

The Semantics of Semantic Annotation*

Harry Bunt

Department of Communication and Information Sciences,
Tilburg University
P.O.Box 90153, 5000 LE Tilburg, The Netherlands
harry.bunt@uvt.nl

Abstract. This is a speculative paper, describing a recently started effort to give a formal semantics to semantic annotation schemes. Semantic annotations are intended to capture certain semantic information in a text, which means that it only makes sense to use semantic annotations if these have a well-defined semantics. In practice, however, semantic annotation schemes are used that lack any formal semantics. In this paper we outline how existing approaches to the annotation of temporal information, semantic roles, and reference relations can be integrated in a single XML-based format and can be given a formal semantics by translating them into second-order logic. This is argued to offer an incremental approach to the incorporation of semantic information in natural language processing that does not suffer from the problems of ambiguity and lack of robustness that are common to traditional approaches to computational semantics.

Keywords: semantic annotation, semantic interpretation, temporal annotation, reference annotation, semantic roles, underspecified semantic representation.

1. Introduction

The most interesting and challenging computer applications of natural language require the exploitation of semantic information. This is for example the case for intelligent spoken or multimodal dialogue systems, and for interactive question answering given a data base of natural language texts. Efforts to make computers exploit the semantics of utterances and texts have so far met with very limited success, however. This has two fundamental reasons: the *ambiguity problem* and the *robustness problem*.

1. The ambiguity problem: Computing the meaning conveyed by natural language expressions requires the availability of a vast amount and wide range of context information, such as knowledge of the domain of discourse, knowledge of the interactive (or other) setting in which language is used, knowledge of what occurred earlier in the discourse, knowledge of what nonlinguistic sources of information are available (e.g. shared visual context), and so on. In the absence of such information, natural language expressions are massively ambiguous; it has been estimated, for example, that a printed sentence of average length in Dutch or English has more than half a million possible readings when considered in isolation (Bunt and Muskens, 1999). Well-established methods of formal semantics, such as Montague-style or DRT-style semantics, capitalize on the context-*independent* interpretation of syntactic structure and function words, and have hardly any devices for taking context information into account. It may be noted that the ambiguity problem that arises in context-independent interpretation of natural language is to some extent artificial: it is in part caused by the aim to derive ‘disambiguated interpretations’

* Thanks to Kiyong Lee for comments on an earlier version.

within the framework of a formal logical system, which brings a level of granularity which forces one to deal with issues that are often irrelevant in practical situations.

2. The robustness problem: The methods of formal semantics tend not to be robust enough when applied to practical uses of natural language, such as spoken dialogue, on-line chatting, sms messages, or dynamic web pages. This is because linguistic semantic theories have been developed as components of grammatical theories, and have been informed primarily by the analysis of carefully constructed, grammatically perfect sentences rather than by the informal, elastic way in which language is used in spoken and multimodal interaction, which commonly involves nonsentential and grammatically irregular utterances that work semantically and pragmatically quite well.

In this paper I explore a novel approach to the computation of semantic information, namely through semantic *annotation*. I will argue and illustrate that this approach may make it possible to address both the ambiguity problem and the robustness problem successfully. I will argue that the approach offers the exciting perspective of a practical, incremental approach to the automatic use of semantic information from natural language; a use which may become more and more powerful as more sophisticated semantic annotation tools and methods are developed.

2. Semantic Interpretation and Semantic Annotation

Attempts to address the ambiguity problem include relaxing the aim of deriving fully disambiguated interpretations to the more modest aim of constructing *underspecified semantic representations*. These are partial representations of the meaning of an utterance, which leave open certain aspects of the meaning for which no or incomplete information is available. Underspecified semantic representations can be viewed as quasi-formal representations of partially disambiguated sentences; ‘quasi-formal’ in the sense of being cast in a formal syntax but not having a formal semantics. In computational work, underspecified semantic representations are treated as an intermediate stage of semantic interpretation, and can be used in computations (such as inferencing) only after being disambiguated – which takes the form of replacing the underspecified bits by fully specified parts (see e.g. van Deemter, 1996; Bunt, 2007). Underspecified semantic representations are a computationally attractive idea, but if they always have to be disambiguated before anything can be done with them, then they may help to postpone having to deal with ambiguity explosions in natural language processing systems, but they don’t really present a solution to the ambiguity problem.

Robustness problems in natural language processing, especially in syntactic processing, are often addressed by replacing the aim of producing a full syntactic disambiguation (i.e., producing a set of full parses) by that of identifying important *chunks*, such as noun phrases and preposition phrases; this is often pursued in combination with statistical or machine learning techniques. Only identifying syntactic and semantic chunks, without information about their semantic role in the sentence, can be useful for certain applications (such as spoken dialogue systems based on semantic slot filling), but is semantically primitive. Corpus-based, statistical and machine learning approaches to language processing have proved to be more robust than traditional approaches based on predefined grammars; however, while successful for syntactic processing, these approaches have so far not been applied with much success in the area of computational semantics.

The approach that is outlined in this paper is based on the observation that semantic annotations are intended to capture some of the meaning of the annotated text. Annotations have traditionally been viewed as a kind of *labels*, potentially useful for identifying linguistic patterns in corpora. But since semantic annotations capture semantic characteristics of linguistic material, it ought to be possible to interpret them as partial descriptions of the meaning of that

material. In other words, it should be possible to view semantic annotations as expressions in a language which has a well-defined semantics. In fact, the use of a semantic annotation language *without* a semantics would make little sense, since there is no reason to think that semantically undefined annotations would describe the meanings of natural language expressions any better than the expressions themselves (cf. Bunt and Romary, 2002; 2004). Still, existing work in this area, for instance on semantic role annotation (as in the FrameNet and PropBank initiatives) or on the annotation of temporal information (as in the TimeML effort) make use of uninterpreted annotation languages. (It is only recently, as part of an ISO initiative to develop annotation standards, that an effort has begun to define an annotation language for temporal information which has a formal semantics.)

