(L_1)$;/$\sqcup$ ... ;/$\sqcup$ $I_a(L_k)$

The semantics of an entity structure with dependence relations is defined as follows, where s_ϵ is the functional segment of entity structure ϵ, and f_α the communicative function of α, κ_2 a function which combines the semantic contents of the dependent dialogues acts, and γ a similar combination function for feedback acts:[6]

(21)
a. $I_a(\langle s, \alpha, E, \text{functional} \rangle) =$
$I_a(\alpha)(\kappa_2(\kappa_1(s), \{\kappa_1(s_\epsilon) | \epsilon \in E\}, f_\alpha))$
b. $I_a(\langle s, \alpha, E, \text{feedback} \rangle) =$
$I_a(\alpha)(\gamma(s, \{\epsilon | \epsilon \in E\}, f_\alpha))$

The problem in semantically interpreting dependence relations, described in Section 3.2.1, is indeed resolved by this approach. For example, for an Answer act α expressed in the functional segment s which is functionally dependent on the Question act β, i.e. if we have an entity structure $\langle s,\alpha,\{\beta\}\rangle$, we obtain from (21) the interpretation:

(22) $I_a(\langle s_\alpha, \alpha, \{\beta\}\rangle) = I_a(\alpha)(\kappa_2(\kappa_1(s), \kappa_1(s_\beta), Answer))$

The semantics of dialogue acts being defined in terms of information-state updates, the question arises what exactly is an information state in this context; what kinds of information does it contain, and how is it represented. Various proposals have been made in this respect; e.g. Poesio & Traum (1998); Bunt (2000); Ahn (2001); Cooper (2000); Lascarides & Asher (2007; 2009), which all have in common that they assume information states (or 'context models') to have a certain internal structure, reflecting that they may contain rather diverse kinds of information. For an information state (IS) to be an adequate basis for dialogue act semantics, an obvious requirement is that, for a given range of dialogue act types, it contains those kinds of information

[6] Depending on the type of feedback act, the interpretation of feedback dependence relations may involve previous dialogue acts, their functional segments, or the semantic contents. (21) therefore refers more generally to previous entity structures.

that are updated by these dialogue acts. The DiAML semantics assumes that an IS is structured in such a way that the dimension of the semantic content of a given dialogue act can be interpreted as an indication of which IS part is to be updated. (For an IS that does not satisfy this assumption, the interpretation of a dimension would simply not impose any restriction on the parts of the IS that may be involved in a particular update.)

4.2. Qualifiers

Communicative function qualifiers come in two varieties: those that narrow down the meaning of a communicative function, called *restrictive qualifiers*, and those that add something to the meaning of a communicative function, called *additive qualifiers*. The semantics of a restrictive qualifier q_r and an additive qualifier q_a is defined by (23a) and (23b), respectively.

(23) a. $I_a(\langle f, q_r\rangle) = (I_a(f))(I_a(q_r))$
b. $I_a(\langle f, q_a\rangle) = \lambda S.\lambda z.[(I_a(f))(S, z) \sqcup (I_a(q_a))(S, z)]$

The formal semantics of the Answer function specifies an information state update operation which involves the proposition *the speaker assumes that the semantic content of the answer is true*, where *assume* is defined as the disjunction of *strong belief*, i.e. a belief that the speaker has no doubts about, and *weak belief*, i.e. a belief that the speaker is uncertain about. The update operator has a parameter which can be instantiated as *weak* or *strong*; application of (23a) has the effect of making one of these instantiations, with the effect that he speaker's *assumption* about the answer content is narrowed down to a weak or a strong belief. The conditionality qualifiers *conditional* and *unconditional* work analogously.

The 'enriching' character of additive qualifiers is reflected in their formal semantics in the fact that (23b) stipulates that they give rise to an additional part of the IS update operation defined by the qualified function, viz. the second clause of the $\sqcup$ operation. In that clause, the variable S stands for the agent who performed the dialogue act under consideration (e.g. an AcceptOffer); z stands for the semantic content; and $I_a(q_a)$ is a predicate, like HAPPY. This will result in an update of the addressee's IS such that he knows that S is 'happy', e.g. about what the addressee offered him.

