Chapter 5
Using Temporal Signals

Words are but the signs of ideas.

Preface to the Dictionary
SAMUEL JOHNSON

5.1 Introduction

In Chap. 4, we saw that a proportion of difficult temporal relations were associated with a particular separate word or phrase that described the temporal relation type – a **temporal signal**. The failure analysis in Sect. 4.3.1 finds signals to be of use in over a third of difficult TLINKs. Despite their demonstrable impact on temporal link labelling (see Sect. 3.5.4), no work has been undertaken toward the automatic annotation of temporal signals, and little toward their exploitation. This chapter begins to address these deficiencies.

Temporal signals (also known as temporal conjunctions) are discourse markers that connect a pair of events and times and explicitly state the nature of their temporal relation. Humans resolve events and times in discourses that machines cannot yet automatically label. It is assumed that there must be information in the document and in world knowledge that allows resolution of events, times and relations between them. Temporal signals form part of this information. Intuitively, these words contain temporal ordering information that human readers can access. This chapter investigates the role that temporal signals play in discourse and finds methods for automatically annotating them.

To illustrate:

Example 9 *"The exam papers were submitted before twelve o'clock."*

In Example 9 there is an event, the submitting of exam papers, and a time, twelve o'clock, that are temporally related. The word *before* serves as a signal that describes the nature of the temporal relation between them.

These temporal signals can occur with difficult temporal links and seem to provide explicit information about temporal relation type. It is worth investigating their

L.R.A. Derczynski, *Automatically Ordering Events and Times in Text*,
Studies in Computational Intelligence 677, DOI 10.1007/978-3-319-47241-6_5

potential utility in the relation typing task. If these signals are found to be useful, we may determine how to detect and use them automatically, instead of relying on existing manual annotations. To begin investigation the process of automatic signal annotation, a thorough account of temporal signals is required, followed by an examination of current resources that include temporal signal annotations. Next one may cast the signal annotation problem as a two step process. Firstly, one must know how to determine which words and phrases in a given document are temporal signals. Secondly, one needs to work out with which intervals a given temporal signal is associated, given many candidates. The tasks jointly comprise automatic temporal signal annotation.

This chapter is therefore structured as follows. In Sect. 5.2, we formally introduce background material regarding temporal signals. Section 5.3 reports on the effect that signal information has on an existing relation typing approach compared with the approach's performance sans signal information, finding that adding features that describe temporal signals yields a large error reduction for automatic relation typing. Accordingly, after surveying signal annotations in existing corpora (Sect. 5.4), a method for automatically finding words and phrases that occur as temporal signals is introduced, which first requires the construction of a high-quality ground truth dataset (Sect. 5.5). After developing an approach to finding temporal signal expressions using this new dataset (Sect. 5.6), Sect. 5.7 describes a method for associating temporal signal (once found) with a pair of temporally-related intervals whose relation is described by the temporal signal. The overall performance of the presented temporal signal annotation system is then evaluated. The chapter concludes with an evaluation of the impact this automatic signal annotation has on the overall relation typing task (Sect. 5.8), which is a positive one.

5.2 The Language of Temporal Signals

Signal expressions explicitly indicate the existence and nature of a temporal relation between two events or states or between an event or state and a time point or interval. Hence a temporal signal has two arguments, which are the temporal "entities" that are related. One of these arguments may be deictic instead of directly attached to an event or time; anaphoric temporal references are also permitted. For example, the temporal function and arguments of *after* in "*Nanna slept after a long day at work*" are clear and are available in the immediately surrounding text. With "*After that, he swiftly finished his meal and left*" we must look back to the antecedent of *that* to locate the second argument.

Sometimes a signal will appear to be missing an argument; for example, sentence-initial signals with only one event in the sentence (*"Later, they subsided."*). These signals relate an event in their sentence with the discourse's current temporal focus – for example, the document creation time, or the previous sentence's main event.

Signal surface forms have a compound structure consisting of a **head** and an optional **qualifier**. The head describes the temporal operation of the signal phrase and

the qualifier modifies or clarifies this operation. An example of an unqualified signal expression is *after*, which provides information about the nature of a temporal link, but does not say anything about the absolute or relative magnitude of the temporal separation of its arguments. We can elaborate on this magnitude with phrases which give qualitative information about the relative size of temporal separation between events (such as *very shortly after*), or which give a specific separation between events using a duration as a modifying phrase (e.g. *two weeks after*). In both cases, the signal applies to the ordering of events either side of the separation, rather than the separation itself.

