Annotating temporal and event quantification

Harry Bunt and James Pustejovsky
Tilburg Center for Creative Computing/ Department of Computer Sciences
Tilburg University, The Netherlands / Brandeis University
harry.bunt@uvt.nl, jamesp@cs.brandeis.edu

Abstract

In this paper we propose someimprovements to the proposed ISO-TimeML standard for the semantic annotation of information about time and events. We argue that these improvements are called for, either in order to deal with suboptimal choices in the XML-based representation of annotation structures, or for resolving some of the difficulties that arise due to the impossibility to separate semantic phenomena relating to time and events, like temporal quantification, from their more general form. We indicate solutions for both types of cases.

1 Introduction

The definition of annotation languages has in recent years become a focal area of interest in the International Organization for Standardization ISO. Expert groups have been formed with the aim to develop standards for the representation and annotation of language resources, such as the ISO-TEI standard for feature structure representations, the Lexical Markup Framework, standards for annotating documents with morphosyntactic, syntactic, and semantic information, and the Linguistic Annotation Framework, a meta-standard for these efforts.

In the area of semantic annotation, the project *Semantic Annotation Framework* was started, with several parts for dealing with different kinds of semantic information. Part 1 deals with the annotation of information related to time and events, and has proposed a standard (ISO 2009) which is based on TimeML (Pustejovsky et al., 2003), and is therefore called 'ISO-TimeML'. The ISO-TimeML standard constitutes a significant step forward in the development of semantic annotation languages, in particular in comparison to its predecessor TimeML.

Yet, ISO-TimeML also has certain shortcomings, some having to do with with suboptimal representational choices, some with underlying conceptual choices that lack a solid foundation – in both cases mostly the result of taking over elements of TimeML. In this paper we will outline some of these deficiencies, and indicate how they may be resolved.

This paper is organized as follows. In section 2 we very briefly characterize ISO-TimeML and its relation to TimeML. In section 3 we describe a number of deficiencies of ISO-TimeML as an annotation representation language, and in section 4 we discuss some underlying conceptual problems, mostly having to do with the analysis of descriptions of recurring events and temporal and inter-event quantification. We end with concluding remarks in section 5.

2 Information about Time and Events

2.1 General characteristics

ISO-TimeML is meant to allow the assignment of semantic mark-ups to expressions which: (1) denote an event, state, or a process, such as finite verb forms and certain nouns (such as *accident* and *concert*; (2) relate events to aspectually related or subordinate events, like *started to laugh; wanted to cry*; (3) describe dates, times, particular periods, such as *the twenty-first century; last week*; (4) indicate relations between temporal entities, such as *at, before,* *during, for*;(5) denote an extent of time, such as *two seconds*; (6) describe a frequency of occurrence of events for a given extent of time, such as *four times last week; twice a day*; (7) anchor events in time.

ISO-TimeML is an adaptation of the TimeML language, developed by James Pustejovsky and associates (e.g. Pustejovsky et al., 2005). Apart from some technical improvements, ISO-TimeML has two fundamental properties that set it apart: (1) compliant with the requirement of *semantic adequacy* (Bunt & Romary, 2002) it has a formal semantics; and (2) compliant with the Linguistic Annotation Framework (LAF, Ide & Romary, 2004) it distinguishes between annotations and their representation, and in supporting *stand-off* rather than inline annotation, i.e. annotations are represented in separate files, separate from the document containing the primary language data.

A prototypical example of an annotation representation in ISO-TimeML is the following, for the sentence *John left on 31 December 2007*:

(1)
```
<isoTimeML xmlns:
"http://www.iso.org/isoTimeML
xml:id="a1">
<EVENT xml:id="e1" target=
"#token2" pred="LEAVE"
type= "TRANSITION" class=
"OCCURRENCE" tense="PAST"
aspect="NONE" pos="VERB"
vform="NONE" mood="NONE"
polarity="POS"/>
<SIGNAL xml:id="s2"
target="#token3"/>
<TIMEX3 xml:id="t1"
targets="#token4 #token5
#token6 type="DATE"
value="2007-12-31"/>
<TLINK eventID="#e1"
relatedToTime="#t1"
signalID="#s1"
relType="IS_INCLUDED"/>
</isoTimeML>
```

