Towards formal interpretation of semantic annotation

Harry Bunt, Chwhynny Overbeeke

Department of Language and Information Science
Tilburg University, Netherlands
harry.bunt@uvt.nl, info@chwhynny.com

Abstract

In this paper we present a novel approach to the incremental incorporation of semantic information in natural language processing which does not fall victim to the notorious problems of ambiguity and lack of robustness, namely through the formal interpretation of semantic annotation. We present a formal semantics for a language for the integrated annotation of several types of semantic information, such as (co-)reference relations, temporal information, and semantic roles. This semantics has the form of a compositional translation into second-order predicate logic. We show that a truly semantic approach to the annotation of different types of semantic information raises interesting issues relating to the borders between these areas of semantics, and to the consistency of semantic annotations in multiple areas or in multiple annotation layers. The approach is incremental in the sense that it is designed to be extendable to the semantic annotation of many types of semantic information.

1. Introduction

In this paper we explore a novel approach to the incorporation of semantic information in natural language processing without falling victim to the notorious problems of ambiguity and lack of robustness, namely through the formal interpretation of semantic annotation.

This approach is based on the observation that semantic annotations are intended to capture some of the meaning of the annotated text. Annotating a text with semantic roles, for instance, is intended to capture the semantic relations between verbs and their arguments. Annotations have traditionally been viewed as a kind of labels, useful for identifying linguistic patterns in corpora. of for finding certain types of information in a text, such as temporal information in documents with temporal annotations. Since semantic annotations capture something of the meaning of the annotated text, it should be possible to do more with them then use them as mere labels; it should be possible to use them as partial descriptions of meaning. This idea has been behind several semantic annotation efforts; e.g. in describing their motivation behind the TimeML specification language for events and temporal expressions in text, Pustejovsky et al. (2003) argue that *"event recognition drives basic inferences from text,"* and claim that TimeML annotations can therefore improve the performance of text-based question answering systems. But in spite of this, TimeML is defined as an XML-based annotation language with a well-defined syntax, but without a semantics, so that there is no solid ground for allowing the "basic inferences" that motivate the effort. TimeML is no exception in this respect (the FrameNet and PropBank projects presenting other examples); it is just one instance of the general trend in the design of semantic annotation systems to take their semantics for granted.

There is in fact something paradoxical about using a semantic annotation language *without* a semantics, since there is *a priori* no reason why semantically undefined annotations would describe something of the meanings of natural language expressions any better than the expressions themselves. Bunt & Romary (2002) have therefore argued that one of the fundamental requirements of a language for semantic annotation is that it should have a formal semantics. In this paper we explore the possibility of designing annotation languages and schemes which do have a well-defined semantics. We will show that defining a formal semantics for an annotation scheme/language may have the additional advantage of improving its design, compared to considering designs which have only a syntactic definition.

The inspiration for this paper comes from participating in two recent and ongoing efforts in the area of semantic annotation, namely the European eContent project LIRICS (Linguistic Infrastructure for Interoperable Resources and Systems, http://lirics.loria.fr), and the expert group on semantic content (http://iso-tdg3.uvt.nl) of the International Organisation for Standardization ISO. One important activity developed by the ISO expert group concerns a project to define an international standard for annotating temporal information, based on TimeML and therefore informally known as ISO-TimeML. Other activities, performed in concert with the LIRICS project, concern the design of sets of well-defined and well-documented concepts (following ISO standard 12620) for semantic annotation, which are made publicly available 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 dialogue utterances.

This paper is concerned with the interpretation of annotations in the areas of temporal information, referential relations, and semantic roles. We propose an XML-based integrated format for the representation of annotations concerned with these types of information, inspired by the TimeML-format.[1] We refer to the language for the integrated annotation of this information as "SemML" (Semantic Markup Language), and we define a compositional formal semantics for this language by translating SemML ex-

---

[1] The use of this format is just a matter of convenience for the presentation in this paper. In fact, the XML format is of secondary importance, and the formal semantics should be defined at a more abstract level – cf. ISO (2008).

pressions in a compositional manner into a formal logical language (that of Second-Order Predicate Logic, to be precise) that has a well-established formal semantics.

This paper is organized as follows. In the next section we consider briefly the annotation of temporal and event information according to ISO-TimeML, and its proposed semantics in terms of Interval Temporal Logic. In particular we will indicate some deficiencies and limitations of that semantics, suggesting that an event-based semantics is preferable to a semantics based on temporal intervals. In section 3 we consider the annotation of semantic role and reference information in an XML-based format, similar to that of TimeML, using the data categories defined in the LIRICS project. We argue that the interpretation of quantified semantic role information and/or reference information cannot be done in an adequate fashion in first-order logic, and we show how second-order logic can be used. In section 4 we discuss the effects of integrating the annotation of time, semantic roles, and reference in a single representation structure. We point out the existence of certain overlaps between the annotation concepts used in ISO-TimeML, those used for semantic role annotation, and those used for reference annotation. In section 5 we present a compositional semantics for the integrated SemML language. We discuss the problems with devising a compositional translation from SemML to formal logic, and present a solution to these problems. The paper concludes in section 6 with some remarks on what has been achieved in this paper, on its limitations, and on its perspectives.