5.2.1 Related Work

Signals help create well-structured discourse. Temporal signals can provide context shifts and orderings [1]. These signal expressions therefore work as discourse segmentation markers [2]. It has been shown that correctly including such explicit markers makes texts easier for human readers to process [3].

Further, words and phrases that comprise signals are sometimes polysemous, occurring in temporal or non-temporal senses. For the purposes of automatic information extraction, this introduces the task of determining when a given candidate signal is used in a temporal sense.

Brée [4] performed a study of temporal conjunctions and prepositions and suggested rules for discriminating temporal from non-temporal uses of signal expressions that fall into these classes. Their approach relies heavily upon the presentation of contrasting examples of each signal word. This research went on to describe the ambiguity of nine temporal prepositions in terms of their roles as temporal signals [5].

Schlüter [6] identifies signal expressions used with the present perfect and compares their frequency in British and US English. This chapter later attempts a full identification of English signal expressions.

Vlach [7] presents a semantic framework that deals with duratives when used as signal qualifiers (see above). Our work differs from the literature in that is it the first to be based on gold standard annotations of temporal semantics and that it encompasses all temporal signal expressions, not just those of a particular grammatical class.

Intuitively, signal expressions contain temporal ordering information that human readers can access easily. Once temporal conjunctions are identified, existing semantic formalisms may be readily applied to discourse semantics. It is however ambiguous which temporal relation any given signal attempts to convey, as investigated by [8] and studied in TimeBank later in this chapter (Sect. 5.4.2). Our work quantifies this ambiguity for a subset of signal expressions.

5.2.2 Signals in TimeML

This section includes work from [9].

TimeML's description of a signal is[1]:

> SIGNAL is used to annotate sections of text, typically function words, that indicate how temporal objects are to be related to each other. The material marked by SIGNAL constitutes the following:
>
> - indicators of temporal relations such as temporal prepositions (e.g. *"on"*, *"during"*) and other temporal connectives (e.g. *"when"*) and subordinators (e.g. *"if"*). This functionality of the SIGNAL tag was introduced by [10].
> - indicators of temporal quantification such as *"twice"*, *"three times"*.

Signals in TimeML are used to mark words that indicate the type of relation between two intervals and also to indicate multiple occurrences of events (temporal quantification). For the task of temporal relation typing, we are only interested in this former use of signals. The annotation guidelines suggest that in TimeML one should annotate a minimal set of tokens – typically just the "head" of the signal.

For example, in the sentence *John smiled after he ate*, the word *after* specifies an event ordering. Example 10 shows this sentence represented in TimeML.

Example 10

```
John <EVENT id="e1"> smiled </EVENT> <SIGNAL id="s1"> after </SIGNAL>
he <EVENT id="e2"> ate </EVENT> .
<TLINK id="l1" eventID="e1" relatedToEvent="e2"
  relType="AFTER" signalID="s1" />
```

TimeML allows us to associate text that suggests an event ordering (a SIGNAL) with a particular temporal relation (a TLINK). To avoid confusion, it is worthwhile clarifying our use of the term "signal". We use **SIGNAL** in capitals for tags of this name in TimeML and **signal/signal word/signal phrase** for a word or words in discourse that describe the temporal ordering of an event pair. Examples of the signals found in TimeBank are provided in Table 5.1.

It is important to note that not every occurrence of text that could be a signal is used as a temporal signal. Some signal words and phrases are polysemous, having both temporal and non-temporal senses: e.g. *"before"* can indicate a temporal ordering (*"before 7 o'clock"*) or a spatial arrangement (*"kneel before the king"*). This book refers to expressions that could potentially be temporal signals as **candidate signal phrases**. Only candidate signal phrases occurring in a temporal sense are of interest.

The signal text alone does not mean a single temporal interpretation. A temporal signal word such as *after* (for example) is used in TimeBank in TLINKs labelled AFTER, BEFORE and INCLUDES. For example, there is no set convention to the order in which a TLINK's arguments should be defined; the AFTER TLINK in Example 10 could just as well be encoded as:

```
<TLINK id="l1" eventID="e2" relatedToEvent="e1"
  relType="BEFORE" signalID="s1" />
```

[1] TimeML Annotation Guidelines, http://timeml.org/site/publications/specs.html.