ISO-TimeML follows standard ISO 24610, jointly developed with the Text Encoding Initiative (see Burnard and Bauman, 2007) for how to anchor annotation representations to primary text using the `target` and `targets` atrtributes, which point to source text tokens. To save space, in the examples to follow we will suppress the enclosing `"isoTmeML"` tags, abbreviate ”xml:id” by ”id”, leave out the `target`(s) attributes and values, and leave out attributes that have a `"NONE"` value or another default value. We will also leave out the attributes `class` and `type`, which refer to certain event classifications, assuming the values of the `pred` attribute to be elements in an ontology or other semantic resource that contains such classifications, making their annotation redundant.

2.2 Abstract syntax

The abstract syntax of ISO-TimeML defines the set-theoretical structures which constitute the information about time and events that may be contained in annotations. The abstract syntax definition consists of two parts: (a) a specification of the elements from which these structures are built up, called a ‘conceptual inventory’; and (b) a set of rules which describe the possible combinations of these elements into annotation structures. What these combinations mean, i.e. which information is captured by an annotation structure, is specified by the semantics associated with the abstract syntax.

a. Conceptual inventory

The concepts which can be used to build ISO-TimeML annotations fall into five categories, all formed by finite sets of temporal and event-related entities and relations, plus the concepts of real and natural numbers. The categories of temporal and event-related entities and relations are the following:

- finite sets of elements called ‘event types’; ‘tenses’, ‘aspects’, ‘polarities’, and ‘signatures’;
- finite sets of elements called ‘temporal relations’; ‘duration relations’; ‘numerical relations’: ‘event subordination relations’, and ‘aspectual relations’:
- a finite set of elements called ‘time zones’:
- finite sets of elements called ‘calendar years’; ‘calendar months’; ‘calendar weeks’; ‘calendar day numbers’; (with 31 elements); ‘week days’; and ‘clock times’:
- a finite set of elements called ‘temporal units’.

b. Annotation construction rules

Annotation structures in ISO-TimeML consist of *entity structures* and *link structures*. Entity structures contain semantic information about a segment of source text; link structures describe semantic relations between segments of source text.

An entity structure is a pair <s, a> consisting of a stretch of source text s and an annotation a. A link structure is a triple $<e_1, e_2, r>$ consisting of two entity structures and a relational element. An ISO-TimeML annotation structure is a pair $<E, L>$, where E is a nonempty set of entity structures and L is a set of link structures.

Entity structures:

Entity structures $<s, a>$ come in six varieties, depending on the a component.

1. An *event structure* is an 6-tuple $<e, t, a, \sigma, k, v>$ where e is a member of the set of event types; t and a are a tense and an aspect, respectively; σ is a set-theoretical type, such as $individual\ object$ or $set\ of\ individual\ objects$; k is a natural number or a numerical predicate (like *more than five*; and v is a veracity (including claimed truth or falsity, corresponding to positive or negative polarity in natural language).
2. An *instant structure* is either a triple $<$ $time\ zone$, $date$, $clocktime$$>$, or a triple $<$$time$-$amount\ structure$, $instant\ structure$, $temporal\ relation$$>$ (*"half an hour before midnight"*).
3. A *date structure* is a triple consisting of a calendar year, a calendar month, and a calendar day number;
4. The following set-theoretical structures are *interval structures*:
   (a) a pair $<t_1, t_2>$ of two instant structures, corresponding to the beginning and end points of the interval;
   (b) a triple $<$$time$-$amount\ structure$, $interval\ structure$, $temporal\ relation$$>$ (*"a week before Christmas"*);
   (c) a triple $<t_1, t_2, R>$ where t_1 and t_2 are either instant structures or interval structures, and where R is a duration relation (*"from nine to five"*).
5. A *time-amount structure* is a pair $<n, u>$ or a triple $<R, n, u>$, where n is a real number, R a numerical relation, and u a temporal unit.

Link structures:

There are seven types of link structures in ISO-TimeML: (1) for anchoring events in time; (2) for temporally relating one event to another; (3) for relating intervals and instants to each other; (4) for measuring the duration of an event; (5) for measuring the length of a temporal interval (6) for subordination relations between events; and (7) for aspectual relations between events.