2. Temporal information

For temporal information, our point of departure is the ISO-TimeML standard under development (see ISO, 2007), which is a further development within ISO of the TimeML annotation language (Pustejovsky et al., 2003; Pustejovsky et al., 2007). ISO-TimeML annotations can deal with the mark-up of temporal objects (like dates: *29 February 2007*; times: `11:25 a.m.`; days: *Tuesday*; periods: *last week; yesterday; the 20th century*; durations: *5 minutes, 2.5 hours*); of the temporal anchoring of events and states: *John was away last Monday; Mary is pregnant since August*; and of (temporal) relations between events: *After his talk with Mary, John drove to Boston; mary started to laugh; John saw mary smiling*. 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)   <SIGNAL sid="s1">
       After </SIGNAL>
       his
       <EVENT eid="e01" eiid="e1"
       class="OCCURRENCE" pos="NOUN"
       tense="PAST" aspect="NONE">
       talk </EVENT>
       with Mary, John
       <EVENT eid="e02" eiid="e2" tense="PAST"
       aspect="NONE" class="OCCURRENCE"
```

```
       pos="VERB">
       drove </EVENT>
       to Boston
       <TLINK eventInstance="e2" signalID="s1"
       relatedToEventInstance="e1"
       tempRelType="AFTER"/>.
```

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-Hartmann (2007) and http://www.cse.dmu.ac.uk/STRL/ITL/). On this approach the structure (1b) is interpreted as a statement about the time intervals associated with the events mentioned in the sentence. This interpretation is represented in (1c), where P1 and P2 are unary predicates that characterize those sets of intervals during which John talked with Mary and drove to Boston, respectively. The interval variables are to be understood as existentially quantified.

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

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"*. Representing our contextually determined interval using the variable I_0, we can express these truth conditions using the formula

(2) $\neg \exists I_1: \text{DURING}(I_1, I_0) \wedge P_{e1}(I_1)$

(ISO 2007, p. 35). Note that on this approach events do not have to be represented explicitly. A predicate such as P_{e1}, expressing that John drove to Boston during a certain interval of time, can for instance be expanded as $P_{e1} = \lambda I$. DRIVE(john, boston, I). The inclusion of a temporal argument in event predicates suffices for representing temporal information.

This semantics for ISO-TimeML shows that it is possible in principle to formulate a formal interpretation of temporal annotations, but it also has important limitations. It is limited in being defined only for a fragment of the ISO-TimeML language, not including for instance tense, aspect and subordination relations among events. It has a more fundamental limitation due to its basis in Interval Temporal Logic. This means that all semantic information in this approach is interpreted in terms of properties of temporal intervals. This limits the kinds of information about events that can be expressed. For instance, the subordination relation between a WANT and a HELP event expressed in the sentence *He wanted to help* cannot be reduced to a relation between the temporal intervals during which these events occurred. Also, when we want to annotate and interpret other than temporal information, such as semantic roles and coreference relations, then obviously we need a richer underlying ontology than that of ITL.

A deficiency of the ITL-based semantics, at least in the form in which it has been formulated, is that it is not compositional. The semantics of a language is compositional if it allows the interpretation of every expression to be computed recursively from the meanings of its subexpressions

plus its syntactic composition. The ITL-based semantics has a two-stage nature, consisting of (1) a translation from ISO-TImeML into ITL; (2) the formal semantics of the ITL translation. Since ITL, being a first-order predicate language, has a compositional semantics, the compositionality of the ISO-TimeML semantics is determined by the 'compositionality' of the translation into ITL. A translation may be called compositional iff every expression can be translated into the target language in a recursive manner, depending only on the translations of its subexpressions and its syntactic composition. Since the ISO-TimeML translation into ITL has been defined in such a way that the translation of a subexpression of a given expression depends on the translations of other subexpressions, this translation is not compositional. Katz (2007) and Lee (2008), who have also presented attempts to provide a semantics to temporal annotations, run into the same problem.[2]

3. Coreference and semantic roles

For the annotation of coreference and semantic roles we take the LIRICS annotation schemes (Bunt and Schiffrin, 2007) as our point of departure. These schemes allow annotations to capture information about referential entities, various types of (co-)reference relations among referential entities, and semantic roles relating events to their participants. Using these data categories we can represent the referential and semantic role information in sentence (1a) as follows in SemML.[3]

(3) After
```
<REFENTITY rid="x1" animacy="ANIMATE"
naturalGender="MALE" pos="PRON"
case="GEN" cardinality="1">
his </REFENTITY>
<EVENT eid="e01" eiid="e1" class
="OCCURRENCE" pos="NOUN"> talk /EVENT>
with
<REFENTITY rid="x2" animacy="ANIMATE"
naturalGender="FEMALE" pos="PN"
cardinality="1"> Mary </REFENTITY>
<REFENTITY rid="x3" animacy="ANIMATE"
naturalGender="MALE" pos="PN"
cardinality="1">
John </REFENTITY>
<EVENT eid="e02" eiid="e2" tense="PAST"
aspect="NONE" class="OCCURRENCE"
pos="VERB"> drove </EVENT>
to
<REFENTITY rid="x4" animacy="INANIMATE"
naturalGender="NONE" pos="PN"
cardinality="1"> Boston </REFENTITY>
<REFLINK referent="x1" antecedent="x3"
refRelType="IDENTITY" />
<SEMROLES anchor="e1">
```

---

[2] Kiyong Lee has adopted the solution outlined below (section 5) to deal with this problem.

[3] The LIRICS annotation scheme supports a much richer annotation of referential entities various other features than is shown here. Here we only use a few features for the purpose of illustration.

