A Semantically Compositional Annotation Scheme for Time Normalization

Steven Bethard, Jonathan L Parker
University of Alabama at Birmingham
Birmingham, AL, USA
bethard@uab.edu, aescgar@uab.edu

Abstract

We present a new annotation scheme for normalizing time expressions, such as *three days ago*, to computer-readable forms, such as `2016-03-07`. The annotation scheme addresses several weaknesses of the existing TimeML standard, allowing the representation of time expressions that align to more than one calendar unit (e.g., *the past three summers*), that are defined relative to events (e.g., *three weeks postoperative*), and that are unions or intersections of smaller time expressions (e.g., *Tuesdays and Thursdays*). It achieves this by modeling time expression interpretation as the semantic composition of temporal operators like UNION, NEXT, and AFTER. We have applied the annotation scheme to 34 documents so far, producing 1104 annotations, and achieving inter-annotator agreement of 0.821.

Keywords: time expressions, normalization, compositionality

1. Introduction

Time normalization is the task of translating natural language expressions of time, such as *three days ago*, to computer-readable forms, such as `2015-10-12`. Accurate time normalization is critical for enabling temporally-constrained search over free text language resources. Applications range from the study of literary texts (Fischer and Strötgen, 2015) to mining patient records for potential causes of disease (Lin et al., 2014).

The most popular scheme for annotating such normalized forms is ISO-TimeML (Pustejovsky et al., 2010), an extension of the TIDES annotation guidelines (Ferro et al., 2005). Figure 1 shows a sample of such annotations. Time expressions are annotated as phrases and the VALUE attribute indicates the normalized form. This type of annotation has formed the basis for a wide variety of annotated corpora, including the TimeBank (Pustejovsky et al., 2003), WikiWars (Mazur and Dale, 2010), TimeN (Llorens et al., 2012), and the various TempEval shared tasks (Verhagen et al., 2007; Verhagen et al., 2010; UzZaman et al., 2013).

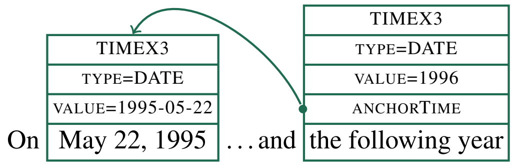

Figure 1: ISO-TimeML annotation of times in a sentence from the TimeBank article APW19980418.0210

However, there are a few drawbacks of the ISO-TimeML approach. First, it struggles to represent times that do not align to a single calendar unit (day, week, month, etc.), such as *the past three summers*, since this cannot be described with some prefix of a YYYY-MM-DDTHH:MM:SS date-time. Second, it does not permit times to be defined relative to events (only relative to times), as in *three weeks postoperative*, where the three-week period is defined relative to some operation event. Finally, the flat nature of the annotations does not reflect the compositional semantics of time expressions. For example, *following* can be applied not only to *year* as in Figure 1, but also to *week*, *Tuesday*, or *December*, and the semantics is identical: find the first such calendar unit in the future relative to the anchor time (*May 22, 1995*).

We thus propose a new annotation scheme for time normalization that can faithfully represent a wider variety of time expressions, that annotates fine-grained components of time expressions, and that formally defines the semantics of each annotation in terms of mathematical operations over intervals on the timeline.

2. Definitions

Each annotation type in our schema corresponds to a formal interpretation of one of the following temporal concepts.

timeline, T: The infinite sequence of time points to which time expressions should be anchored. We will assume that each time point marks the start of a second. For example, 2015-08-03 09:35:47 and 1990-01-01 00:00:00 are both time points. For simplicity, when a time point ends in 00:00:00, we will abbreviate that time point by omitting the 00:00:00.

period, $\Delta : U \rightarrow \mathbb{N}$: An amount of time, expressed as counts of standard time units $U = \{$YEARS, MONTHS, etc.$\}$. For example, *a week* (i.e., WEEKS $\rightarrow 1$) or *three months* (i.e., MONTHS $\rightarrow 3$). Note that periods are independent of the timeline. For example, given only the period expression *10 weeks*, it is impossible to assign time points of the form NNNN-NN-NN NN:NN:NN to its start and end.

interval, $i \in I = \{[t_i, t_j) : t_i \in T \wedge t_j \in T \wedge t_i < t_j\}$: An interval on the timeline, defined by a starting point (inclusive) and an ending point (exclusive). For example, the expression *1990* corresponds to the interval [1990-01-01, 1991-01-01). Figure 2 shows a graphical depiction of *1990* as a timeline interval.

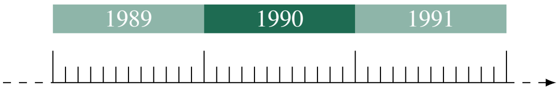

Figure 2: The interval *1990* on the timeline.

repeating interval, $R \subset I$**:** A sequence of intervals on the timeline. For example, *January*, *08:00 am*, or *Friday the 13th*. We view such expressions as the infinite sequence of all possible intervals they could refer to. For example, *January* could refer to [2000-01-01, 2000-02-01), to [2001-01-01, 2001-02-01), to [2002-01-01, 2002-02-01), etc. Figure 3 shows a graphical depiction of *January* as a repeating interval on the timeline.