Table 5.1 A sample of phrases most likely to be annotated as a signal when they occur in TimeBank. All corpus data was provided by the CAVaT tool [11]

Phrase	Corpus freq.	Occurrences as signal	Likelihood of being a signal (%)
subsequently	3	3	100
after	72	67	93
's	10	8	80
follows	4	3	75
before	33	23	70
until	36	25	69
during	19	13	68
as soon as	3	2	67

See Table 5.2 for the distribution of relation labels described by a subset of signal words and phrases.

As described above, signals sometimes reference abstract points as their arguments. These abstract points might be a reference time (Sect. 6.3) or an implicit anaphoric reference. As TimeML does not include specific annotation for reference time, one should instead assume that the signal co-ordinates its non-abstract argument with the interval at which reference time was last set. For example, in *"There was an explosion Tuesday. Afterwards, the ship sank"*, we will link the *sank* event with *explosion* (the previous head event) and then associate our signal with this link.

5.3 The Utility of Temporal Signals

Do signals help temporal relation typing? Given the role that they might play in the relation typing task suggested in Sect. 4.3.1 and having a high-level definition of temporal signals, it is next important to establish their potential utility. Since we have in TimeML a signal-annotated corpus, to answer this question, one can compare the performance of automatic relation typing systems with and without signal information. Positive results would motivate investigation into further work on automatic signal annotation. This section relates such a comparison, and includes work from [12]. An extended investigation into this section's findings can be found in [13].

Although accurate event ordering has been the topic of research over the past decade, most work using the temporal signals present in text has been only preliminary. However, as noted in Chap. 3, specifically focusing on temporal signals when

Table 5.2 Signal expressions and the TimeML relations that they can denote. Counts do not match because a single signal expression can support more than one temporal link

Signal expression	TLINK count	AFTER	BEFORE	BEGINS	BEGUN_BY	DURING	ENDED_BY	ENDS	IAFTER	IBEFORE	INCLUDES	IS_INCLUDED	SIMULTANEOUS
after	76	62	3	4				5	2				
when	57	16	3	1	2	1			1	1	9	9	14
until	37	4	7	1			21	1	1	2			
before	36	1	28	2			1	2	1	1			
since	19	9	1	2	7								
already	13		6								4	3	
previously	18	6	12										
while	9												9
meanwhile	9		1			2						1	5
followed	4	2	2										
former	12		12										

Table 5.3 TLINKs and signals in the largest TimeML-annotated corpora

Corpus	Total TLINKs	With SIGNAL	Without SIGNAL
TimeBank v1.2	6418	718 (11.2 %)	5700
AQUAINT TimeML v1.0	5365	178 (3.3 %)	5187
ATC (combined)	11783	896 (7.6 %)	10887
ATC event-event	6234	319 (5.1 %)	5915

classifying temporal relations can yield a performance boost. This section attempts to measure that performance boost.

In TimeML, a signal is either text that indicates the cardinality of a recurring event, or text that explicitly states the nature of a temporal relation. Only the latter sense is interesting for the current work. This class of words and phrases includes temporal conjunctions (e.g. *after*) and temporal adverbials (e.g. *currently*, *subsequently*), as well as set phrases (e.g. *as soon as*). A minority of TLINKs in TimeML corpora are annotated with an associated signal (see Table 5.3).

While the processing of temporal signals for TLINK classification could potentially be included as part of feature extraction for the relation typing task, temporal signals are complex and useful enough to warrant independent investigation. When the final goal is TLINK labelling, once salient features for signal inclusion and representation have been found, one might skip signal annotation entirely and include these features in a temporal relation type classifier. As we are concerned with the characterisation and annotation of signals, we do not address this possibility here, instead attempting to understand signals as an intermediate step towards better overall temporal labelling.

The following experiment explores the question of whether signal information can be successfully exploited for TLINK classification by contrasting relation typing with and without signal information. The approach replicated as closely as possible is that of [14], briefly summarised as follows.