```
<SEMROLE participant="x1"
roleType="AGENT" />
<SEMROLE participant="x2"
roleType="PARTNER" />
</SEMROLES>
<SEMROLES anchor="e2">
<SEMROLE participant="x3"
roleType="AGENT" />
<SEMROLE participant="x4" roleType=
"FINAL_LOC" />
</SEMROLES> </SEMROLES>
```

A formal representation of the referential information in this structure clearly goes beyond the possibilities of ITL, as it calls for the representation of individual entities other than temporal intervals. This is easily done in first-order predicate logic, representing anaphoric relations as relations between existentially quantified individual variables

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

Since semantic roles are relations between referential entities and events, the formal representation of the content of semantic role annotations requires the use of events as a sort of individuals, as is standard practice in neo-Davidsonian semantics. We will do this by means of unary event predicates that denote classes of events, and binary predicates representing semantic role relations, as in:

TALK$(e_1) \wedge$ AGENT$(e_1,$john$) \wedge$ PARTNER$(e_1,$mary$) \wedge$ DURATION(e_1,I_1).

(Note that we use semantic role *predicates*, as in AGENT$(e_1,$john$)$ rather than functions, as in AGENT$(e) =$ john, since we do not want to exclude that possibility that an event has more than one participant in the same semantic role.)

Using this format, the information expressed by the SEMROLE annotation structures in (2) can be captured in first-order event logic as in (5):

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

The interpretation of information about reference and semantic roles cannot always be accomplished in first-order logic. For instance, coreference relations may exist not just between individuals but also between sets of individuals. The following example illustrates this.

(6a) After washing their hands, the men lifted the piano.

There is a coreference relation between the set of men who collectively lifted the piano and whose members individually washed their hands. To represent the identity of the two sets of men we need to be able to consider these sets as second-order individuals, to which second-order predicates apply. In addition, we should take into account that the expression "the men" quantifies over the *set of contextually*

determined men, rather than over the set of *all men*. This is a phenomenon that occurs ubiquitously in natural language when definite expressions are used. To designated sets of contextually salient individuals we will use predicates with the subscript '0'; e.g. MAN_0 designates the set of contextually salient men. This leads to the following representation:

(6b) $\exists$ e $\in$ LIFT: PAST(e) *wedge* AGENT(MAN_0,e) $\wedge$ THEME(thepiano,e) *wedge* $\forall$x: [MAN_0(x) $\rightarrow$ $\exists$ e': [WASHANDS(e') $\wedge$ AGENT(x,e') $\wedge$ $\wedge$ AFTER(e,e')]]

This example also shows that the interpretation of semantic roles annotation, like that of coreference relations, forces us to consider sets of individuals as arguments of predicates and quantifiers, like the set MAN_0 in an AGENT role.

The annotation and interpretation of referential and semantic role information touches on issues of quantification, distributivity (collective versus individual participation in events), and scoping. A full discussion of these issues requires a more extensive investigation into the annotation of quantification phenomena than has been undertaken so far, and goes beyond the scope of the present paper; see also Bunt (2007).

4. Integrated annotation

Annotations of temporal information like (1b) and of coreference and semantic roles like (3) can be merged into a single integrated SemML annotation structure in a straightforward fashion, as far as building XML representations is concerned, but there are some conceptual issues to consider when doing so. This becomes clear when we define a formal interpretation of these structures. For example, semantic roles are translated into two-place predicates denoting a relation between an eventuality (state or event) and a participant in that eventuality. However, sometimes semantic roles are occupied by events, as in *Mary expects John to come home before seven o'clock.* The EXPECT eventuality has a COME event as its THEME. In (ISO-)TimeML, such relations between eventualities are annotated by SLINK (subordination link) tags. This is all right as long as only temporal information is annotated and relations between eventualities, but when semantic roles are also annotated this creates a problem: it would be wrong the construe the relation between the two eventualities as *both* a subordination relation and a thematic relation. Since this relation is not primarily temporal in nature, in the case of integrated semantic annotation no SLINK should be used in such cases, only the semantic role.

The reverse situation occurs when the temporal anchoring of eventualities is annotated. Although semantic roles are usually defined as relations between an eventuality and a participant in that eventuality (Bunt & Schiffrin, 2007), most semantic role annotation schemes (such as FrameNet and PropBank) annotate the moment or period when an eventuality occurs also as having a semantic role. (And the same goes for temporal qualifications like durations, and for locations and locational qualifications like *from Paris to Rome* or *along the Arno river*). But when both temporal annotation and semantic role annotation are performed and expressed in integrated annotation structures, then temporal anchoring relations are better regarded as temporal, and dropped as semantic roles.

Aspectual relations and temporal anchoring also need to be reconsidered when performing integrated semantic annotation. For example, in a case like *Amy started to laugh*, ISO-TimeML annotates an aspectual link (ALINK) as follows (simplified):