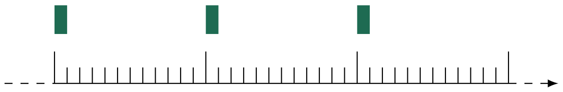

Figure 3: The repeating interval *January* on the timeline.

temporal operator, $F : \ldots \to \ldots$**:** A function that takes periods, intervals and/or repeating-intervals as arguments and produces new periods, intervals or repeating-intervals. For example, the word *following* often evokes the NEXT operator ($I \times R \times \mathbb{N} \to \mathcal{P}(I)$), which takes an interval, a repeating interval, and a positive integer n, and produces the n sub-intervals of the repeating interval that follow the anchor interval.

3. Annotation Types

For each of time concepts above, we define annotation types for marking them in the text. The annotation types have properties and links to other annotations that encode the information necessary for their formal interpretation as above.

3.1. Periods

PERIOD annotations implement the *period* concept. The annotation spans a temporal unit (DAYS, WEEKS, etc.) and may link to a number. Examples are shown in Figure 4.

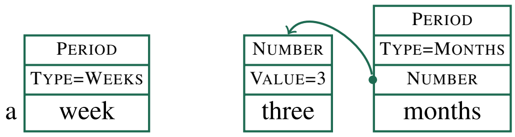

Figure 4: PERIOD annotations: *a week* and *three months*

3.2. Intervals

Only one annotation directly represents an interval on the timeline: YEAR. All other intervals are the result of temporal operators. A YEAR represents the interval from the first second of the year (inclusive) to the first second of the next year (exclusive). The YEAR annotation is also used to identify decades and centuries, by marking the missing digits with the '?' character. Examples are shown in Figure 5.

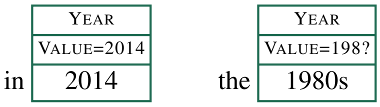

Figure 5: YEAR annotations: *in 2014* and *the 1980s*

3.3. Repeating Intervals

Though *repeating intervals* can represent any set of intervals on the timeline, they are most commonly used to identify intervals named by the calendar system: *Tuesday*, *January*, etc. The following annotations have been defined to follow the semantics of the ISO calendar system (ISO 8601, 2004):

MONTH-OF-YEAR e.g., *February*
WEEK-OF-YEAR e.g., *week 2 [of 1990]*
DAY-OF-MONTH e.g., *3rd [of March]*
DAY-OF-WEEK e.g., *Thursday*
HOUR-OF-DAY e.g., *08[:35:17]*
MINUTE-OF-HOUR e.g., *[08:]35[:17]*
SECOND-OF-MINUTE e.g., *[08:35:]17*

The schema also includes annotation types for generic calendar intervals (e.g., *every day*), and TimeML extensions of the ISO calendar system (e.g., PART-OF-DAY like *morning* or *evening*). Examples are shown in Figure 6.

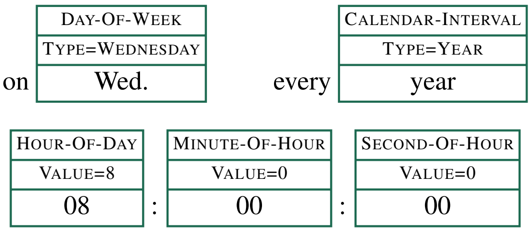

Figure 6: Repeating interval annotations: *on Wed.*, *every year*, and *08:00:00*

3.4. Temporal Operators

Temporal operators take the above periods, intervals, and repeating-intervals and semantically compose them to produce new times. Our annotation scheme contains 18 operators, each with a formal definition of their semantics.

Sum Two periods can be combined to produce a new period whose duration is the sum of the original two. Formally:

$$\text{SUM}(\Delta_1 : \text{PERIOD}, \Delta_2 : \text{PERIOD}) : \text{PERIOD} = \Delta_1 \uplus \Delta_2$$

where $\uplus$ is the multiset sum. Figure 7 shows an example SUM annotation which produces the formal interpretation $\text{YEARS} \to 2, \text{DAYS} \to 1$.

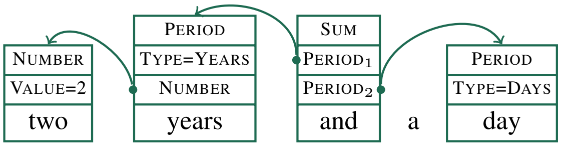

Figure 7: SUM annotation: *two years and a day*

Difference Two periods can also be combined by subtracting the amount of time in one from the other. Formally:

$$\text{SUM}(\Delta_1 : \text{PERIOD}, \Delta_2 : \text{PERIOD}) : \text{PERIOD} = \Delta_1 \setminus \Delta_2$$

where $\setminus$ is the multiset difference. Figure 8 shows an example DIFFERENCE annotation which produces the formal interpretation $\text{YEARS} \to 2, \text{DAYS} \to -1$.