1. A temporal anchoring structure is a triple $<$$event\ structure$, $interval\ structure$, $temporal\ anchoring\ relation$$>$, or a triple $<$$event\ structure$, $instant\ structure$, $temporal\ anchoring\ relation$$>$:
2. An event-temporal relation structure is a triple $<$$event\ structure$, $event\ structure$, $temporal\ relation$$>$;
3. An intra-temporal relation is a triple $<$$interval$ or $instant\ structure$, $interval$ or $instant\ structure$, $temporal\ relation$$>$
4. An event-duration structure is a triple $<$$event\ structure$, $time$-$amount\ structure$, $duration\ relation$$>$:
5. An interval measurement structure is a pair $<$$interval\ structure$, $time$-$amount\ structure$$>$:
6. A subordination structure is a triple $<$$event\ structure$, $event\ structure$, $subordination\ relation$$>$:
7. An aspectual structure is a triple $<$$event\ structure$, $event\ structure$, $aspectual\ relation$$>$.

2.3 Concrete syntax

ISO-TimeML also comes with a (partial) semantics and with a concrete syntax. The concrete syntax is a specification of how the information in annotation structures may be represented in XML, as illustrated in example (1). This concrete syntax is very similar to that of the original TimeML language. The discussion of the semantics is beyond the scope of this paper, but see Bunt (*forthc.*).

3 Representation issues

While ISO-TimeML, like the original TimeML, has a very broad syntactic coverage of expressions relating to time and events. from a semantic point of view ISO-TimeML still has certain shortcomings, mainly due to the following causes:

- TimeML, the main source of inspiration for ISO-TimeML, did not have a semantics. When developing ISO-TimeML, two alternative approaches were followed for defining a semantics: on the one hand, a semantics was devised for a rather limited part of the concrete syntax of the language, based on Interval Temporal Logic (Pratt-Hartman, 2007); on the other hand an event-based semantics was specified for the abstract syntax (Bunt & Overbeeke, 2008). However, the development of these semantics has (yet) not been fed back systematically into the specification of the concrete syntax, which has by and large remained the same as that of TimeML.

- While ISO-TimeML focuses on information related to time and events, from a semantic point of view it is not really possible to separate such information from general semantic phenomena such as quantification and modality. Where no adequate way to annotate such phenomena has been developed, it would not be realistic to expect ISO-TimeML to provide this.

In this section we discuss some phenomena which ISO-TimeML does not handle in a satisfactory manner, and indicate possible solutions. The proposed solutions rest on: (1) improving the concrete syntax in order to be a more accurate rendering of the abstract syntax[1] and (2) improving the conceptual view on semantic issues, as reflected in the abstract syntax.

3.1 Measuring amounts of time

ISO-TImeML so far does not have a satisfactory way to represent amounts of time. The sentence *John taught for two hours on Tuesday* is for instance marked up as follows (leaving out attributes and values that are of little relevance to the present discussion):

(2)
```
<EVENT id="e1" pred="TEACH"
tense="PAST"/>
<SGNAL id="s1" pred="FOR"/>
<TIMEX3 id="t1" type="MEASURE"
value="M2H"/>
<TLINK eventID="#e1"
relatedToTime="#t1"
relType="SIMULTANEOUS"/>
<SIGNAL id="s1"/>
<TIMEX3 id="t2" pred="MONDAY"
type="DATE"
value="xxxx.wxx.2"/>
<TLINK eventID="#e1"
relatedToTime="#t2"
relType="IS_INCLUDED"/>
```

Both the representation of *two hours* and of the relation between the teaching event and its duration are not satsfactory. In its representation of the amount of time *two hours*, the representation does not accurately reflect the conceptual view of amounts of time which is expressed in the abstract syntax, where a *time-amount structure* is defined as consisting of a numerical specification and the specification of a unit of measurement. In the concrete syntax, these two components are not present as such; instead, an alphanumerical string is used. Semantically, this is clearly not optimal. The relation between the event and its duration is expressed by 'SIMULTANEOUS', but this seems conceptually wrong: simultaneity is a relation between events. We therefore propose the following changes to ISO-TimeML.