5. DiAML representations in XML

As mentioned in the description of the preliminary version of ISO 24617-2 (see Bunt et al., 2010), the definition of the Dialogue Act Markup Language implements the distinction between 'annotations' and 'representations' that is made in the ISO Linguistic Annotation Framework (LAF; Ide & Romary, 2003; ISO 24612:2010), where the term 'annotation' refers to the linguistic information that is added to segments of primary data, independent of the format in which the information is represented, while the term 'representation' refers to the format in which an annotation is rendered, independent of its content. According to LAF, annotations rather than representations are the proper level of standardization.

The implementation of this distinction in the DiAML definition follows a design methodology developed in Bunt (2010), of which the basic principles are the following:

1. A distinction is made between an 'abstract syntax' and a 'concrete syntax'. The abstract syntax consists of a 'conceptual inventory', i.e. a set of basic concepts that can be used to build annotation structures, and a specification of the possible ways to construct well-formed annotation structures out of these basic concepts. This specification defines annotation structures as set-theoretical structures, independent of any particular representation format. The concrete syntax defines a format for representing the annotation structures defined by the abstract syntax.

2. The design of a representation format for annotation structures should apply the notion of an *ideal concrete syntax*, which is defined by the following properties:

(24)
- *Completeness:* every annotation structure defined by the abstract syntax can be represented by an expression defined by the concrete syntax ;
- *Unambiguity:* every representation defined by the concrete syntax is the rendering of exactly one annotation structure defined by the abstract syntax.

Given an abstract syntax specification, an ideal concrete syntax can be designed systematically, as shown in Bunt (2010), by introducing in the concrete syntax entities and relations that correspond directly to the structures and components of annotation structures as defined in the abstract syntax. This leads to representations that are optimally transparent, with conceptually clear structures and elements. The definition of DiAML in ISO 24617-2 includes an XML-based reference representation format which has been designed in this way. This examples of such representations used throughout this paper illustrate the transparency and conceptual clarity that results from applying this design methodology, showing straightforward correspondences between XML elements, attributes and values on the one hand, and entities and relations in the metamodel shown in Figure 1.

3. A formal semantics is defined for the annotation structures defined by the abstract syntax. The expressions defined by a concrete syntax inherit their semantics from that of the annotation structures which they represent. This ensures that any two representation formats, defined by an ideal concrete syntax, can be converted into each other in a meaning-preserving way.

Recently, the ANVIL tool for multimodal dialogue annotation (Kipp, 2001; 2008) has been extended with facilities for generating ISO 24617-2 annotations in the DiAML representation format (see Bunt, Kipp and Petukhova, 2012.)

6. Automatic incremental annotation

Petukhova and Bunt (2011) report on an incremental, token-based approach to the segmentation and annotation of spoken dialogue, with a focus on the recognition of their

communicative functions. This approach starts from local classifiers for individual communicative functions; automatically learned from corpus data, annotated with ISO 24617-2 functions; for AMI data this resulted in 64 classifiers. Higher-level ('global') classifiers were subsequently trained that have, along with features extracted locally from the input data, the partial outputs predicted so far by all local classifiers. Using the predictions of local classifiers for the five previous tokens and five following tokens, the F-scores were obtained shown in Table 1. These results show convincingly that the ISO 24617-2 annotation scheme can be effectively used for automatic annotation.

Dimension	*BN*	*Rip*
Task	82.6	**86.1**
Auto-Feedback	96.9	**98.1**
Allo-Feedback	**96.3**	95.7
Turn Management	90.9	**91.2**
Time Management	90.4	**93.4**
Discourse Structuring	**82.1**	78.3
Own Comm. Management	78.4	**81.6**
Partner Comm. Management	**71.8**	70.0
Social Oblig. Management	**98.6**	**98.6**

Table 1: F-scores for incremental token-based recognition of communicative functions. *BN = Bayes Net, Rip = Ripper*

7. Conclusions

The requirement that semantic annotations have a formal semantics was shown to have direct consequences for the design of annotation structures. For dialogue act annotation, the consequence is that functional and feedback dependence relations have to be expressed with attributes in the XML elements representing dialogue act structures. This turns out to simplify the representations and make them semantically fully transparent.