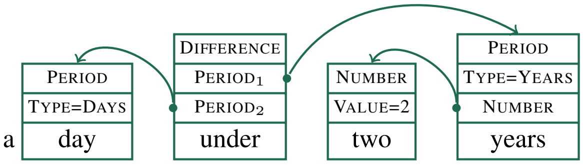

Figure 8: DIFFERENCE annotation: *a day under two years*

Union Two repeating intervals can be combined to produce a new repeating interval that is the union of the sub-intervals of the original two. Formally:

$$\textsc{Union}(R_1\colon \textsc{R-Interval}, R_2\colon \textsc{R-Interval}) = R_1 \cup R_2$$

Figure 9 shows an example UNION annotation along with a graphical depiction of its formal interpretation: the union of all calendar intervals named *Monday* and all calendar intervals named *Friday*.

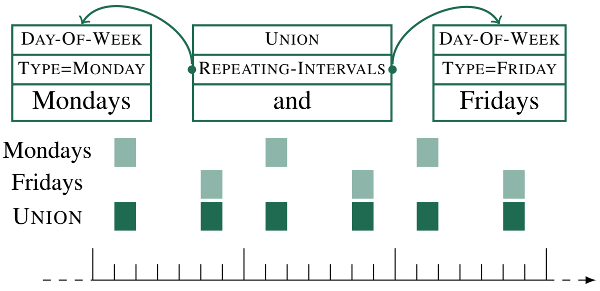

Figure 9: UNION annotation: *Mondays and Fridays*

Intersection Two repeating intervals can be combined to produce a new repeating interval that is the intersection of the sub-intervals of the original two. Formally:

$$\textsc{Intersection}(R_1\colon \textsc{R-Interval}, R_2\colon \textsc{R-Interval}) = R_1 \cap R_2$$

Figure 10 shows an example INTERSECTION annotation along with a graphical depiction of its formal interpretation: the intersection of all calendar intervals named *Saturday* with all calendar intervals named *March*.

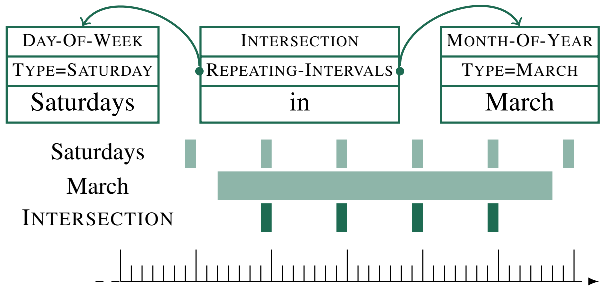

Figure 10: INTERSECTION annotation: *Saturdays in March*

Every-Nth A repeating interval can be sub-sampled by retaining only every n^{th} of the sub-intervals. Formally:

$$\textsc{Every-Nth}(R\colon \textsc{R-Interval}, n\colon \mathbb{N})\colon \textsc{R-Interval} = \{r_i \in R\colon i \bmod n = 0\}$$

Figure 11 shows an example EVERY-NTH annotation along with a graphical depiction of its formal interpretation: every odd-numbered calendar interval named *Friday*.

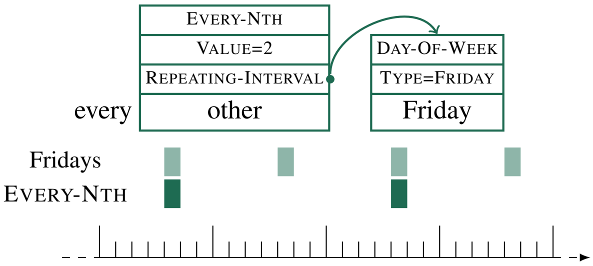

Figure 11: EVERY-NTH annotation: *every other Friday*

Last (Period) The first variant of the LAST operator takes an INTERVAL and a PERIOD, and creates an interval of the given length that ends just before the given interval. Formally:

$$\textsc{Last}([t_1, t_2)\colon \textsc{Interval}, \Delta\colon \textsc{Period})\colon \textsc{Interval} = [t_1 - \Delta, t_1)$$

Figure 12 shows an example LAST annotation along with a graphical depiction of its formal interpretation: the four days preceding the time at which the document was written. Note that the INTERVAL argument is DOC-TIME, a special time defined by the document metadata that identifies the time interval at which the document was written.

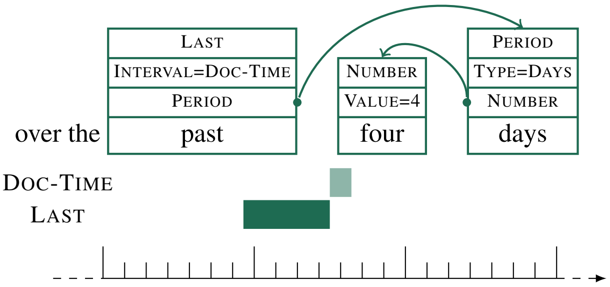

Figure 12: LAST (PERIOD) annotation: *over the past four days*

Last (Repeating Interval) The second variant of the LAST operator takes an INTERVAL, a REPEATING-INTERVAL, and an integer n, and finds the n latest repeated intervals that appear before the given interval. Formally:

$$\textsc{Last}([t_1, t_2)\colon \textsc{Interval}, R\colon \textsc{R-Interval}, n\colon \mathbb{N}) = n \text{ latest of } \{[t_s, t_e) \in R\colon t_e \leq t_1\}$$

Figure 13 shows an example LAST annotation along with a graphical depiction of its formal interpretation: the two *summer* intervals preceding the document creation interval.