1. a new element MLINK is introduced for representing the relation between events and their durations, with attributes pointing to the representations of an event and an amount of time, respectively;

2. a new element TIME_AMOUNT is introduced, with numerically valued attribute and an attribute for specifying a unit of measurement.

With these changes, the sentence *John taught for two hours on Tuesday* can be represented as follows:

[1] In Bunt, 2009) a more radical approach is suggested, which leads to a more comprehensive overhaul of the ISO-TimeML representation format.

(3)
```
<EVENT id="e1" pred="TEACH"
tense="PAST"/>
<SGNAL id="s1" pred="FOR"/>
<TIME_AMOUNT id="a1" aNum="2"
unit="HOUR"/>
<MLINK eventID="#e1"
timeAmountID="#a1"
signalID="#s1"/>
```

3.2 Recurring events

Besides references to a single event, as in example (1), in natural language we also encounter lots of cases where reference is made to multiple or recurring events, as in *John called twice.* In the TimeML-based representation format of ISO-TimeML, the annotation of this sentence is represented as follows:

(4)
```
<EVENT id="e1" tense="PAST"/>
<TIMEX3 id="t1" freq="2X"/>
<TLINK eventID="#e1"
relatedToTime="#t1"
relType="DURING"/>
```

This representation is unsatisfactory in several respects. First, the `EVENT` part refers to an event `e1`, temporally linked to a temporal 'entity' "twice", while the source text refers to two events. Second, what kind of entity is 'twice'? ISO-TimeML uses the `TIMEX3` tag for all temporal entities, and distinguishes these entities by means of the `type` attribute into *dates, times, periods*, and *sets*. Clearly, 'twice' does not fit any of these categories. In fact, "twice" should not be considered as a temporal entity at all; it is rather a kind of counter, expressing how many times a certain type of event occurred; it doesn't provide any *temporal* information. Third, and closely related to the previous point, what is the relation between the events and 'twice'? The ISO-TimeML representation uses the 'DURING" relation for this purpose, but that is a relation in time, it seems clear that the relation between events and the number of them is not of that nature, but is more like the cardinality of a set. So far, ISO-TimeML does not have any type of relation that might be appropriate here.

The abstract syntax reflects a conceptual view that differs from the one underlying this representation. First, the various types of temporal objects distinguished at the conceptual level do not include anything like 'twice'; instead, the interpretation of *twice* as a counter is captured by the k (for 'cardinality') element in an event structure. Second, as mentioned the ISO-TimeML representation uses `TIMEX3` elements for all entity structures that somehow relate to time, distinguished by the value of the `type` attribute, forcing an annotator to represent a reference to a set of events by `<TIMEX3 ... type="SET".../>`. This is rather unfortunate, since the issue of whether a source text expression refers to a set of entities rather than to a single entity is independent of the type of entity (event, temporal interval, amount of time,...). This representational choice does not accurately reflect the conceptual distinctions made in the abstract syntax, where the 'signature' element (σ) captures such distinctions as between sets and individuals, while a distinction like that between a temporal interval and its length is made by using different entity structures.

We see here that ISO-TimeML representations do not deal with certain phenomena in a satisfactory way, due to an imperfect match between distinctions made in the abstract syntax and those expressed in the concrete representation format. We therefore propose certain changes to the representation format.

1. an attribute `signature` is introduced for the `EVENT` element, which can have the values `'individual'` and `'set'`;

2. the attribute `freq` is replaced by `card` (for 'cardinality'), which has numerical values.

With these changes, the sentence *John called twice* can be represented as follows:

(5)
```
<EVENTid="e1" type="CALL"
tense="PAST" signature="SET"
cardinality="2"/>
```

This representation says, rather laconically, that two *call* events occurred. Because the `signature` attribute has the value `"SET"`, the annotation does not refer to a single event but to a set of events.

A sentence with a genuine frequency description, such as *John calls home twice a day*, in fact describes a quantified relation between a set of recurring events and the set of periods in which they occur – see the next section.

In ISO-TimeML, such a sentence is represented as shown in (6), leaving out atributes and values of no particular interest here.