While the semantic annotation schemes that have been applied so far do not have a formal semantics, I believe that it is possible to define annotation languages for such schemes which do have a well-defined semantics. In fact, defining a formal semantics for an existing annotation scheme may be helpful for improving the scheme's design.

3. Semantic Annotation Schemas

The inspiration of this paper comes mostly from participating in two recent and ongoing efforts in the area of semantic annotation, namely in the International Organisation for Standards ISO, in particular in its expert group on semantic content (<http://iso-tdg3.uvt.nl>), and in the European eContent project LIRICS (Linguistic Infrastructure for Interoperable Resources and Systems, <http://lirics.loria.fr>). One of the most important activities in the ISO expert group concerns the development of an international standard for the annotation of temporal information in documents, provisionally known as ISO-TimeML. Other activities, performed in concert with the LIRICS project, concern the design of sets of well-defined and well-documented (following ISO standard 12620) concepts for semantic annotation in an on-line registry. The focus of the latter activities is in three areas of semantic annotation: semantic roles, referential relations, and communicative functions of utterances in interactive discourse ('dialogue acts'). In this paper we focus on the interpretation of annotations concerned with temporal information, referential relations, and semantic roles.

The combination of semantic annotations for different areas requires a common, integrated format. The ISO and LIRICS activities, while aiming at the use of standardized annotation concepts, do not provide standardized *formats*. XML is a de facto standard in many NLP applications, however, and we believe that an XML-based in-line format as used in ISO-TimeML documents is slightly more readable than a stand-off format. (Stand-off representations are more expressive than in-line representations, since the latter have a problem in marking up discontinuous markables; moreover, stand-off annotations keep the annotated material unaffected, and also have the benefit of allowing multiple annotations to be linked with the same source material. Stand-off representations are therefore recommended by ISO.) We will call the XML-based semantic annotation language that we will develop in the course of this paper "SemML", and define its semantics following the familiar 'interpretation-by-translation' approach, translating SemML expressions into a well-known formal logical language.

3.1 Temporal Information

For temporal information, our point of departure is the ISO-TimeML standard under development. ISO-TimeML is a further development of the TimeML annotation language

(Pustejovsky et al. 2003; 2007) within ISO, taking other studies of temporal information into account and a wide range of natural languages (see ISO, 2007).

The following types of temporal information can be expressed in ISO-TimeML annotations:

- Times (12:25), days (Tuesday) dates (29 February), years (2007), and so on;
- Periods, such as *last week, next year, yesterday, the 20th century,...*
- Durations (*5 minutes, 2.5 hours, seven days,...*)
- The temporal anchoring of events and states: *John drove to Boston last Monday; Harry will meet Sally tomorrow at noon, Mary is pregnant since August,...*
- Temporal relations between events: *After his talk with Mary, John drove to Boston*

As an example, consider the (slightly simplified) ISO-TimeML annotation of sentence (1a), illustrating both the annotation of temporal event *anchoring* and temporal ordering of events:

(1a) After his talk with Mary, John drove to Boston

```
(1b) <TIME_STRUC>
<SIGNAL sid="s1">
After </SIGNAL>
his
<EVENT eid="e01" eiid="e1" eventclass="OCCURRENCE" pos="NOUN"
tense="NONE" aspect="NONE">
talk </EVENT>
<SIGNAL sid="s2">
with </SIGNAL>
Mary, John
<EVENT eid="e02" eiid="e2" eventclass="OCCURRENCE" pos="VERB"
tense="PAST" aspect="NONE">
drove </EVENT>
<SIGNAL sid="s3">
to </SIGNAL>
Boston
<TLINK eventInstance="e2" signalID="s1" relatedToEventInstance="e1"
tempRelType="AFTER"/>
</TIME_STRUC >
```

The ISO-TimeML draft proposal (ISO, 2007) specifies a formal semantic interpretation of the temporal markup using Interval Temporal Logic (ITL), a first-order approach to reasoning about time (see Pratt-Hartman, 2007 and <http://www.cse.dmu.ac.uk/STRL/ITL/>). On this approach, the annotation structure (1b) is interpreted as a statement about the time intervals associated with the two events mentioned in the sentence. This interpretation is represented as in (1c), where P_1 and P_2 stand for unary predicates that characterize those sets of intervals during which John talked with Mary and John drove to Boston, respectively. The interval variables should be understood to be existentially quantified.

(1c) $P_1(I_1) \wedge P_2(I_2) \wedge \text{AFTER}(I_2, I_1)$

Note that only the temporal markup is interpreted here. Temporal relations between events are interpreted as relations between temporal intervals. Other information about the events, such as who did what, is not represented (but is hidden in the predicate constants P_1 and P_2). A sentence such as (2a), stating a temporal relation between an event and a temporal interval is treated as shown in (2b) and (2c), where the predicate $P_{2004-01-31}$ should be interpreted as characterizing the (singleton) set of time-intervals coinciding with the 31st of January, 2004.