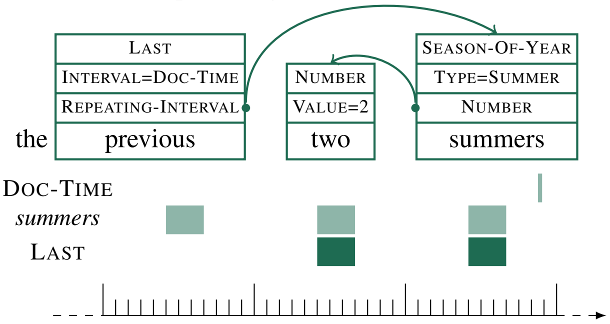

Figure 13: LAST (REPEATING-INTERVAL) annotation: *the previous two summers*

Next (Period) The first variant of the Next operator takes an Interval and a Period, and creates an interval of the given length that starts just after the given interval. Formally:

$$\textsc{Next}([t_1, t_2)\colon \textsc{Interval}, \Delta\colon \textsc{Period})\colon \textsc{Interval} \\ = [t_2, t_2 + \Delta)$$

Figure 14 shows an example Next annotation along with a graphical depiction of its formal interpretation: the week-long period following the document creation interval.

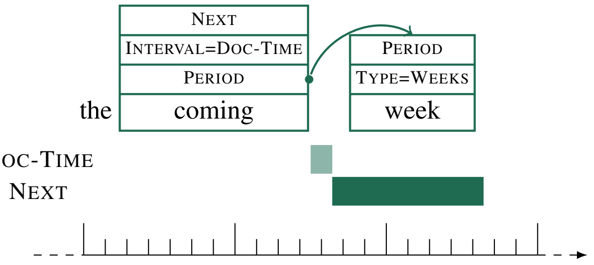

Figure 14: Next (Period) annotation: *the coming week*

Next (Repeating Interval) The second variant of the Next operator takes an Interval, a Repeating-Interval, and an integer n, and finds the n earliest repeated intervals that appear after the given interval. Formally:

$$\textsc{Next}([t_1, t_2)\colon \textsc{Interval}, R\colon \textsc{R-Interval}, n\colon \mathbb{N}) \\ = n \text{ earliest of } \{[t_s, t_e) \in R : t_2 \leq t_s\}$$

Figure 15 shows an example Next annotation along with a graphical depiction of its formal interpretation: the one calendar-week interval following the document creation interval. Note the difference between this and Figure 14: the repeating-interval variant of Next aligns to standard calendar intervals, while the period variant of Next does not.

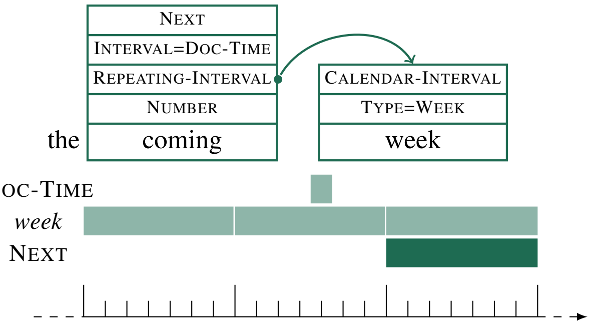

Figure 15: Next (Repeating-Interval) annotation: *the coming week*

This (Period) The first variant of the This operator takes an Interval and a Period, and creates an interval of the given length centered at the given interval: Formally:

$$\textsc{This}([t_1, t_2)\colon \textsc{Interval}, \Delta\colon \textsc{Period})\colon \textsc{Interval} \\ = \left[\frac{t_1 + t_2}{2} - \frac{\Delta}{2}, \frac{t_1 + t_2}{2} + \frac{\Delta}{2}\right)$$

Figure 16 shows an example This annotation along with a graphical depiction of its formal interpretation: a six-day-long interval centered around the document creation time.

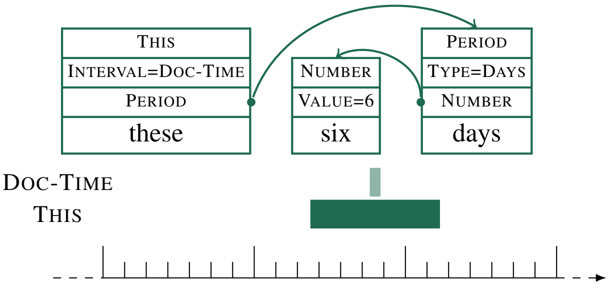

Figure 16: This (Period) annotation: *these six days*

This (Repeating Interval) The second variant of the This operator takes an Interval and a Repeating-Interval, and finds the repeated interval(s) containing the given interval. Formally:

$$\textsc{This}([t_1, t_2)\colon \textsc{Interval}, R\colon \textsc{R-Interval})\colon \textsc{R-Interval} \\ = \{[t_s, t_e) \in R : t_s \leq t_1 \wedge t_2 \leq t_e\}$$

Figure 17 shows an example This annotation along with a graphical depiction of its formal interpretation: the calendar week containing the document creation time.