(6)
```
<EVENTid="e1" pred="CALL"
tense="NONE"/>
<TIMEX3id="t1" freq="2X"/>
<TIMEX3id="t2" type="SET"
value="P1D" quant="EVERY">
<¯TLINK eventID="#e1"
relatedToTime="#t2" />
relType="DURING"/>
```

The criticism that we leveled against the representation (5) also applies in this case; moreover, the temporal quantification is not represented in an adequate fashion. Problematic is that the `EVENT` element does not correspond to a set of events; there is no way in ISO-TimeML to represent a set of events, and that the set of periods involved in the quantification is characterized as `"P1D"`, where 'P' stands for 'period' and 'D' for day; this is the ISO-TimeML way of describing a one-day period. This is from a semantic point of view rather primitive, the more since the abstract syntax supports the articulate characterization of any length of time by means of time-amount structures.

3.2.1 Event Quantification

Quantification in natural language is the phenomenon that a predicate is applied to each or some of the members of a set, or collectively to the set as a whole, or to certain subsets of it. This may be a unary predicate, as in (7a), a binary one, as in (7b-c), or one of higher arity, as in (7d), relating the members of a set to one ore more other sets.

(7) a. These books are heavy.
b. The students have to read five papers.
c. The men moved the pianos.
d. The boys gave the girls some of the sweets.

In (7a), the predicate *heavy* can be understood as applying to the individual members of a certain set of books, or collectively to the set of books as a whole. This is called the *distributivity* of the quantification (Bunt, 1985).

When verbs are viewed as referring to events, as in ISO-TimeML, then *every* sentence expresses a quantification over events. Consider for instance the following example:

(8) Everybody will die.

This sentence can be read either as expressing that for each person there will be an event where this person dies, or that there will be an (apocalyptic) event in which everyone will die ("collectively"). So even an intransitive verb gives rise to quantification, and thus to issues such as collective versus individual involvement and relative scoping.

ISO-TimeML does not consider the relations between events and their participants, therefore issues of relative scoping and collectiveness might seem not to arise. However, in principle *any* relation between two sets of entities is quantified, and so are the relations between events and temporal entities, for instance by means of temporal quantifiers such as *always, sometimes, every Monday*, so we do need some provisions in ISO-TimeML for time- and event-related quantification. ISO-TimeML has the attribute `quant` for this purpose, as one of the attributes of temporal entities.

This is not a satisfactory solution; quantifications, or rather, the properties of a quantification such as the distributivities of the sets of participants involved, are aspects of *relations*, such as the temporal anchoring relation between a set of events and a set of intervals. We therefore propose to introduce a couple of attributes in the representation of link structures, allowing us to annotate quantificational properties. For instance, a `TLINK` element will have attributes `eventDistr` and `timeDistr`, with values like `INDIVIDUAL` and `COLLECTIVE` for indicating the distributivity on either side of the relation. This is illustrated in the ICS representation (9b) of the quantification in *John calls every day*, with its representation in ISO-TimeML in (9c) for contrast.

(9) a. John calls every day.

b.
```
<EVENT id="e1"
pred="CALL" tense="NONE"
signature="SET"/>
<TIMEX3id="t1" type="DAY"
signature="SET"/>
<TLINK id="a1" eventID="#e1"
relatedToTime="#t1"
```

```
relType="INCLUDED_IN"
eventDistr="INDIV"
timeDistr="INDIV"
timeQuant="EVERY"/>
```

c.
```
<EVENT id="e1" tense="NONE"
<TIMEX3id="t1" type="SET"/>
<TLINK id="a1" eventID="#e1"
relatedToTime="#t1"
relType="INCLUDED_IN"
quant="EVERY"/>
```

We noted above that a description of recurring events with a certain frequency, as in *John calls twice every day*, in fact forms a case of event quantification. Using the representation of amounts of time proposed in the previous section, in combination with the representation of quantified relations proposed here, we obtain the following representation of this sentence.

(10)
```
<EVENT id="e1" pred="CALL"
signature="SET"/>
<TIMEX3 id="t1"
<TLINK eventID="#e1"
relatedToTime="#t1"
relType="INCLDUED_IN"
eventDistr="INDIVIDUAL"
timeDistr="INDIVIDUAL"
eventQuant="2"
timeQuant=EVERY"/>
```

This representation can be read as saying that a set of *call* events is temporally anchored in a set of days, such that individual events are anchored at individual days, where every day includes a temporal anchor for two of these events. This is exactly what we want.