The replication had three steps. Firstly, to simplify the problem, the set of possible relation types was reduced (folded) by applying a mapping (see Sect. 3.3.1). For example, as a BEFORE b and b AFTER a describe the same ordering between events a and b, we can flip the argument order in any AFTER relation to convert it to a BEFORE relation. This simplifies training data and provides more examples per temporal relation class. Secondly, the following information from each TLINK is used as features: event class, aspect, modality, tense, negation, event string for each event, as well as two boolean features indicating whether both events have the same tense or same aspect. Thirdly, we trained and evaluated the predictive accuracy of the maximum entropy classifier from Carafe.[2] To match the original approach, ten-fold cross-validation was used, and a one-third/two-thirds split was also introduced to see the effect of reduced ratio of training:evaluation examples. This split the set of event-event TLINKs into a training set of 4156 instances and an evaluation set of 2078 instances.

[2] Available at http://sourceforge.net/projects/carafe/.

Table 5.4 Results from replicating a prior experiment on automatic relation typing of event-event relations

	Corpus	XV accuracy (%)	Train/Eval split (%)	Baseline (%)
Mani et al. results	AQ + TimeBank 1.2a	61.79		51.6
Replicated results	AQ + TimeBank 1.2	60.32	60.04	53.34

In [14], TLINK data came from the union of TimeBank v1.2a and the AQUAINT TimeML corpora. As the TimeBank v1.2a corpus used is not publicly available, we used TimeBank v1.2. This use of a publicly-available version of TimeBank instead of a private custom version was the only change from the previous work. In this work we only examine event-event links, which make up 52.9 % of all TLINKs in our corpus, likely due to minor differences between the TLINK annotations of TimeBank v1.2 and TimeBank v1.2a.

Table 5.4 shows results from replicating the previous experiment on event-event TLINKs. The baseline listed is the most-common-class in the training data. This gives a similar score of 60.32 % accuracy compared to 61.79 % in the previous work. The differences may be attributed to the non-standard corpus that they use. The TLINK distribution over a merger of TimeBank v1.2 and the AQUAINT corpus differs from that listed in the paper.

5.3.1 Introducing Signals to the Relation Labelling Feature Set

Now that a reasonable replication of a prior approach has been established, the goal is to measure the difference in relation typing performance that temporal signals make. This requires feature representations of signals. To add information about signals to our training instances, we use the extra features described below; the two arguments of a TLINK are represented by **e1** and **e2**. All features can be readily extracted from the existing TimeML annotations. Only gold-standard signal annotations from the corpora were used.

- **Signal phrase.** This shows the actual text that was marked up as a SIGNAL. From this, we can start to guess temporal orderings based on signal phrases. However, just using the phrase is insufficient. For example, the two sentences *Run before sleeping* and *Before sleeping, run* are temporally equivalent, in that they both specify two events in the order run-sleep, signalled by the same word *before*.
- **Textual order of e1/e2.** It is important to know the textual order of events and their signals even when we know a temporal ordering. Textual order can have a direct effect on the temporal order conveyed by a signal. To illustrate, *"Bob washes before he eats"* describes a story different from *"Before Bob washes he eats"*.

- **Textual order of signal and e1, signal and e2.** These features describe the textual ordering of both TLINK arguments and a related signal. It will also help us see how the arguments of TLINKs that employ a particular signal tend to be textually distributed. The features are required to disambiguate cases where textual order is unreliable. To illustrate, *"Bob washes before he eats"* and *"Before he eats, Bob washes"* describe the same event ordering but have different text orderings.
- **Textual distance between e1/e2.** Sentence and token count between e1 and e2.
- **Textual distance from e1/e2 to SIGNAL.** If we allow a signal to influence the classification of a TLINK, we need to be certain of its association with the link's events. Distances are measured in tokens.
- **TLINK class given SIGNAL phrase.** Most likely TLINK classification in the training data given this signal phrase (or empty if the phrase has not been seen). Referred to as signal **hint**. Referred to as signal **hint**.

5.3.2 TLINK Typing Results Using Signals

Table 5.5 shows the results of adding features for temporal signals to the basic TLINK relation typing system. Moving to a feature set which adds SIGNAL information, including signal-event word order/distance data, 61.46 % predictive accuracy is reached. The increase is small when compared to 60.32 % accuracy without this information, but TLINKs that employ a SIGNAL in are a minority in our corpus (possibly due to under-annotation).