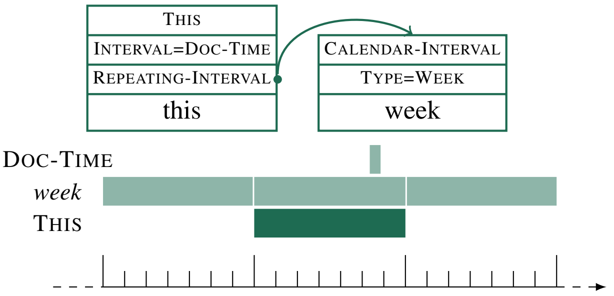

Figure 17: This (Repeating-Interval) annotation: *this week*

Before (Period) The first variant of the Before operator takes an Interval and a Period, and shifts the input interval earlier by the given period. Formally:

$$\textsc{Before}([t_1, t_2)\colon \textsc{Interval}, \Delta\colon \textsc{Period})\colon \textsc{Interval} \\ = [t_1 - \Delta, t_2 - \Delta)$$

Figure 18 shows an example Before annotation along with a graphical depiction of its formal interpretation: an interval the same length as the document creation time but one year earlier on the timeline.

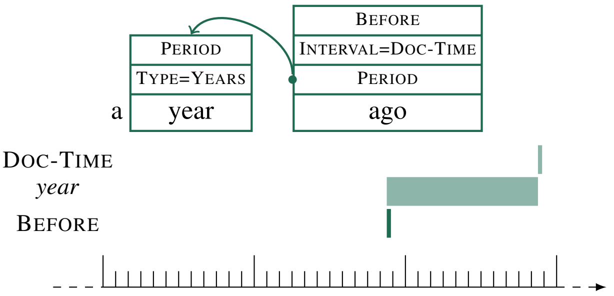

Figure 18: Before (Period) annotation: *a year ago*

Before (Repeating-Interval) The second variant of the Before operator takes Interval, a Repeating-

INTERVAL, and an integer n, and finds the n^{th} latest repeated interval before the input interval. Formally:

$$\textsc{Before}([t_1,t_2)\colon \textsc{Interval}, R\colon \textsc{R-Interval}, n\colon \mathbb{N}) \\ = n^{\text{th}} \text{ latest } \{[t_s,t_e) \in R : t_e \leq t_1\}$$

Figure 19 shows an example BEFORE annotation along with a graphical depiction of its formal interpretation: the second closest calendar interval named *Tuesday* that precedes the document creation time.

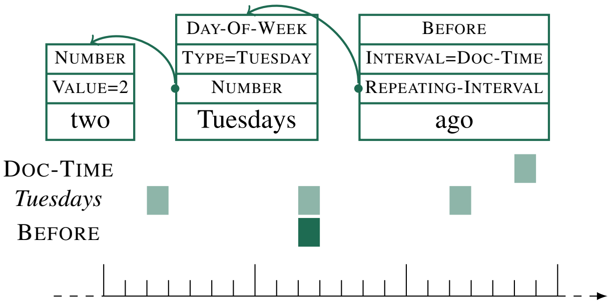

Figure 19: BEFORE (PERIOD) annotation: *two Tuesdays ago*

After (Period) The first variant of the AFTER operator takes an INTERVAL and a PERIOD, and shifts the input interval later by the given period. Formally:

$$\textsc{After}([t_1,t_2)\colon \textsc{Interval}, \Delta\colon \textsc{Period})\colon \textsc{Interval} \\ = [t_1+\Delta, t_2+\Delta)$$

Figure 20 shows an example AFTER annotation along with a graphical depiction of its formal interpretation: an interval the same length as the document creation time but one year later on the timeline.

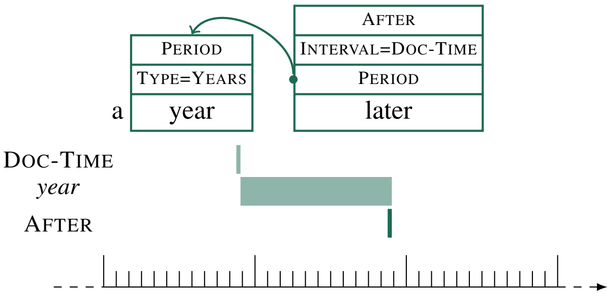

Figure 20: AFTER (PERIOD) annotation: *a year later*

After (Repeating-Interval) The second variant of the AFTER operator takes INTERVAL, a REPEATING-INTERVAL, and an integer n, and finds the n^{th} earliest repeated interval after the input interval. Formally:

$$\textsc{After}([t_1,t_2)\colon \textsc{Interval}, R\colon \textsc{R-Interval}, n\colon \mathbb{N}) \\ = n^{\text{th}} \text{ earliest } \{[t_s,t_e) \in R : t_2 \leq t_s\}$$

Figure 21 shows an example AFTER annotation along with a graphical depiction of its formal interpretation: the second closest calendar interval named *Tuesday* that follows the document creation time.