```
Amy
<EVENT eiid="e1" tense="PAST">
started /EVENT>
to
<EVENT eiid="e2" tense="PAST">
laugh /EVENT>
<ALINK eventInstanceID="e1"
relatedToEvent="e2" relType="INITIATES"/>
```

The relation indicated by relType="INITIATES" is intended to express that the eventuality e2 starts when the eventuality e1 occurs. However, this is only the temporal aspect of the relation between the two events; another aspect is that the LAUGH event is the THEME of the START event: it is what was started. So here we have a semantic relation that cuts across the boundary between the areas of temporal and thematic annotation and interpretation.

5. Formal interpretation of SemML

The interpretation-through-translation of SemML structures into the language of Second-Order Logic (SOL) requires (1) the translation of SemML constants, like attributes and values; (2) that of SemML subexpressions like EVENT, REFENTITY, SEMROLE and TLINK structures; (3) the combination of the results of (1) and (2). To make the translations more readable, we often also translate the natural language expressions that are marked up, as in (5); this is done in an intuitively obvious way.

5.1. Interpreting XML attributes and values

Where possible, we will use the same names for SemML attributes and values in SOL, translating for instance the SemML values MALE and THEME to the SOL predicates MALE and THEME. This applies notably to all semantic role values in SemML.[4] This has the advantage that our translation of semantic role annotations is independent of a particular choice of semantic roles; it can be applied to any set of semantic roles. Some SemML terms have other obvious translations, for instance the coreference relation indicated by relType="IDENTIY" is translated as equality. By way of illustration, the tables 1 and 2 show the translations of the values of the naturalGender attribute in REFENTITY tags for personal pronouns and of the rlink values in coreference tags.

5.2. Interpreting SemML tags

A SemML expression, represented in XML, is a list of tags that annotate referential and temporal entities, events, sig-

---

[4]With the exception of MANNER and ATTRIBUTE, of which we use the values as unary predicates.

XML value	Translation
"MALE"	λx. MALE(x)
"FEMALE"	λx. FEMALE(x)
"NONE"	λx. $\neg$ANIMATE(x)

Table 1: Translation table for the `naturalGender` attribute in REFENTITY tags.

XML value	Translation
PART_OF	λx. λy. PART_OF(x,y)
SUBSET_OF	λx. λy. x $\subseteq$ y
MEMBER_OF	λx. λy. x $\in$ y
IDENTITY	λx. λy. x = y
UNSPECIFIED	λx. λy. UNSPECIFIED_REL(x,y)

Table 2: Translation table for RLINK tags.

nals, relations that anchor or interrelate events in time, semantic roles, and referential links. The following cases exemplify the translation rules for these tags.

The translation of eventuality tags depends on their polarity and is defined by one rule for each case:

```
<EVENT eiid="e" tense=T aspect=A
polarity="POS"> ⤳
```
λE. λP. $\exists$e $\in$ E: P(e) $\wedge$ T'(e) $\wedge$ A'(e)

```
<EVENT eiid="e" tense=T aspect=A
polarity="NEG"> ⤳
```
λE. λP. $\neg\exists$e $\in$ E: P(e) $\wedge$ T'(e) $\wedge$ A'(e)

The tense and aspect values in these tags are translated into constraints on the temporal properties of the event under consideration; for details see Bunt & Overbeeke (2008).
For tags marking up referential entities there are a few translation rules, e.g. a separate one for personal pronouns. The most important rule is:

```
<REFENTITY rid="x" refType="SET"
pos="PERSPRON" naturalGender=G
animacy=A num="PLUR"> ⤳
```
λP. $\exists$X: $\forall$x: (x $\in$ X $\rightarrow$ G'(x) $\wedge$ A'(x)) $\wedge$ P(X)

For the annotation of referential links, semantic roles, and temporal anchoring the translation is quite straightforward. For example, the following translation rule applies to semantic role annotations:

```
<SEMROLES semID=r1 anchor=e1>
<SEMROLE referentID=x1 distr="INDIV"
roleType=R1>
<SEMROLE referentID=x2 distr="INDIV"
roleType=R2>
</SEMROLES> ⤳
```
λx. λy. λe. R1'(X,E) $\wedge$ R2'(y,e)

where R1' and R2' are the translations of R1 and R2, respectively.

5.3. Interpretation of complete annotations

In order to translate an entire SemML annotation, as represented in XML, into a formal representation in SOL, we need to combine the translations of the XML tags correctly and in a compositional way. The compositionality of the translation is not only a standard methodological requirement of a formal semantic specification, but is also of practical importance for defining a manageable translation process. A compositional definition allows any well-formed subexpression of a complex annotation structure to be translated independent of the rest of the structure.

A key question for designing a correct and compositional translation is: How do we know the translations of which tags should be combined? For simple sentences with only one eventuality this seems trivial, but even then one needs to decide for each referential entity whether it is a participant in that eventuality and should be assigned a semantic role. The annotation structures indicate this by the identifiers of REFENTITY substructures that are used in the semantic role tag with that eventuality as the anchor. For example, in the sentence *The center of Los Angeles was destroyed by the earthquake*, the referential entity that participates in the eventuality in a PATIENT role is the referent of *the center of Los Angeles*, not the entity LA. This is represented in the annotation structure by the occurrence of x1 as value of the `referentID` attribute in the SEMROLE tag:

```
(7) <REFENTITY rid="x1"
    refType="INDIV">
    <REFENTITY rid="x2"
    refType="INDIV">
    The center </REFENTITY>
    of
    <REFENTITY rid="x3" refType="INDIV"
    pos="PROPNAME">
    Los Angeles </REFENTITY>
    </REFENTITY>
    <EVENT eiid="e1" tense="PAST"
    aspect="PERFECTIVE">
    was destroyed </EVENT>
    by
    <REFENTITY rid="x4"
    refType="INDIV">
    the earthquake </REFENTITY>
    <SEMROLES semID="s1" anchor="e1">
    <SEMROLE referentID="x1"
    roleType="PATIENT">
    <SEMROLE referentID="x4"
    roleType="CAUSE">
    </SEMROLES>
```

When we translate the various tags separately into logical formulas, we lose the linking information since the formulas contain only bound variables instead. In the case of example (7) the SEMROLES tag translates to λx. λe. AGENT(x,e).

This formula should be combined with the translations of *The center of Los Angeles* and *was destroyed*, but the information that that combination should be made, has been lost in translation.

In order to combine the translations of the XML tags correctly, we need to keep track of the linking identifiers in the annotation structures. A way of doing this is to extend the translations from logical formulas to more complex structures containing the necessary linking information. To this end, we extend the SOL formulas with two elements which we call the *combination index* and the *label index* respectively. The combination index is a list of the XML identifiers that identify the SemML subexpressions whose translations should be combined with the formula; the label index is a list of elements that have already been used in constructing it. Moreover, in order to take quantifier scopes correctly into account when we combine formulas, we label all quantified subformulas (like is often done in underspecified semantic representations), and we pair the identifiers in a label index with the index of the quantified subexpression where they are introduced. Thus the representation will be of the form:

(8) $<<x_1,..x_k>,<<y_1,L_1>,...,<y_n,L_n>>,\varphi>$

The translation rules of the XML tags are adapted to this triple format. The translations of the EVENT, REFENTITY, and TIMEX3 tags will have an empty combination index and a label index containing their XML identifier plus their SOL quantifier label. The translations of SEMROLES, RLINK, TLINK, SLINK and ALINK tags will have an empty label index and a combination index containing the XML identifiers used in `eventInstanceID`, `relatedToEventInstance`, `subordinatedEventInstance`, `referentID`, `relatedToReferent`, `timeID`, or `relatedToTime` attributes of these tags. The translation process has been completed when a triple has been built with an empty combination index and where the translations of all linked XML tags have been used to construct the logical formula. The translations of complex subexpressions of SemML annotations are built by applying a number of operations, defined in (Bunt, 2007). These operations can be applied in any order. The operations are the following.[5]

1. A kind of function application called *late unary application*, where a unary function F is applied to an argument expression of the form $\lambda x_1,...,x_k. E(x_1,...,x_k)$:

$F \,\square\, \lambda x_1,...,x_k \,\lambda a.\, \text{EXPR} = \lambda x_1,...,x_k\, F(\lambda a.\, \text{EXPR})$

2. An operation called *lambda insertion-application* (designated by $\oplus$) is defined, which combines a lambda abstraction $\lambda a.\, F$, where F is a function, with an expression of the

---

Besides these operations, an additional one is so-called lambda cross-application, which is not used in this paper and therefore left out of consideration.

form $\lambda x_1,...,x_k.\ \alpha\ \exists z: \beta$, where β does not contain any quantifiers, into $\lambda x_1,...,x_k.\ \lambda a.\ \alpha\ \exists z: F(z) \land \beta$.

3. An operation called *merge-application* (designated by $\odot$), is defined for any two representations $E_1 = <A_1, B_1, \alpha>$ and $E_2 = <A_2, B_2, \lambda z_1,z_2,...,z_k.\ \beta>$, where the combination index A_2 is a sublist of the list of first elements in the pairs constituting the label index B_1; α contains a leat one (generalized) quantifier, and β is not of the form $\lambda x.\ \psi$. Merge-application moves the formula in E_2 inside the scope of the quantifier Q in E_1 that corresponds to the item in the combination index that has narrowest scope. So if α is a formula of the form $\gamma\ L_i: Q_i x_i\ \delta$, with quantifier Q, then the formula resulting from merge-application is $\gamma\ L_i: Q_i x_i\ [\lambda z_1,z_2,...,z_k.\ \beta](x_i)(z_2)...(z_k) \land \delta$.

These operations, defined for formulas, now need to be extended to triples. For operations on triples we will use the same symbols, but in the triple definitions we subscript them with a '3' to indicate that they apply to triples. Note that '·' is used to indicate the concatenation of lists, and '-' the subtraction of lists.

Late unary application

$<x_1|a_1, B_1, \varphi_1> \square_3 <A_2, <x_1,L_1>|b_2, \varphi_2> \rightsquigarrow$
$<a_1, <x_1,L_1>|b_2, \varphi_1 \square \varphi_2>$

Lambda insertion-application

$<ci_1, li_1, \varphi_1> \oplus_3 <<>, li_2, \varphi_2> \rightsquigarrow$
$<ci_1\text{-}li_2, li_1 \cdot li_2 \cdot ci_1, \varphi_1 \oplus \varphi_2>$

Merge-application

$<A_1, <<x_1,L_1>,...,<x_i,L_i>>, \varphi> \odot_3 <<x_1,...,x_k>, B_2, \psi>$
$\rightsquigarrow <A_1 \cdot <x_{i+1},...,x_k>, B_1 \cdot B_2, \varphi \odot \psi>$

5.4. Worked Example

We show how the translation of a complex SemML annotation structure into SOL is achieved by means of these operations by considering the step-by-step translation of example (9), which combines the annotation of semantic roles, reference relations, temporal anchoring, and temporal ordering of eventualities, In the description we use **E'** for the translation of an EVENT tag, **R'** for the translation of a REFENTITY tag, **TL'** for the translation of a TLINK tag, and so on.

(9) *Igor called Amy at noon, before she saw him.*
```
<REFENTITY rid="x1" refType="INDIV"
pos="PROPERNOUN" animacy="ANIMATE"
num="SING" definiteness="DEF">
Igor </REFENTITY>
<EVENT eiid="e1" tense="PAST"
polarity="POS">
called </EVENT>
<REFENTITY rid="x2" refType="INDIV"
pos="PROPERNOUN" animacy="ANIMATE"
num="SING" definiteness="DEF">
Amy </REFENTITY>
```