(2a) John drove to Boston on Saturday, January 31, 2004.

```
<2b><TIME_STRUC>
John
<EVENT eid="e01" eiid="e1" eventclass="OCCURRENCE" pos="VERB"
  tense="PAST" aspect="NONE">
drove
</EVENT>
<SIGNAL sid="s1">
to
</SIGNAL>
Boston
<SIGNAL sid="s2">
on
</SIGNAL>
<TEMP_ENTITY tid="t1" value="2004-01-31">
Saturday, January 31
</TEMP_ENTITY>
<TLINK eventInstance="e2" signalID="s1" relatedToTime="t1"
  tempRelType="DURING"/>
</TIME_STRUC >
```

(2c) $P_{2004-01-31}(I_1) \wedge P_2(I_2) \wedge \text{DURING}(I_2, I_1)$

Of interest is also the treatment of negation in this approach. A sentence like *John did not drive to Boston* is interpreted as “within some contextually determined interval no event of John’s driving to Boston took place. If we represent our contextually determined interval using the variable I_{ei1} (corresponding to the eiid value of the relevant EVENT tag), then we can express these truth conditions using the formula

$$\neg \exists I_1 \text{ DURING}(I_1, I_{ei1}) \wedge P_{ei1}(I_1)$$

(ISO 2007, p. 35). Note that on this approach events do not have to be represented explicitly. A predicate such as P_{ei1} , expressing that John drove to Boston during a certain interval of time, can for instance be expanded as $P_{ei1} = \lambda I. \text{DRIVE}(\text{john}, \text{boston}, I)$. The inclusion of a temporal argument in event predicates suffices for representing temporal information.

3.2 Coreference Information

Representing only the temporal information in a sentence is not very useful. Knowing that an event of a certain type occurred at a certain time is only useful when we have information about that event, such as who was involved, and in what way. As a first step towards representing the latter type of information we now turn to annotation structures for referential entities and coreference relations. We will subsequently turn to the annotation and representation of the involvement of referential entities in events, in subsection 3.3.

We will take the LIRICS annotation scheme for reference annotation as our point of departure, which is based on the reference annotation framework proposed by Salmon-Alt and Romary (2005). This scheme (see Bunt and Schiffrin, 2007), allows annotations to capture information about referential entities, coreference and anaphoric relations. Using elements from the LIRICS

scheme, we can represent the referential information in sentence (1a) in SemML as shown in (4a).¹

```
(4a) <REF_STRUC>
After
<REFENT rid="x1" animacy="ANIMATE" naturalGender="MALE" pos="PRON"
  case="GEN" cardinality="1">
his </REFENT>
talk with
<REFENT rid="x2" animacy="ANIMATE" naturalGender="FEMALE" pos="PN"
  cardinality="1">
Mary </REFENT>
<REFENT rid="x3" animacy="ANIMATE" naturalGender="MALE" pos="PN"
  cardinality="1">
John </REFENT>
drove to
<REFENT rid="x4" animacy="INANIMATE" naturalGender="NONE" pos="PN"
  cardinality="1">
Boston </REFENT>
<REFLINK referent="x1" antecedent="x3" refRelType="OBJECTAL_IDENTITY" />
</REF_STRUC>
```

A formal semantic representation of the information in this annotation structure calls for the introduction of variables ranging over individual objects other than temporal intervals, to allow the expression of the anaphoric relation through the equality of the variables corresponding to “John” and “his”. This could be done in first-order logic as follows:

(4b) $\exists x_1, x_2, x_3, x_4: \text{MALE}(x_1) \wedge \text{FEMALE}(x_2) \wedge \text{MARY}(x_2) \wedge \text{MALE}(x_3) \wedge \text{JOHN}(x_3) \wedge \text{BOSTON}(x_4) \wedge \text{INANIMATE}(x_4) \wedge x_1 = x_3$

(Stipulating that John is male, that Mary is female, and that Boston is inanimate may seem redundant but really isn’t, as can be seen by replacing “John” or “Mary” in the sentence by “Chris”, and similarly by replacing “Boston” by “Nancy”, which may refer to a female person or to a town in France.)

While the interpretation of temporal information in ISO-TimeML is accomplished in first-order logic, for information about reference this is in general not possible, since coreference relations may exist not just between individuals but also between sets of individuals. The following example illustrates this.

(5a) After they washed their hands, the men lifted the piano.

Assuming that the men washed their hands individually but lifted the piano collectively, there is no coreference relation between the entities that perform the wash-events and the lift-event, but we do have a coreference relation between the two sets of men involved in the events. As sentence (5a) is structurally identical to (1a), it could be represented by the ITL-formula (1c) $P_1(I_1) \wedge P_2(I_2) \wedge \text{AFTER}(I_2, I_1)$, with appropriate re-interpretation of the predicate terms P_1 and P_2 , but to represent the identity of the two sets of men we need to expand these predicates. When we do that, we should take into account that the definite noun phrase *the men* refers to a certain contextually determined set of men, over which a universal quantification is made; “they” refers to that same set. Let us designate such contextually determined sets by predicates with a ‘zero’ subscript (cf. the contextually determined temporal interval I_0 that we used above),

¹ The LIRICS annotation scheme supports a much richer annotation of referential entities with syntactic features that may play a role in identifying coreference relations. Here we only use a few of these features for the purpose of illustration.

so MAN_0 will designate this set of men. Using this notion, the predicates P_1 and P_2 , can be expanded as in (5b), leading to the representation (5c):

(5b) $P_1: \lambda I_1. \forall x: MAN_0(x) \rightarrow WASHANDS(x, I_1)$
 $P_2: \lambda I_2. LIFT(MAN_0, thepiano, I_2)$

(5c) $\forall x: MAN_0(x) \rightarrow WASHANDS(x, I_1) \wedge LIFT(MAN_0, thepiano, I_2) \wedge AFTER(I_2, I_1)$

Like in (1c) above, we have so far left the existential quantification over temporal intervals implicit. That quantification has to be scoped relative to the quantification over men, however; making it explicit and scoping it correctly gives the representation (5d):

(5d) $\forall x: [MAN_0(x) \rightarrow \exists I_1, I_2: [WASHANDS(x, I_1) \wedge LIFT(MAN_0, thepiano, I_2) \wedge AFTER(I_2, I_1)]]$

Note the use of the set-denoting expression MAN_0 as an argument of the LIFT predicate, which makes LIFT a second-order predicate.