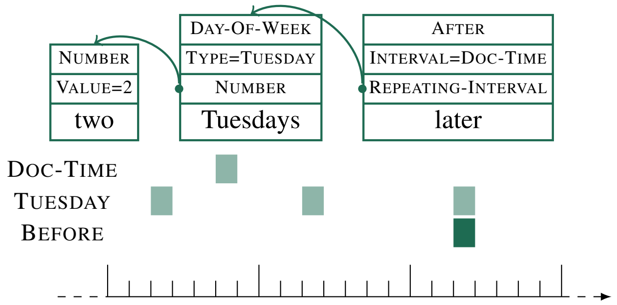

Figure 21: AFTER (PERIOD) annotation: *two Tuesdays later*

Between The BETWEEN operator finds the interval between two input intervals. Formally:

$$\textsc{Between}([t_1,t_2)\colon \textsc{Interval}, [t_3,t_4))\colon \textsc{Interval} \\ = [t_2,t_3)$$

Figure 22 shows an example BETWEEN annotation, along with a graphical depiction of its formal interpretation: the interval starting at the end of 1994 and ending at the document creation time.

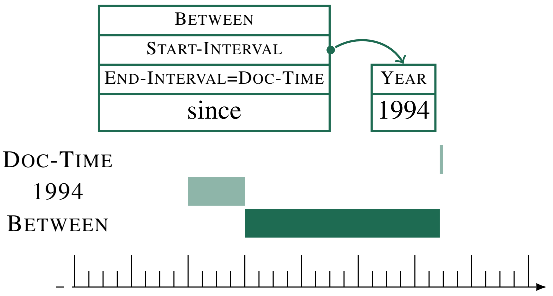

Figure 22: BETWEEN annotation: *since 1994*

Nth The NTH operator selects the n^{th} sub-interval of a repeating interval, counting from the start of another interval. Formally:

$$\textsc{Nth}([t_1,t_2)\colon \textsc{Interval}, R\colon \textsc{R-Interval}, n\colon \mathbb{N}) \\ = n^{\text{th}} \text{ of } \{[t_s,t_e) \in R : t_1 \leq t_s \wedge \leq t_e \leq t_2\}$$

Figure 23 shows an example NTH annotation, along with a graphical depiction of its formal interpretation: the fifth day following the start of the 2016 interval.

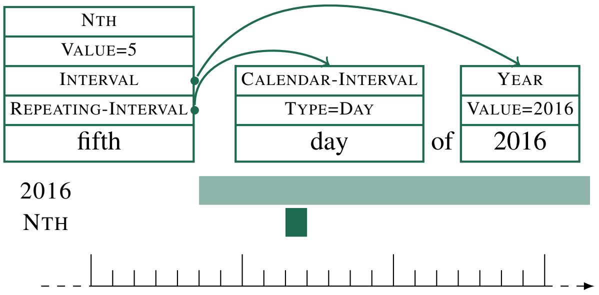

Figure 23: NTH annotation: *fifth day of 2016*

Two-Digit-Year The TWO-DIGIT-YEAR operator creates a one year interval from a two-digit number and the century of another interval. Formally:

$$\textsc{Two-Digit-Year}([t_1, t_2]\colon \textsc{Interval}, n\colon \mathbb{N}) \\ = [(C(t_1) + n)\text{-01-01}, (C(t_1) + n + 1)\text{-01-01})$$

where $C(x)$ is the year of x rounded down to the nearest hundred. Figure 24 shows an example Two-Digit-Year annotation, along with a graphical depiction of its formal interpretation: the year 96 in the century 1900 (i.e., 1996).

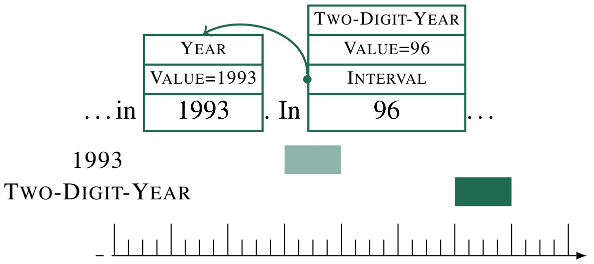

Figure 24: Two-Digit-Year annotation: *... in 1993. In 96...*

3.5. Other Annotations

In addition to periods, intervals, repeating intervals, and temporal operators, the schema includes a few annotations to handle other phenomena that are needed for the interpretation of time expressions.

Number annotations specify a numeric value.
Modifier annotations specify inexact time expressions.
Event annotations specify events that are anchor intervals.

Figure 25 shows an example with such annotations.

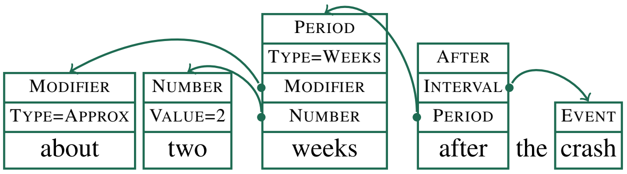

Figure 25: Number, Modifier, and Event annotations: *about two weeks after the crash*

4. Corpus Annotation

To test the validity of the schema, it was used to annotate news articles from the TimeBank corpus. Three annotators participated in the annotation process, all of whom had some background in linguistics and computer science: a faculty member with more than ten years of experience, one undergraduate student with two years of experience, and one undergraduate student with one semester of experience. The Anafora annotation tool (Chen and Styler, 2013) was used for the annotation. Anafora supports the kind of strongly typed annotations required by our schema, where the different properties of an annotation are restricted to different annotation types.