Note that many if not all of the complexities associated with quantification in natural language also turn up in relation with events. For example, the phenomenon of collective quantification, as illustrated in example (7c), occurs for events in the sentence *Peter tries to run five times around the block every morning.* Here the relation between the daily *try* events and the *run* events is collective on the latter side, since Peter does not try to run around the block once or twice, but the object of the trying is the collection of five *run* events.

Similarly, *cumulative quantification* (Scha, 1981) occurs in the way in which the duration of a set of *teach* events is quantified in (11), which of course is not intended to mean that each of John's classes lasts 12 hours.

(11) a. John taught 12 hours last week.

b.
```
<EVENTid="e1" type="TEACH"
signature="SET"/>
<TIME_AMOUNT id="l1"
aNum="12" unit="HOUR"/>
<TIMEX3id="t1" type="WEEK"
signature="INDIVIDUAL"/>
<TLINK id="c1" eventID="#e1"
relatedToTime="#t1"
relType="INCLUDED_IN"
eventDistr="INDIVIDUAL"/>
<DURATION event="#e1"
timeExtent="#t1"
eventDistr="CUMULATIVE"/>
```

This representation says that there is a set of *teach* events, each of which occurred some time `INCLUDED_IN` last week, which have a total (`eventDistr="CUMULATIVE"`) duration of 12 hours.

4 Discussion and Conclusions

In this paper we have shown that the proposed ISO-TimeML standard is in need of some improvements, mostly because of the imperfect representation of the annotation structures, defined by the abstract syntax. An example is the representation of the numerical information in the description of a repeated event, as in *John called twice.* In the abstract syntax, the annotation structure for the repeated *call* event has a signature and a cardinality, which makes it possible to express in the concrete representation that there is a set of two *call* events. The ISO-TimeML representation, by contrast, links a *call* event to a rather mysterious entity called `"2X"` as the value of an attribute `'freq'`.

The problems that we have discussed with the representation of quantification in relation to time and events partly have a deeper origin than mismatches between abstract and concrete syntax. They relate to the fact that the annotation of quantification in this domain cannot be separated from the analysis of quantification more generally, which is a vast area of research in formal and computational semantics. The representational approach that we have proposed here, using a number of attributes in the elements representing link structures that capture aspects of quantification, is inspired by a study of the underspecified semantic representation of quantification and modification in terms of feature structures (Bunt, 2005). This approach may open the way to developing a standard for the annotation of quantification and modification in natural language more generally.

Acknowledgements

We have enjoyed many in-depth discussions of issues in the temporal and event-related semantics of natural language with the members of the TimeML team at Brandeis University, and with our colleagues in the ISO-TimeML project, in particular with Kiyong Lee.

References

Bunt, H. (1985). *Mass Terms and Model-Theoretic Semantics*. Cambridge University Press.

Bunt, H. (2005). Quantification and Modification Represented as Feature Structures. In *Proceedings 6th International Workshop on Computational Semantics (IWCS-6)*, pages 54–65, Tilburg, Netherlands.

Bunt, H. (2010). A methodology for designing semantic annotation languages exploring semantic-syntactic ISO-morphisms. In *Proceedings of the Second International Conference on Global Interoperability for Language Resources (ICGL 2010)*, Hong Kong.

Bunt, H. and Overbeeke, C. (2008). An extensible, compositional semantics of temporal annotation. In *Proceedings of LAW-II: the Second Linguistic Annotation Workshop*, Marrakech.

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

Burnard, L. and Bauman, S. (2007). *TEI P5: Guidelines for Electronic Text Encoding and Interchange*. TEI Consortium.

Ide, N. and Romary, L. (2004). International Standard for a Linguistic Annotation Framework. *Natural Language Engineering*, 10:211–225.

ISO (2009). *Language Resource Management - Semantic Annotation Framework (SemAF) - Part 1: Time and Events*. Secretariat KATS. ISO International Standard 24617-1:2009(E)), 11 October 2009.

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

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

Scha, R. (1981). Distributive, Collective and Cumulative Quantification. In Groenendijk, J. and Stokhof, M., editors, *Formal Methods in the Study of Language*. Mathematical Center, Amsterdam.