Besides identity of referents (“OBJECTAL_IDENTITY“ in the annotation structure), the LIRICS annotation scheme also supports the representation of other coreference relations such as SUBSET_OF (for dealing with “...some of them...”) and MEMBER_OF (for “...one of them...”).

3.3 Semantic Role Information

For annotating semantic roles we need to mark up (1) the elements in a sentence that denote events,² in which there are participants; (2) the entities that play a role as participants in these situations; and (3) the semantic roles linking the participants to the situations that they are involved in. Using the SemML notation for marking up events, we can represent the semantic role annotation in example sentence (1a) as in (6a):

```
(6a) <SEMROLE_STRUC>
After
<REFENT rid="x1" animacy="ANIMATE" naturalGender="MALE" pos="PRON"
  case="GEN" cardinality="1">
his </REFENT>
<EVENT eid="e01" eiid="e1" eventclass="OCCURRENCE" pos="NOUN"
  tense="NONE" aspect="NONE">
talk </EVENT>
with
<REFENT rid="x2" animacy="ANIMATE" naturalGender="FEMALE" pos="PN"
  cardinality="1">
Mary </REFENT>
<REFENT rid="x3" animacy="ANIMATE" naturalGender="MALE" pos="PN"
  cardinality="1">
John </REFENT>
<EVENT eid="e02" eiid="e2" eventclass="OCCURRENCE" pos="VERB"
  tense="PAST" aspect="NONE">
drove </EVENT>
to
<REFENT rid="x4" animacy="INANIMATE" naturalGender="NONE" pos="PN"
  cardinality="1">
Boston </REFENT>
<SEMROLES anchor="e1">
  <SEMROLE participant="x1" roleType="AGENT" />
  <SEMROLE participant="x2" roleType="PARTNER" />
```

² Throughout this paper we use the term “event” in a very broad sense, including punctual as well as temporally extended events and states. We occasionally use “situation” as a synonym.

```

</SEMROLES>
<SEMROLES anchor="e2">
  <SEMROLE participant="x3" roleType="AGENT" />
  <SEMROLE participant="x4" roleType="FINAL_LOC" />
</SEMROLES>
</SEMROLE_STRUC>

```

To formally represent the content of the semantic role annotation we cannot again use the ITL-based framework that we used so far, since semantic roles are not relations between referential entities and temporal intervals, but between referential entities and events, since a referential entity may very well participate in several events that occur over the same interval, and play different roles in these events. We therefore introduce event variables in the semantic representations, and instead of using n-ary predicates as in $TALK(john, mary, I_1)$ we use unary event predicates and conjunctions to represent semantic roles, as in $TALK(e_1) \wedge AGENT(e_1, john) \wedge PARTNER(e_1, mary) \wedge DURATION(e_1, I_1)$.

The use of event variables makes temporal interval variables superfluous if we express temporal relations between events directly, rather than between the intervals during which they hold. (This has the additional advantage of staying close to the ISO-TimeML annotation, which employs temporal relations between event instances.) Time-intervals then turn up only when a sentence refers explicitly to a time, as in (2) above.

The information expressed by the SEMROLE annotation structures in (6a) can be captured in first-order event logic as in (6b):

(6b) $\exists e_1, e_2, x_1, x_2, x_3, x_4: TALK(e_1) \wedge AGENT(e_1, x_1) \wedge PARTNER(e_1, x_2) \wedge DRIVE(e_2) \wedge AGENT(e_2, x_3) \wedge FINAL_LOC(e_2, x_4)$

Like the representation of coreference relations, that of semantic roles in general requires the use of variables ranging over sets of individuals. For example, the representation of the sentence *The men lifted the piano* requires, in the collective reading, the expression of the information that the set of contextually determined men designated by “the men” plays the AGENT role in the LIFT event:

(7) $\exists e: LIFT(e) \wedge AGENT(e, MAN_0) \wedge PAST(e) \wedge PATIENT(e, thepiano)$

Allowing sets of individuals to act as participants in semantic roles in events is not limited to cases of collective quantification, but is the rule rather than the exception. For consider an sentence such as:

(8a) The men moved some of the boxes.

There is no reason to assume that the men acted purely collectively or purely individually. They may have acted partly collectively, when moving some heavy boxes, and partly individually, when moving boxes that were not heavy. Also, there is no reason to assume that the boxes were involved collectively (moved all in one go), nor that they were involved individually (moved one by one). They may have varied a lot in size or weight, and some of them may have been moved individually, others in piles. So the formal representation should allow sets of men as well as individual men in agent roles, and individual boxes as well as sets of boxes in patient roles. Moreover, the men will in all likelihood have been involved not just in one move-event, but in *a set* of such events. So we need to represent that contextually determined sets of men and boxes were involved in a set of move-events, such that some of the boxes were somehow moved and all the men were somehow involved. The agent and patient roles in these events are played by sets or individual men and boxes. To represent this correctly, we need predicates like AGENT

and THEME which are applicable to events and to *sets* of referential entities. We will continue to use AGENT, THEME, and so on as first-order predicates, applicable only to individual entities, and use AGENT*, THEME*, etc. for their second-order counterparts. For a singleton set {a} we have the obvious correspondence: $SR^*({a}) = SR(a)$ for every semantic role predicate SR. Using these second-order predicates and, for the sake of compactness, notations of restricted quantification: $\forall x \in X: P(x)$ to abbreviate: $\forall x: [X(x) \rightarrow P(x)]$, and $\exists X \subseteq Y: P(X)$ to abbreviate $\exists X: X \subseteq Y \wedge P(X)$, we can represent (8a) as:

(8b) $\forall x \in MAN_0: \exists X \subseteq MAN_0: \exists e \in MOVE: [X(x) \wedge AGENT^*(e, X) \wedge \exists Y \subseteq BOX_0: PATIENT^*(e_1, Y)] \wedge \exists z \in BOX_0: \exists Z \subseteq BOX_0: \exists e' \in MOVE: [Z(z) \wedge PATIENT^*(e', Z) \wedge \exists X' \subseteq MAN_0: AGENT^*(e_1, X')]$

A very compact representation of this may be obtained by introducing a second-order predicate expressing involvement in a set of events in a certain semantic role, defined as follows:

$INVOLV^{(2)}(x, E, X_1, R_1, X_2, R_2) =_D \exists X \subseteq X_1, e \in E, Y \subseteq X_2: [X(x) \wedge R_1'(e, X) \wedge R_2'(e_1, Y)]$

With the help of this predicate, we can rephrase (8a) as follows:

(8b) $\forall x \in MAN_0: INVOLV^{(2)}(x, MOVE, MAN_0, AGENT, BOX_0, PATIENT) \wedge \exists x \in BOX_0: INVOLV^{(2)}(x, MOVE, BOX_0, PATIENT, MAN_0, AGENT)$

It may be worth noting that the representations (8a) and (8b) are completely underspecified w.r.t. the relative scopes of the quantifications or their distribution (collective or otherwise). Yet, the representations are not underspecified from a logical point of view: they are well-formed formulas of second-order logic and can as such be used in formal reasoning.

4. Integrated Semantic Annotation and its Interpretation

4.1 General considerations

While the annotation and formal representation of temporal information seem rather well-delineated subtasks, dealing with referential information is more open-ended because it raises issues of quantification, distributivity, and scoping, and of how to deal with such issues in underspecified representations. (See Bunt, 2007) for an overview of underspecified representation techniques.) Rather than going into these issues here, in this paper we focus on the representation of just that semantic information that is captured by semantic annotations.

We also have to address the overlaps between semantic role annotation, temporal annotation and reference annotation. Besides temporal relations like AFTER and WHILE, ISO-TimeML also considers subordination links (SLINK relations). For example, in the sentence *John wants to drive to Boston* the annotation of time and events would relate the WANT and the DRIVE event as follows:

```
(9) <TIME_STRUC>
John
<EVENT eid="e01" eiid="e1" eventclass="OCCURRENCE" pos="VERB"
      tense="PRESENT" aspect="NONE">
wants </EVENT>
to
<EVENT eid="e02" eiid="e2" eventclass="OCCURRENCE" pos="VERB"
      tense="NONE" aspect="NONE">
drive </EVENT>
to Boston
<SLINK eventInstance="e1" subordinateEventInstance="e2" relType="INTENSIONAL" />
</TIME_STRUC >
```

Saying that a subordination relation exists between the WANT event and the DRIVE event is more a syntactic than a semantic observation. Semantically, the DRIVE event is a participant in the WANT event: it is what John wants, which corresponds to having the THEME role in the WANT event. Subordination links in ISO-TimeML always come with a relType value, with the possible values INTENSIONAL, FACTIVE, COUNTER-FACTIVE, EVIDENTIAL, NEGATIVE-EVIDENTIAL and CONDITIONAL. In all cases except when relType="CONDITIONAL", the subordinated event has a THEME role in the subordinating one. (The CONDITIONAL case does not correspond to any semantic role relation between the two events, but to an if-then relation.) Should integrated semantic annotation specify both the SLINK relation and the THEME relation between these events? We believe that the relation between SLINKed events is not really a temporal one, but that their THEME relation has temporal consequences which are specific for the embedding event type. Hence we propose not to annotate SLINK relations in the presence of semantic role annotations.

A related issue is the annotation of aspectual relations in ISO-TimeML using ALINK, as in *Mary started (stopped, continued,..) to laugh*. Again, there is a THEME relation with temporal consequences between the events, which are different for each aspectual verb/event.

4.2 Translating semantic annotations into formal representations

An annotation structure in SemML is a list of structures that annotate referential entities, events, (temporal) signals, temporal objects, relations that anchor events in time, temporal relations between events, semantic roles relating referential entities (or events) to events, and coreferential links between referential entities. The identifiers introduced in the various structures can be regarded as variables that have the entire annotation structure as their scope. An identifier has a unique point where it is introduced in an annotation structure. In SemML such a substructure describes either a referential entity, a temporal entity, an event, or a temporal signal.

The translation of SemML structures into the language of Second-Order Logic (SOL) requires a number of components:

1. Semantically significant SemML tags (like attributes and values) need to be translated into SOL.
2. Natural language content words (verbs, nouns, adjectives,..) have to be translated into SOL expressions (typically first-order predicates).
3. Substructures like REF_ENTITY and SEMROLES structures have to be translated into SOL.
4. Combinations of substructures have to be translated into SOL.

Components 1 and 2 naturally take the form of a SemML-SOL lexicon, while 3 and 4 require rule schemata that define a compositional translation.

For the lexical translation of SemML constants and natural language content words we will mostly use the same constants in both languages, translating for instance the SemML attribute values PAST and AGENT to the SOL predicates PAST and AGENT. Some SemML terms have other more obvious translations, for instance OBJECTAL_IDENTITY as '='. Some SemML terms do not translate to SOL as such, for instance, the feature distr="COLL" saying that a quantification is collective, corresponds in SOL to applying a second-order predicate to an argument denoting a set of individuals (a first-order predicate). Spelling out the translations of all SemML terms would require a lot of space and is, we hope, in most cases rather obvious, so we will not make that part of the translation explicit here.