The first 10 documents in the corpus were used for schema refinement. All annotators annotated the documents, compared their annotations, and discussed disagreements. If revisions to the schema were necessary, they were made, and the annotation process was restarted. This process took roughly 50 hours of effort per annotator to converge.

4.1. Inter-Annotator Agreement

Once the schema had stabilized into the form described in the preceding sections, five new documents were selected that none of the annotators had previously seen. These documents were annotated independently by the two undergraduate students, with no discussion between annotators. This took a couple hours per annotator.

Agreement was then calculated as F_1, the harmonic mean of precision and recall[1], which in the case of two annotators reduces to:

$$F_1 = \frac{2 \cdot N_{1 \wedge 2}}{N_1 + N_2}$$

where N_1 (N_2) is the number of annotations from the first (second) annotator and $N_{1 \wedge 2}$ is the number of annotations matching between the two annotators. For span F_1, annotations were considered matching if they had the same annotation type and span (i.e., character offsets). For full F_1, annotations were only considered matching if the annotation type, span, and all properties and links matched exactly. For the overall F_1, we use “micro” F_1, where each annotation counts equally, regardless of type.

Table 1 shows the resulting inter-annotator agreement. Overall agreement reached F_1 of 0.917 for agreement on types and spans, and F_1 of 0.821 for agreement on all parts of the schema. The disagreements fell into 3 different categories:

missed annotations (44%) One annotator found a time concept that the other missed. For example, *08-08* in the header *2ndLd 08-08 0257 BC-Kenya-Embassy* actually does represent August 8, but only one of the annotators found this expression.

added annotations (32%) One annotator marked something as a time concept that should not have been. For example, one annotator marked *just* in *just minutes apart* as a Modifier(Type=Approx). However, since *just minutes* represents the same period as *minutes* does (i.e., *just minutes* is not an approximate version of *minutes*) *just* should not have been annotated as a Modifier.

wrong link (24%) Both annotators marked the same time concepts, but disagreed on how to link them. For example, consider the snippet:

> “The incidents of Aug. 7 underscore that terrorists know no boundaries...” said the U.S. Deputy Ambassador Peter Burleigh.

Both annotators identified *Aug. 7* as a Last (i.e., as referring to an August 7th closely preceding some anchor interval), but one annotator marked the document creation time as the anchor interval, and the other annotator marked *said* as the anchor interval. The latter is correct since the speech event could be years before the document creation time.

[1] Note that the κ coefficient (Cohen, 1960) converges to F_1 in cases like ours where the number of non-annotations is much larger than the number of annotations (Hripcsak and Rothschild, 2005).

Annotation Type	N_1	N_2	Span F_1	Full F_1
AMPM-Of-Day	1	1	1.000	1.000
After	3	3	1.000	0.000
Before	0	1	0.000	0.000
Between	1	0	0.000	0.000
Calendar-Interval	4	4	1.000	1.000
Day-Of-Month	22	23	0.933	0.889
Day-Of-Week	9	9	1.000	0.778
Event	5	2	0.000	0.000
Hour-Of-Day	12	12	1.000	1.000
Intersection	3	0	0.000	0.000
Last	15	15	0.933	0.733
Minute-Of-Hour	12	12	1.000	1.000
Modifier	4	2	0.333	0.333
Month-Of-Year	22	23	0.933	0.889
Nth	1	1	1.000	0.000
Number	5	5	1.000	1.000
Part-Of-Day	2	1	0.667	0.667
Period	8	8	1.000	0.750
Second-Of-Minute	5	5	1.000	1.000
This	4	5	0.889	0.667
Time-Zone	6	6	1.000	1.000
Two-Digit-Year	5	5	1.000	0.800
Year	10	10	1.000	1.000
Overall	159	153	0.917	0.821

Table 1: Inter-annotator agreement on the double-annotated sample from the TimeBank Corpus. N_1 (N_2) is the number of annotations identified by the first (second) annotator. Span F_1 is the agreement on just the types and spans of the annotations, while Full F_1 requires agreement on all the annotation properties and links as well.

4.2. The SCATE Corpus

Having achieved reasonable inter-annotator agreement, the student annotators then began to annotate the remainder of the TimeBank corpus. Table 2 shows the current size of our Semantically Compositional Annotation of Time Expressions (SCATE) corpus. The annotations may be downloaded from `https://github.com/bethard/anafora-annotations`.

	N
Documents	34
Annotations	1104
Month-Of-Year	139
Day-Of-Month	119
Last	112
Year	85
Day-Of-Week	77
Minute-Of-Hour	67
Hour-Of-Day	66
Period	64
This	56
Calendar-Interval	55
Number	43
Second-Of-Minute	37
Time-Zone	26
Event	26
Two-Digit-Year	25
Next	24
Modifier	17
Before	12
After	11
Between	9
Part-Of-Day	8
Intersection	6
Nth	5
Season-Of-Year	3
AMPM-Of-Day	3
Frequency	2
Union	1
Sum	1

Table 2: Annotations in the corpus as of March 2016

5. Discussion

The high level of inter-annotator agreement (0.821 F_1) in the first application of our time normalization scheme to a corpus is encouraging, and we believe the fine-grained, formally compositional nature of our annotation scheme is a significant advance over existing approaches such as ISO-TimeML. Others have attempted to provide formal semantics for ISO-TimeML, but they showed that "the formal semantic interpretation of the TimeML markup language is not nearly as straightforward as one might have expected" (Katz, 2007). Instead of trying to *post-hoc* assign a formal semantics to an annotation scheme, we have built our annotation scheme from the ground up to ensure it has a consistent formal interpretation.