The low magnitude of the performance increase seen in Table 5.5 could be due to the way in which training examples are selected. There are in total 11 783 TLINKs in the combined corpus, of which 7.6 % are annotated including a SIGNAL; for just TimeBank v1.2, the figure is higher at 11.2 % (see Table 5.3 and also Fig. 5.1). The proportion of signalled TLINKs in our data – event-event links in the combined AQUAINT/TimeBank 1.2 corpus – is lowest at 5.1 %. It is possible that signalled TLINKs are classified significantly better using this extended feature set, but account for such a small part of this dataset that the overall difference is small. To test this, the experiment is repeated, this time splitting the dataset into signalled and non-signalled TLINKs.

Table 5.5 TLINK classification with and without signal features, using both 10-fold cross validation and a one-third/two-thirds split between evaluation and training data

Predictive accuracy	XV	Split (%)
Baseline (most common class)	53.34 %	53.34
Without signal features	60.32 %	60.04
With basic signal features	**61.46 %**	60.81
With signal features including hint	n/a	**61.98**

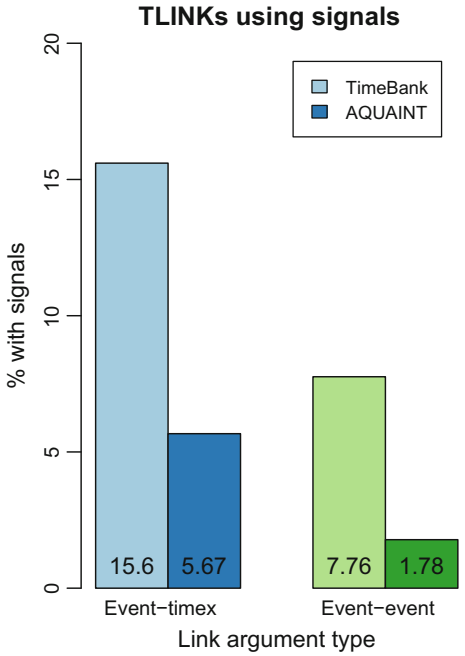

Fig. 5.1 Signalled TLINKs by argument type (event-event or event-tlink) in TimeBank 1.2 and the AQUAINT TimeML corpus. The *paler columns* correspond to TimeBank, the *darker* AQUAINT

If there is no performance difference between feature sets when classifying TLINKs that *do* use signals, then our hypothesis is incorrect, or the features used are insufficiently representative. If signals are helpful, and our features capture information useful for temporal ordering, we expect a performance increase when automatically classifying signalled TLINKs. Results in Table 5.6 support our hypothesis that

Table 5.6 Predictive accuracy from Carafe's maximum entropy classifier, using features that do or do not include signal information, over signalled and non-signalled event-event TLINKs in ATC. The baseline is accuracy when the most-common-class is always assigned

	Cross validation		Train/Eval split	
Predictive accuracy	Unsignalled (%)	Signalled (%)	Unsignalled (%)	Signalled (%)
Baseline (most common class)	52.68	62.41	52.68	62.41
Plain features	**62.05**	55.65	61.81	60.32
Plain, signal features	62.05	**69.57**	61.81	**82.19**
Plain, signal features, hint	62.05	41.72	–	–

signals are useful, but we are performing nowhere near the maximum level suggested above. Data sparsity is a problem here, as the combined corpus only contains 319 suitable TLINKs, and both source corpora show evidence of signal under-annotation. The results also suggest that the signal hint feature was not helpful; this is the same result found by [15].

Exploring the strongest feature set (basic+signals; no hint), and attempting to combat the data sparsity problem, we used 10-fold cross validation instead of a split; results are also in Table 5.6. This again shows a distinct improvement in the predictive accuracy of signalled TLINKs using this feature set over the features in previous work. Cross-validation also gives better overall accuracy. This is likely because of the low volumes of training data mean that the real difference in number of examples between 10-fold cross validation and a one-third/two-thirds split can make a large contribution to classifier performance.

5.3.3 Utility Assessment Summary

When learning to classify signalled TLINKs, there is a significant increase in predictive accuracy when features describing signals are introduced. This suggests that signals are useful when it comes to providing information for classifying temporal links, and also that the features we have used to describe them are effective.

Now that it is confirmed that signals are helpful in temporal relation typing, the next task is to determine how to annotate them automatically. A good account of existing resources may give clues for this process. After this, one needs to explore how to discriminate whether or not a candidate signal expression is used as a temporal signal in text. Next, after finding a temporal signal, we need to determine which intervals it temporally connects. Finally, we can attempt to annotate a temporal link based on the signal.