For component 3, the translation of substructures, we will indicate schematic rules for a few important cases, to give a good idea of where this is going. For the substructures for temporal entities and relations, ISO-TimeML already has a partial formal semantics including a small set of translation rules into ITL, which can be reused adapting them to the use of events rather than temporal intervals (except when temporal entities are to be interpreted). For the interpretation of referential links, the translation of the annotation structures is quite straightforward. When we translate the SemML semantic roles simply to SOL predicates, things are simple as long as only first-order individuals are considered as participants in events. Complications arise mainly in relation to non-individual quantification.

For component 4, finally, we will specify a few combination rules for compositionally translating SemML combinations of substructures into SOL.

4.2.1 Translation rules for SemML substructures

The rules listed here are rule schemata, in that they contain variables in their arguments (indicated in *italics*) which can have any of their possible values. Whenever v is such a variable, v' designates its translation into SOL.

A. Rules for events, referential entities, and temporal entities

Rule EV1 (for EVENT structures)

```
<EVENT eid=E01 iid=e1 eventclass=C pos=PS tense=TE aspect=AS>
word </EVENT>
=:  $\lambda P. \exists e: \textit{word}'(e) \wedge \textit{TE}'(e) \wedge \textit{AS}'(e) \wedge P(e)$ 
```

where *word* indicates the nominal or verbal markable that is annotated, and T' represents the translation of the tense value of the verbal markable; if the *word* is nominal (denoting an event) the tense attribute will have the value NONE, which obviously does not translate to anything in the semantic representation.

Rule RE1 (for REF_ENTITY structures)

```
<REF_ENTITY rid=r1 refType="INDIV" pos="PROPER_NAME" >
NP </REF_ENTITY>
=:  $\lambda P. \exists x: \textit{NP}'(x) \wedge P(x)$ 
```

Rule RE2 (for REF_ENTITY structures)

```
<REF_ENTITY rid=r1 num="PLUR" refType="SET" >
[PRENOM] NOM </REF_ENTITY>
=:  $\lambda P. \exists x \subseteq \textit{NOM}' \wedge P(x)$ 
```

Rule RE3 (for REF_ENTITY structures corresponding to pronouns)

```
<REF_ENTITY rid=r1 refType="INDIV" pos="PERS_PRON" naturalGender=G>
word </REF_ENTITY>
=:  $\lambda P. \exists x: \textit{G}'(x) \wedge P(x)$ 
```

Rule TE1 (for TEMP_ENTITY structures)

```
<TEMP_ENTITY tid=t1 tempType="DATE" value="2007-10-31" >
expression </TEMP_ENTITY>
=:  $\lambda P. \exists t: \textit{YEAR}(t,2007) \wedge \textit{MONTH}(t,10) \wedge \textit{MONTHDAY}(t,31) \wedge P(t)$ 
```

Notice that the rules for translating substructures for events, referential entities and temporal entities all introduce an existentially quantified variable. This is important for the combination rules mentioned below.

B. Rules for semantic roles, referential relations, and temporal relations

Rule SR (for SEMROLES structures)

<SEMROLES anchor= a >
 <SEMROLE participant= $p1$ roleType= $SR1$ >
 <SEMROLE participant= $p2$ roleType= $SR2$ >
 ...
 <SEMROLE participant= pk roleType= SRk >
 </SEMROLES>
 $=: \lambda x_1, x_2, \dots, x_k, e. SR1'(e, x_1) \wedge SR2'(e, x_1) \wedge \dots \wedge SRk'(e, x_1)$

Rule RR (for REFLINK structures)

<REFLINK referent= $r1$ antecedent= $r2$ refRelType= R >
 $=: \lambda x_1, x_2. R'(x_1, x_2)$

Rule TR (for TLINK structures annotating temporal relations between events)

<TLINK eventInstance= $e1$ relatedToEvent= $e2$ tempRelType= R >
 $=: \lambda e_1, e_2. R'(e_2, e_1)$

4.2.2 Combination rules

For convenience in formulating combination rules, we introduce a particular type of function application, where a one-argument function is applied to an argument expression of the form $\lambda x_1, x_2, \dots, x_k. E(x_1, x_2, \dots, x_k)$. We define so-called “late unary application” (notation: $F \square a$) of a unary function F to such an argument as follows.

$$F \square \lambda x_1, x_2, \dots, x_k. E(x_1, x_2, \dots, x_k) =_{\text{D}} \lambda x_1, x_2, \dots, x_{k-1}. F(\lambda x_k. E(x_1, x_2, \dots, x_k))$$

For instance, if F is the function $(\lambda Q. \exists z: [WOMAN(z) \wedge Q(z)])$ (a function applicable to unary predicates) and the argument is the two-place predicate $(\lambda x_1, x_2. \exists e: TALK(e) \wedge AGENT(e, x_1) \wedge PARTNER(e, x_2))$, then late unary application of that function to this argument goes as follows, resulting in a unary predicate:

$$(\lambda Q. \exists z: [WOMAN(z) \wedge Q(z)]) \square (\lambda x_1, x_2. \exists e: TALK(e) \wedge AGENT(e, x_1) \wedge PARTNER(e, x_2)) \\ = \lambda x_1. \exists z: [WOMAN(z) \wedge \exists e: TALK(e) \wedge AGENT(e, x_1) \wedge PARTNER(e, z)]$$

We will also use two other operations. The first one, that we call “lambda insertion-application” combines a lambda abstraction $\lambda a.F$ with an expression of the form $\lambda x_1, x_2, \dots, x_k. E_1 \exists z: E_2$ into $\lambda x_1, x_2, \dots, x_k, a. E_1 \exists z: F(z) \wedge E_2$. We designate this operation by \oplus .