There are a couple limitations of the research that we are currently working to address. First, the annotation scheme has been developed primarily with reference to examples from the TimeBank corpus, which may bias the expressiveness of the scheme towards expressions common in newswire. To address this, we have obtained the THYME corpus of clinical notes (Styler et al., 2014), and will soon begin testing the scheme on medical language. Second, while the scheme fully formalizes the semantic composition of the time elements, some of the operations are not easy for humans to carry out manually (e.g., subtracting n weeks from a date). To address this, we are currently developing a Scala/Java library that can read in annotations from the schema and produce the corresponding machine readable timeline intervals. This library will be available from `https://github.com/bethard/timenorm`.

Acknowledgments

The project described was supported in part by R01LM010090 (THYME) from the National Library Of Medicine. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health.

References

Chen, W.-T. and Styler, W. (2013). Anafora: A web-based general purpose annotation tool. In *Proceedings of the*

2013 NAACL HLT Demonstration Session, pages 14–19, Atlanta, Georgia, June. Association for Computational Linguistics.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46.

Ferro, L., Gerber, L., Mani, I., Sundheim, B., and Wilson, G. (2005). Tides 2005 standard for the annotation of temporal expressions. Technical report.

Fischer, F. and Strötgen, J. (2015). When does german literature take place? – on the analysis of temporal expressions in large corpora. In *Proceedings of DH 2015: Annual Conference of the Alliance of Digital Humanities Organizations*, Sydney, Australia, 6.

Hripcsak, G. and Rothschild, A. S. (2005). Agreement, the f-measure, and reliability in information retrieval. *Journal of the American Medical Informatics Association*, 12(3):296–298.

ISO 8601. (2004). Data elements and interchange formats – information interchange – representation of dates and times. Technical report, Geneva, Switzerland.

Katz, G., (2007). *Annotating, Extracting and Reasoning about Time and Events: International Seminar, Dagstuhl Castle, Germany, April 10-15, 2005. Revised Papers*, chapter Towards a Denotational Semantics for TimeML, pages 88–106. Springer Berlin Heidelberg, Berlin, Heidelberg.

Lin, C., Karlson, E. W., Dligach, D., Ramirez, M. P., Miller, T. A., Mo, H., Braggs, N. S., Cagan, A., Gainer, V., Denny, J. C., and Savova, G. K. (2014). Automatic identification of methotrexate-induced liver toxicity in patients with rheumatoid arthritis from the electronic medical record. *Journal of the American Medical Informatics Association*.

Llorens, H., Derczynski, L., Gaizauskas, R., and Saquete, E. (2012). TIMEN: An open temporal expression normalisation resource. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, May. European Language Resources Association (ELRA).

Mazur, P. and Dale, R. (2010). WikiWars: A new corpus for research on temporal expressions. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 913–922, Cambridge, MA, October. Association for Computational Linguistics.

Pustejovsky, J., Hanks, P., Saurí, R., See, A., Gaizauskas, R., Setzer, A., Radev, D., Sundheim, B., Day, D., Ferro, L., and Lazo, M. (2003). The TimeBank corpus. In *Corpus linguistics*, volume 2003, pages 647–656.

Pustejovsky, J., Lee, K., Bunt, H., and Romary, L. (2010). Iso-timeml: An international standard for semantic annotation. In Nicoletta Calzolari (Conference Chair), et al., editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, may. European Language Resources Association (ELRA).

Styler, IV, W. F., Bethard, S., Finan, S., Palmer, M., Pradhan, S., de Groen, P. C., Erickson, B., Miller, T., Lin, C., Savova, G., and Pustejovsky, J. (2014). Temporal annotation in the clinical domain. *Transactions of the Association for Computational Linguistics*, 2:143–154, 4.

UzZaman, N., Llorens, H., Derczynski, L., Allen, J., Verhagen, M., and Pustejovsky, J. (2013). SemEval-2013 Task 1: TempEval-3: Evaluating Time Expressions, Events, and Temporal Relations. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 1–9, Atlanta, Georgia, USA, June. Association for Computational Linguistics.

Verhagen, M., Gaizauskas, R., Schilder, F., Hepple, M., Katz, G., and Pustejovsky, J. (2007). SemEval-2007 Task 15: TempEval Temporal Relation Identification. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 75–80, Prague, Czech Republic, June. Association for Computational Linguistics.

Verhagen, M., Sauri, R., Caselli, T., and Pustejovsky, J. (2010). SemEval-2010 Task 13: TempEval-2. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 57–62, Uppsala, Sweden, July. Association for Computational Linguistics.