5.4 Corpus Analysis

In order to understand temporal signals, this section investigates the role of hand-annotated temporal signals in the TimeBank dataset. Further, casual examination reveals that words acting in a temporal signal role in existing datasets are not always annotated as such. Under-annotation can depend on how well the annotator understands the task, and the clarity of annotation guidelines. This section discusses the TimeML definition of signals and describes an augmented corpus which has received extra annotation.

Using the TimeBank corpus, we set out to answer the following questions:

1. Of the expressions which can function as temporal signals, what proportion of their usage in the TimeBank corpus is as a temporal signal? E.g. how ambiguous are these expressions in terms of their role as temporal signals?
2. Of the occurrences of these expressions as temporal signals, how ambiguous are they with respect to the temporal relation they convey?

The following section (which includes material from [9]) provides provisional answers to these questions – provisional as one of the difficulties we encountered was significant under-annotation of temporal signals in TimeBank. We have addressed this to some extent, but more work remains to be done. Nonetheless we believe the current study provides important insights into the behaviour of temporal signals and how they may be exploited by computational systems carrying out the temporal relation detection task.

5.4.1 Signals in TimeBank

The TimeML <SIGNAL> element bounds a lexicalised temporal signal. Summary information on the SIGNAL elements in TimeBank 1.2 is in Table 5.7 and the number of links per signal in Table 5.8. Although permitted under TimeML 1.2.1 for denoting cardinality, no signals have been assigned to event instances for this purpose, although there is one unassigned signal annotation that does indicate event cardinality.

Table 5.7 How <SIGNAL> elements are used in TimeBank

Annotated SIGNAL elements	758
Signals used by a TLINK	721
Signals used by an ALINK	1
Signals used by a SLINK	39
TLINKs that use a SIGNAL	787
Signals used by more than one TLINK	54

Table 5.8 The number of TLINKs associated with each temporal signal word/phrase, in TimeBank. Signals not used on TLINKs (e.g. those used on aspectual or subordinate links, or for event cardinality) are excluded. The distribution appears to be Zipfian [16]

Argument pairs co-ordinated	Frequency
1	597
2	41
3	12
5	1

In cases where a specific duration occurs as part of a complex qualifier-head temporal signal, e.g. *two weeks after*, TimeBank has followed the convention that the signal head alone is annotated as a SIGNAL and the qualifier is annotated as a TIMEX3 of type DURATION.

5.4.2 Relation Type Ambiguity

The nature of the temporal relation described by a signal is not constant for the same signal phrase, though each signal tends to describe a particular relation type more often than other types. Table 5.2 gives an excerpt of data showing which temporal relations are made explicit by each signal expression. The variation in relation type associated with a signal is not as great as it might appear as the assignment of temporal relation type has an element of arbitrariness – one may choose to annotate a BEFORE or AFTER relation for the same event pair by simply reversing the temporal link's argument order, for example. There is no TimeML convention regarding how TLINK annotation arguments should be ordered. Nevertheless, it is possible to draw useful information from the table; for example, one can see that *meanwhile* is much more likely to suggest some sort of temporal overlap between events than an ordering where arguments occur discretely.

5.4.2.1 Closed Class of Signals

To what extent are the words sometimes annotated as temporal signals in TimeBank actually used as time relaters?

As temporal signals and phrases are likely to be a closed class of words, our approach is to first define a set of temporal signal candidate words. For each occurrence of one of these words in a discourse, we will decide if it is a temporal signal or not.

Because they do not contribute to temporal ordering, annotated signals that indicate the cardinality of recurring events were removed before experimentation. We have derived a closed class of 102 signal words and phrases from [17] (see for example Sect. 10.5, "Time Relaters"), given in Table 5.9. This list is long but may not be comprehensive. Automatic signal annotation can be approached by finding words in a given document that are both within this closed class of candidate signal phrases and also occur having a temporal sense. TimeBank contains 62 unique signal words and phrases (ignoring case), annotated in 688 SIGNAL elements and used by 718 TLINKs. Of these 62, over half (39) are also found in our list above. The remaining 23 signals correspond to only 45 signal mentions, supporting 46 temporal links. Thus, if we can perfectly annotate every signal we find in text based on our closed class, we will have described 93.1 % of TLINK-supporting signals and be better able to label 93.6 % of TLINKs that have a supporting signal.