The second operation, that we call “cross-application” and designate by \otimes , takes two expressions of the form: $\lambda v. \exists x: E_1(v, x) \wedge E_2$ and $\lambda w. \exists y: E_1(y, w) \wedge E_3$ and merges the two into $\exists x, y: E_1(y, x) \wedge E_2 \wedge E_3$.

Rule C1. Combines an event representation (E') with a semantic roles representation (SR').
 $E' + SR' =: E' \square SR'$

Rule C2. Combines a referent representation (R') with the representation of an event plus semantic roles (ER').
 $R' + ER' =: R' \square ER'$

Rule C3. Combines the translation of a TLINK or REFLINK structure with that of the translation of an event structure or a referential entity structure.
 $T' + E' =: T' \oplus E'$

Rule C4. Combines the translations of two event structures with their participants and referential and temporal relations by means of cross-applications.

$E1' + E2' =: E1' \otimes E2'$

4.2.3 Worked example

The rules in the previous subsection are only a small subset of a larger set that will be needed to translate any meaningful SemML annotation structure into SOL. These rules are sufficient to show how the annotations of temporal information, semantic roles, and referential information in the example sentence (1a) *After he talked with Mary, John drove to Boston*, might be formally interpreted by translating it into second-order logic. For ease of reference we have labelled the relevant substructures in the example in **boldface**.

```
(10a) <SEM_STRUCT>
<SIGNAL sid="s1">
After
</SIGNAL>
R1 <REFENT rid="x1" pos="PRON" refType="INDIV" naturalGender="MALE" num="SING">
his
</REFENT>
E1 <EVENT eid="e01" eiid="e1" eventclass="OCCURRENCE" pos="NOUN"
tense="NONE" aspect="NONE">
talk
</EVENT>
<SIGNAL sid="s2">
with
</SIGNAL>
R2 <REFENT rid="x2" naturalGender="FEMALE" pos="PN">
Mary
</REFENT>
R3 <REFENT rid="x3" naturalGender="MALE" pos="NP" refType="INDIV">
John
</REFENT>
E2 <EVENT eid="e02" eiid="e2" eventclass="OCCURRENCE" pos="VERB"
tense="PAST" aspect="NONE">
drove
</EVENT>
<SIGNAL sid="s2">
to
</SIGNAL>
R4 <REFENT rid="x4" animacy="INANIMATE" pos="PN" refType="INDIV">
Boston
</REFENT>
<SEMROLES anchor="e1">
<SEMROLE participant="x1" roleType="AGENT" />
<SEMROLE participant="x2" roleType="PARTNER" />
</SEMROLES>
SR1 <SEMROLES anchor="e2">
<SEMROLE participant="x3" roleType="AGENT" />
<SEMROLE participant="x4" roleType="GOAL" />
SR2 </SEMROLES>
TR1 <TLINK eventInstance="e2" signalID="s1" relatedToEventInstance="e1"
tempRelType="AFTER"/>
RR1 <REFLINK referent="x1" antecedent="x3" refRelType="OBJECTAL_IDENTITY" />
</SEM_STRUC>
```

The referential entity substructures **R1-R4** are translated into SOL by rules **RE1** and **RE3**; the event substructures **E1** and **E2** by rule **EV1**, and the relational substructures **SR1**, **SR2**, **TR1**

and **RR1** by rules **SR**, **TR**, and **RR**, respectively. By late unary function application, Combination rule **C1** constructs SOL representations from the translations of **E1** and **SR1** and of those of **E2** and **SR2**. The TLINK substructure for the temporal relation between the two events gives rise to an additional conjunct in the representation of each of the events, through application of rule **C3**, because the two identifiers occurring in the temporal relation are introduced in the substructures of these events. Similarly, the REFLINK substructure introduces an additional conjunct in the translations of the substructures in which the identifiers are introduced that occur in the coreference relation. Repeated late unary function application (rule **C2**) adds the referential information of the four noun phrases. Finally, cross-application of the translations corresponding to the two events with their participants and their referential and temporal relations (rule **C4**), yields the final result (10b):

(10b) $\exists x_1, x_2, x_3, x_4, e_1, e_2: \text{MALE}(x_1) \wedge \text{TALK}(e_1) \wedge \text{FEMALE}(x_2) \wedge \text{MARY}(x_2) \wedge \text{AGENT}(e_1, x_1) \wedge \text{PARTNER}(e_1, x_2) \wedge \text{MALE}(x_3) \wedge \text{JOHN}(x_3) \wedge \text{DRIVE}(e_2) \wedge \text{PAST}(e_2) \wedge \text{INANIMATE}(x_4) \wedge \text{BOSTON}(x_4) \wedge \text{AGENT}(e_2, x_3) \wedge \text{FINAL_LOC}(e_2, x_4) \wedge \text{AFTER}(e_2, e_1) \wedge x_1 = x_3$

5. Reflection, Unresolved Issues, and Perspectives

In the above we have sketched an approach to the interpretation of semantic annotations, in particular for three important areas where major annotation efforts have recently been getting under way: temporal information, as treated in the project “Semantic Annotation Framework, Part 1: Time and Events” (“ISO-TimeML”) for short, carried out by an expert group within the International Organisation for Standards ISO, and semantic roles and reference information, pursued in the European project LIRICS in concert with the ISO expert group TC 37/SC 4/TDG 3 (“Semantic Content”).