```
at
<TIMEX3 tid="t1" type'TIME"
value="T12:00">
noon </TIMEX3>
before
<REFENTITY rid="x3" refType="INDIV"
pos="PERSPRON" naturalGender="FEMALE"
animacy="ANIMATE" num="SING"
definiteness="DEF">
she </REFENTITY>
<EVENT eiid="e2" tense="PAST"
polarity="POS">
saw </EVENT>
<REFENTITY rid="x4" refType="INDIV"
pos="PERSPRON" naturalGender="MALE"
animacy="ANIMATE" num="SING"
definiteness="DEF">
him </REFENTITY>.
<SEMROLES semid="p1" anchor="e1">
<SEMROLE referentID="x1" distr="INDIV"
roleType="AGENT"/>
<SEMROLE referentID="x2" distr="INDIV"
roleType="THEME"/>
</SEMROLES>
<SEMROLES semid="p2" anchor="e2">
<SEMROLE referentID="x2" distr="INDIV"
roleType="PIVOT"/>
<SEMROLE referentID="x2" distr="INDIV"
roleType="THEME"/>
</SEMROLES>
<TLINK eventinstanceID="e1"
signalID="s2" relatedToEventIntance="e2"
relType="BEFORE">
<RLINK referentID="x3"
relatedToReferent="x2"
relType="IDENTITY"/>
<RLINK referentID="x4"
relatedToReferent="x1"
relType="IDENTITY"/>
```

We start[6] with the first SEMROLES tag and event tag, and combine their translations by means of late unary application. (TA standards for 'temporal anchoring'; T_0 for the utterance's speech time.)

SR1' $\rightsquigarrow$ $<<e1,x1,x2>, <>, \lambda z.\lambda y. \lambda a.$ AGENT(y,a) $\wedge$ THEME(z,a)>
E1' $\rightsquigarrow$ $<<>, <<e1,L1>>, \lambda Q.$ L1: $\exists e_1 \in$ CALL: Before(TA(e_1),T_0) $\wedge$ Q(e_1)>

SR1' □ E1' $\rightsquigarrow$ $<<x1,x2>, <<e1,L1>>, \lambda y.$ L1: $\exists e_1 \in$ CALL: Before(TA(e_1),T_0) $\wedge$ AGENT(y,e_1) $\wedge$ THEME(z,e_1)>

We translate the TIMEX3 and TLINK tags, again combine their translations by late unary application, and combine the result by merge-application with that of translating the

---
[6]Just for convenience. Tags can be translated and their translations combined in any order.

---

event plus semantic roles. For convenience we abbreviate the SOL representation of *noon* by a unary predicate NOON.

T1' $\rightsquigarrow$ $<<>, <<t1,L2>>, \lambda P.$ L2: $\exists t_1$: NOON(t_1) $\wedge$ P(t_1)>
TL1' $\rightsquigarrow$ $<<e1,t1>, <>, \lambda a. \lambda b.$ TA(a) = b>

T1' □ TL1' $\rightsquigarrow$ $<<e1>, <<t1,L2>>, \lambda a.$ L2: $\exists t_1$: NOON(t_1) $\wedge$ TA(a) = t_1>

(SR1' □ E1') ⊙ (T1' □ TL1') $\rightsquigarrow$ $<<x1,x2>, <<t1,L2>,<e1,L1>>, \lambda y.$ L1: $\exists e_1 \in$ CALL: Before(TA(e_1),T_0) $\wedge$ AGENT(y,e_1) $\wedge$ L2: $\exists t_1$: NOON(t_1) $\wedge$ TA(e_1) = t_1>

We complete the translation of *Igor called Amy at noon* by combining the translations of the REFENTITY tags of the two participants with late unary application:

R1' □ (R2' □ (SR1' □ E1') ⊙ (T1' □ TL1')) $\rightsquigarrow$ $<<>, <<t1,L2>,<e1,L1>,<x1,L3>,<x2,L4>>,$ L3: $\exists x_1 \in$ IGOR$_0$: ANIMATE(x_1) $\wedge$ L4: $\exists x_2 \in$ AMY$_0$: ANIMATE(x_2) $\wedge$ L1: $\exists e_1 \in$ CALL: Before(TA(e_1),T_0) $\wedge$ AGENT(x_1,e_1) $\wedge$ THEME(x_2,e_1) $\wedge$ L2: $\exists t_1$: NOON(t_1) $\wedge$ TA(e_1) = t_1>

To translate *She saw him*, we combine the translations of the EVENT tag and the SEMROLES tag as before:

SR2' □ E2' $\rightsquigarrow$ $<<x3,x4>, <<e2,L5>>, \lambda a. \lambda b.$ L6:$\exists e_2 \in$ SEE: Before(TA!(e_2),T_0) $\wedge$ AGENT(a,e_2) $\wedge$ THEME(b,e_2)>

Before combining this with the translations of the participants' REFENTITY tags, we combine the latter with the RLINK tags indicating their referential antecedents:

R3' $\rightsquigarrow$ $<<>, <<x3,L6>>, \lambda Q.$ L6: $\exists x_3 \in$ FEMALE$_0$: Q(x_3)>
R4' $\rightsquigarrow$ $<<>, <<x4,L7>>, \lambda R.$ L7: $\exists x_4 \in$ MALE$_0$: R(x_4)>
RL1' $\rightsquigarrow$ $<<x3,x2>, <>, \lambda z_1. \lambda z_2. z_1 = z_2$>
RL2' $\rightsquigarrow$ $<<x4,x1>, <>, \lambda u_1. \lambda u_2. u_1 = u_2$>