For the ISO standard for dialogue act annotation, this realization came just in time to prevent the proposal of incorrect annotation representations. It would seem commendable to check other standards under development for different areas of semantic annotation for their semantic foundations.

8. References

Ahn R (2001) Agents, Object and Events: A computational approach to knowledge, observation and communication. PhD Thesis, Eindhoven University of Technology.

Allen J, Core M (1997) DAMSL: Dialogue Act Markup in Several Layers (Draft 2.1). Technical Report. University of Rochester, Rochester, NY.

Asher, N., and Lascarides, A. 2003. *Logics of Conversation*. Cambridge University Press, UK.

Bunt, H. (1994) Context and Dialogue Control. *Think Quarterly* 3, 19–34.

Bunt, H. (2000a) Dialogue Pragmatics and Context Specification. In: Harry Bunt & William Black (eds.) *Abduction, Belief and Context in Dialogue*. Benjamins, Amsterdam, p. 81–150.

Bunt, H. (2009a) The DIT^{++} taxonomy for functional dialogue markup. In D. Heylen, C. Pelachaud, R. Catizone and D. Traum (eds.) *Proceedings of AAMAS 2009 Workshop "Towards a Standard Markup Language for Embodied Dialogue Acts'*, Budapest, pp. 13-24.

Bunt, H. (2010). Multifunctionality in Dialogue. *Computers, Speech, and Language* 22, 224-245.

Bunt, H. (2011). The semantics of dialogue acts. In *Proceedings of the 9th International Conference on Computational Semantics (IWCS 2011)*, Oxford, pp. 1-14.

Bunt, H., J. Alexandersson, J. Carletta, J.-W. Chae, A. Fang, K. Hasida, K. Lee, V. Petukhova, A. Popescu-Belis, L. Romary, C. Soria, and D. Traum (2010). Towards an ISO standard for dialogue act annotation. In *Proceedings of the 7th International Conference on Language Resources and Systems (LREC 2010)*, Malta, pp. 2548–2558.

Bunt, H., M. Kipp, and V. Petukhova (2012). Towards an ISO standard for dialogue act annotation. In *Proceedings of te 8th International Conference on Language Resources and Systems (LREC 2012)*, Istanbul.

ISO DIS 24617-2 (2010) Language resource management – Semantic annotation framework (SemAF), Part 2: Dialogue acts. ISO, Geneva.

Cooper R (2000) Information states, attitudes and dependent record types. In: Cavedon L, Blackburn P, Braisby N, Shimojima A (eds) *Logic, Language and Computation*, Vol 3, CSLI Publications, Stanford, pp 85–106.

Hobbs, J. 1985. On the Coherence and Structure of Discourse. CSLI Research Report 85-37.

Hovy E., Maier E (1995) Parsimonious or profligate: how many and which discourse structure relations? Unpublished manuscript.

Kipp, M. 2001. ANVIL - a Generic Annotation Tool for Multimodal Dialogue. In *Proceedings of Eurospeech 2001*, pp. 1367–1370.

Kipp, M. 2008. Spatiotemporal coding in ANVIL. In *Proceedings of LREC 2008*, Marrakech.

Mann, W. and Thompson, S. 1988. *Rhetorical structure theory: toward a functional theory of text organisation*. MIT Press, Cambridge, MA.

Petukhova, V. and H. Bunt (2011) Incremental dialogue act understanding. In *Proceedings of the 9th International Conference on Computational Semantics (IWCS 2011)*, Oxford, pp. 235-244.

Poesio M, Traum D (1997) Conversational actions and discourse situations. *Computational Intelligence* 13(3) pp. 309 – 347

Prasad, R., N. Dinesh, A. Lee, E. Miltsakaki, L. Robaldo, A.Joshi and B. Webber (2008) The Penn Discourse Treebank 2.0. In *Proceedings of LREC 2008*, Marrakech.

Tonelli, S., G. Riccardi, R. Prasad and A. Joshi (2009) Annotation of Discourse Relations for Conversational Spoken Dialogs. In *Proceedings LREC 2010*, Malta.

Traum, D. & S. Larsson (2003) The Information State Approach to Dialogue Act Management. In J. van Kuppevelt & R. Smith (eds.) *Current and new Directions in Discourse and Dialogue*, Kluwer, Dordrecht.