We have indicated how the annotation of temporal information, semantic roles, and reference information can be integrated in an XML-based annotation language (“SemML”), and how the annotation structures in that language can be interpreted by translating them in a compositional fashion to Second-Order Logic. In passing, we changed the semantic interpretation of ISO-TimeML from being based on Interval Temporal Logic to being event-based, and we noted that the temporal relations between events encoded in ISO-TimeML by means of SLINK and ALINK structures are better encoded as semantic role relations.

Whereas the interpretation of temporal information seems to be possible in first-order logic, the interpretation of coreference relations and quantified semantic roles requires considering relations between sets of individuals, hence it needs the expressive power of second-order logic. Using second-order logic, we proposed a way of interpreting semantic annotations as a kind of underspecified semantic representations; these representations are however underspecified only in the sense of allowing finer-grained interpretations, but these representations are not underspecified in the sense that they would require further specification before being suitable for use in inferential processing. In passing, we proposed a way to represent quantifications with underspecified distribution (collective, individual, or mixed) and underspecified relative scopes in second-order logic in such a manner that they are not underspecified from a logical point of view and can be used in formal reasoning.

The approach that we have outlined is naturally extensible to other areas of semantic annotation than the ones we considered, such as non-temporal (‘rhetorical’) relations between events. This all seems quite promising and exciting, but there are still many issues to be considered in more detail. ISO-TimeML has a semantics defined only for a fragment of the annotation language, and this limitation is inherited by our reformulation of the interval-based semantics by a semantics which is mostly event-based (and which uses temporal intervals only for

interpreting expressions denoting temporal objects). We have explored only parts of what can be expressed in terms of annotations of the style discussed as ‘SemML’.

The interpretation that we have suggested of semantic roles is rather laconic in the sense that it takes a certain set of semantic roles as given, without questioning their validity and without asking how semantic annotations with these roles can be produced: Can human annotators use these roles consistently and reliably? Is automatic annotation with these roles feasible? The only evidence that we have which gives a certain confidence regarding these issues is that the set of semantic roles proposed in the LIRICS project (see Bunt and Schiffrin, 2007) and considered further in the ISO organization, has been applied successfully by untrained annotators for English and Dutch (see Bunt, Petukhova and Schiffrin, 2007).

The treatment of annotation structures for semantic roles in connection with plural NPs brings all the issues relating to quantification. In particular, the distributivity and relative scopes of quantified NPs in a sentence is nearly always only partially specified by the sentence and even not even fully by information from the context in which the sentence is used. Underspecified semantic representation of quantification is therefore very important (and the same goes for the representation of modification). Although the approach that we have indicated in this paper seems promising as a way to cast underspecified representations in a form that allows their use in reasoning without requiring further specification, this approach is currently still speculative in the sense that it has not yet been tested for its applicability to the many forms of underspecified quantification that may occur in natural language.

References

- Bunt, H. (2007) ‘Semantic underspecification: Which technique for what purpose?’ In: H. Bunt and R. Muskens (eds.) *Computing Meaning, Vol. 3*. Berlin: Springer, pp. 55 – 85.
- Bunt, H. and R. Muskens (1999) ‘Computational Semantics’. In: H. Bunt and R. Muskens (eds.) *Computing Meaning, Vol. 1*. Dordrecht: Kluwer, pp. 1 – 32.
- Bunt, H. and L. Romary (2002) ‘Towards multimodal content representation’. In: K.S. Choi (ed.) *Proc. of LREC 2002 Workshop on International Standards of Terminology and Language Resources Management*, Las Palmas, Spain, May 2002. Paris: ELRA, pp. 54 - 60.
- Bunt, H. and L. Romary (2004) ‘Standardization in multimodal content representation: Some methodological issues’. In *Proc. 4th International Conference on Language Resources and Evaluation (LREC 2004)*, Lisbon, May 2004. Paris: ELRA, pp. 2219 - 2222.
- Bunt, H. and A. Schiffrin (2007) ‘Documented compilation of semantic data categories’. LIRICS Deliverable 4.3, revised edition, September 2007.
- Bunt, H., A. Schiffrin and V. Petukhova (2007) ‘Multilingual test suites for semantically annotated data’. LIRICS Deliverable 4.4, August 2007.
- Deemter, van, K. (1996) ‘Towards a logic of ambiguous expressions’. In: Peters, S. and K. van Deemter (eds.) *Semantic Ambiguity and Underspecification*. Stanford: CSLI Press, pp. 203 – 237.
- ISO (2007). *Language resource management – Semantic annotation framework Part 1: Time and events*. ISO CD 24617-1:2007 iso_tc37_sc4_N412, Geneva: ISO.
- Pratt-Hartman, I. (2007) ‘From TimeML to Interval Temporal Logic’. In *Proceedings of the 7th International Workshop on Computational Semantics (IWCS-7)*, Tilburg, pp. 166 – 180.
- Pustejovsky, J., J. Castano, R. Ingria, R. Gaizauskas, G. Katz, R. Sauri and A. Setzer. 2003. ‘TimeML: Robust specification of event and temporal expressions in text.’ In *Proceedings 5th Int. Workshop on Computational Semantics (IWCS-5)*, Tilburg, 337 – 353.
- Pustejovsky, J., R. Knippen, J. Litman and R. Sauri. 2007. ‘Temporal and event information in natural language text.’ In H. Bunt and R. Muskens (eds) *Computing Meaning, Vol. 3*, pp. 301 – 346. Studies in Linguistics and Philosophy 83, Berlin: Springer.
- Salmon-Alt, S. and L. Romary. 2005. ‘The Reference Annotation Framework: A case for

semantic content representation.’ In: *Proceedings of the 6th International Workshop on Computational Semantics (IWCS-6)*, pp. 259 -284.