R3' □ RL1' $\rightsquigarrow$ $<<x2>, <<x3,L6>>, \lambda Q. \lambda z_1.$ L6: $\exists x_3 \in$ FEMALE$_0$: Q(x_3) $\wedge$ $z_1 = x_3$>
R4' □ RL2' $\rightsquigarrow$ $<<x1>, <<x4,L7>>, \lambda R. \lambda u_1.$ L7: $\exists x_4 \in$ MALE$_0$: R(x_4) $\wedge$ $u_1 = x_4$>

And the translation of *she saw him* is:

(RL1' ⊕ R3') @ ((RL2' ⊕ R4') □ (SR2' □ E2')) $\rightsquigarrow$ $<<x1,x2>, <<e2,L5>,<x4,L5>,<x3,L6>>, \lambda u_1.\lambda z_1.$ L6:$\exists x_3 \in$ FEMALE$_0$: L7: $\exists x_4 \in$ MALE$_0$: L5: $\exists e_2 \in$ SEE: Before(TA(e_2),T_0) $\wedge$ AGENT(x_3,e_2) $\wedge$ THEME(x_4,e_2) $\wedge$ $u_1 = x_4$ $\wedge$ $z_1 = x_3$>

To this result we insert-apply the translation of the TLINK tag to obtain the translation of *before she saw him*:

TL1' $\rightsquigarrow$ $<<e1,e2>,<>, \lambda y. \lambda z.$ Before(y,z)>

TL1' ⊕ ((RL1' ⊕ R3') @ ((RL2' ⊕ R4') □ (SR2' □ E2'))) $\rightsquigarrow$ $<<x1,x2,e1>, <<e2,L5>,<x4,L7>,<x3,L4>>,$ $\lambda u_1. \lambda z_1. \lambda y.$ L6: $\exists x_3 \in$ FEMALE$_0$: L7: $\exists x_4 \in$ MALE$_0$: L5: $\exists e_2$

$\in$ SEE: $\text{Before}(\text{TA}(e_2),T_0) \wedge \text{AGENT}(x_3,e_2) \wedge \text{THEME}(x_4,e_2) \wedge u_1 = x_4 \wedge z_1 = x_3 \wedge \text{Before}(y,e_2)>$

This is combined with the representation of *Igor called Amy* using merge-application, giving the desired end result:

(10) **(R1' $\square$ (R2' $\square$ (SR1' $\square$ E1'))) $\odot$ (TL1' $\oplus$ ((RL1' $\oplus$ R3') @ ((RL2' $\oplus$ R4') $\square$ (SR2' $\square$ E2')))))** $\rightsquigarrow <<>,$
$<<e_2,L5>,<x_4,L7>,<x_3,L6>,<x_1,L3>,<x_2,L4>,$
$<e_1,L1>>$, L3: $\exists x_1 \in \text{IGOR}_0$: $\text{ANIMATE}(x_1) \wedge$ L4: $\exists x_2 \in \text{AMY}_0$: $\text{ANIMATE}(x_2) \wedge$ L1: $\exists e_1 \in \text{CALL}$: $\text{Before}(\text{TA}(e_1),T_0) \wedge$ L2: $\exists t_1$: $\text{NOON}(t_1 \wedge \text{AGENT}(x_1,e_1) \wedge \text{THEME}(x_2,e_1) \wedge$ L6: $\exists x_3 \in \text{FEMALE}_0$: L7: $\exists x_4 \in \text{MALE}_0$: L5: $\exists e_2 \in$ SEE: $\text{Before}(\text{TA}(e_2),T_0) \wedge \text{AGENT}(x_3,e_2) \wedge \text{THEME}(x_4,e_2) \wedge x_1 = x_4 \wedge x_2 = x_3 \wedge \text{Before}(e_1,e_2)>$

6. Conclusions and perspectives

In this paper we indicated how various types of semantic information can be annotated in an integrated fashion in an annotation language with a well-defined semantics.

In fact, the integrated *interpretation* of the various kinds of semantic information is of more interest than the integrated *annotation*. One may wish to annotate various types of semantic information in separate layers, using a layered approach to annotation. But in both cases the same issues arise concerning (a) deciding which information belongs to which type (to which layer); and (b) how does the meaning of the annotations of one type (in one semantic layer) consistently combine with that of other types (in other semantic layers)? In this respect, we have come across several phenomena where existing semantic annotation schemes are incompatible.

We noted for instance that the relations between events encoded in ISO-TimeML by means of SLINK and ALINK are predominantly of a thematic rather than a temporal nature, and does belong to the area/layer of semantic role relations. On the other hand, the relations between an event and its place and time, which are usually annotated by means of semantic roles, do not really belong there since they do not constitute relations between an eventuality and a 'participant' in that eventuality; they rather belong to the areas/layers of temporal and spatial information. Also, referential relations may have dependencies with aspectual relations between the verbs denoting the eventualities in which the referents participate, e.g. in the case of control verbs (*Amy started to [Amy] laugh.*). These are just a few examples of how the semantic study of the concepts used in semantic annotation may lead to a better conceptual understanding and to improving the annotation schemas for the various types of information.

It is also worth pointing out that, except for the interferences between concepts in different areas/layers of the kind just mentioned, the approach to semantic annotation interpretation described in this paper is independent of the particular choices of annotation tags; for instance, the set of semantic roles that we considered, taken from the LIRICS project (see Petukhova & Bunt, 2008) could be replaced by another set of roles without any change in the procedure for translating the annotations into formal logic.

Finally, it is clear that the approach that we have outlined can be extended to other areas of semantic annotation, such as spatial information, noun-noun relations, quantification, and discourse relations. It thus offers the exciting perspective of increasingly incorporating semantic information in a robust and incremental fashion in NLP applications, as the technology of semantic annotation continues to develop.

7. References

H. Bunt and C. Overbeeke. 2008. An Extensible Compositional Semantics for Temporal Annotation. In *Submitted to LREC 2008 Workshop LAW-II: Linguistic Annotation Workshop*, Marrakech, Morocco. Paris: ELRA.

H. Bunt and L. Romary. 2002. Towards Multimodal Content Representation. In K. S. Choi, editor, *Proceedings of LREC 2002 Workshop on International Standards of Terminology and Language Resources Management*, pages 54–60, Las Palmas, Spain. Paris: ELRA.

H. Bunt and A. Schiffrin. 2007. *Documented Compilation of Semantic Data Categories*. Nancy: LORIA (http://lirics.loria.fr). LIRICS Deliverable D4.3.

H. Bunt. 2007. The Semantics of Semantic Annotation. In *Proceedings of the 21st Pacific Asia Conference on Language, Information and Computation (PACLIC21)*, pages 13–28.

ISO. 2007. *Language Resource Management - Semantic Annotation Framework (SemAF) - Part 1: Time and Events*. Secretariat KATS, August. ISO Report ISO/TC37/SC4 N269 version 19 (ISO/WD 24617-1).

ISO. 2008. *Language Resource Management - Semantic Annotation Framework (SemAF) - Part 2: Dialogue acts*. Secretariat KATS, February. ISO Report ISO/TC37/SC4 N442 version 01 (ISO/WD 24617-2).

G. Katz. 2007. Towards a Denotatial Semantics for TimeML. In F. Schilder, G. Katz, and J. Pustejovsky, editors, *Annotation, Extraction, and Reasoning about Time and Events*. Springer, Dordrecht.

K. Lee. 2008. Formal Semantics for Interpreting Temporal Annotation. Invited talk to be presented at 18th International Congress of Linguists. Seoul, July 2008.

V. Petukhova and H. Bunt. 2008. LIRICS semantic role annotation: Design and evaluation of a set of data categories. In *Proceedings 6th International Conference on Language Resources and Evaluation (LREC 2008)*, Marrakech, Morocco. Paris: ELRA.

I. Pratt-Hartmann. 2007. From TimeML to Interval Temporal Logic. In *Proc. 7th International Workshop on Computational Semantics (IWCS-7)*, pages 166–180, Tilburg.

J. Pustejovsky, J. Castano, R. Ingria, R. Gaizauskas, G. Katz, R. Saurí, and A. Setzer. 2003. TimeML: Robust Specification of Event and Temporal Expressions in Text. In *Proc. 5th International Workshop on Computational Semantics (IWCS-5)*, pages 337–353, Tilburg.

J. Pustejovsky, R. Knippen, J. Littman, and R. Saurí. 2007. Temporal and Event Information in Natural Language Text. In H. Bunt and R. Muskens, editors, *Computing Meaning, vol. 3*, pages 301–346. Springer, Dordrecht.