Table 5.9 A closed class of temporal signal expressions

after	ensuing	meantime	soon
afterwards	eventually	momentarily	still
again	fifthly	next	subsequent
already	finally	ninethly	subsequently
as	first	now	succeeding
as soon as	firstly	nowadays	suddenly
as yet	following	on	supervening
at	for	once	then
at once	forever	originally	thereafter
at this point	for ever	over	thirdly
before	former	past	through
beforehand	formerly	preceding	throughout
between	fourthly	presently	til
by	frequently	previous	till
coexisting	from	previously	to
coinciding	here	prior	up to
concurrent	hitherto	recently	until
concurrently	immediately	secondly	when
contemporaneous	in	seventhly	whenever
contemporaneously	initially	shortly	while
contemporary	instantly	simultaneous	whilst
directly	last	simultaneously	within
during	late	since	yet
earlier	lately	sixthly	's
early	later	so long as	
eighthly	meanwhile	sometime	

To provide a surface characterisation of the role signals play, the distribution of their part of speech tag (from PTB) over signals in TimeBank is given in Table 5.10. Many uses are as prepositions, perhaps for attaching events to each other by means of prepositional phrases.

Of the closed class entries detailed in Table 5.9, 25 entries occur in the corpus but are never annotated as signal text: *again*, *directly*, *early*, *finally*, *first*, *here*, *last*, *late*, *next*, *now*, *recently*, *eventually*, *forever*, *formerly*, *frequently*, *initially*, *instantly*, *meantime*, *originally*, *prior*, *shortly*, *sometime*, *subsequent*, *subsequently* and *suddenly*.

We could also derive an alternative signal list by extracting all phrases that are found as the first child of `SBAR-TMP` constituent tags, as suggested in Dorr and Gaasterlaand [18]. For example, in Fig. 5.2 (an automatically parsed and function-tagged sentence from TimeBank's `wsj_0520.tml`), the first child of the `SBAR-TMP` constituent is a one-leaf IN tag. The text is *after*, which we would treat as

Table 5.10 Distribution of part-of-speech in signals and the first word of signal phrases

Part of speech	Frequency	Proportion (%)
IN	521	77.3
RB	73	10.8
WRB	53	7.9
JJ	14	2.1
RBR	5	0.7
VBG	4	0.6
CC	2	0.3
RP	1	0.1
JJR	1	0.1

Fig. 5.2 An example SBAR-TMP construction around a temporal signal

```
(S1 (S (NP-SBJ (NNP Nashua))
       (VP (VBD announced)
         (NP (DT the) (NNP Reiss) (NN request))
         (SBAR-TMP (IN after)
           (S (NP-SBJ (DT the) (NN market))
             (VP (VBD closed))))) (. .)))
```

Table 5.11 The set of signal words and phrases suggested by the SBAR-TMP model, broken into correctly and incorrectly detected phrases

Correct examples	Incorrect examples
after	at least
as	as surely
before	several months
once	nearly two months
since	even
until	only
while	soon
when	

a temporal signal. This approach returns a restrictive set of temporal signals, shown in Table 5.11, though contains few false positives.

5.4.3 Temporal Versus Non-temporal Uses

The semantic function that a temporal signal expression performs is that of relating two temporal entities. However, the words that can function as temporal signals also play other roles.

For example, one may use *before* to indicate that one event happened temporally prior to another. This word does not always have this meaning.

Example 12 "I will drag you before the court!"

In Example 12, the reading is that one will be summoned to appear in front of the court – the spatial sense – and not that the reader will be dragged, and then later the court will be dragged. It is important to know the correct sense of these connective words and phrases.

Of all temporal relations (TLINKs) in the English TimeBank, 11.2 % use a temporal signal in the original annotation (Table 5.3). It is important to note that some instances of signal expressions are used by more than one temporal link; see Table 5.8 for details. The most frequent signal word was "in", accounting for 24.8 % of all signal-using TLINKs. However, only 13.3 % of occurrences of the word "in" have a temporal sense. The word "after" is far more likely to occur in a temporal sense (91.7 % of all occurrences).

As an aside, the notion that temporal signals might be easily picked out based upon word class may be dispelled by examining the distribution of parts-of-speech possessed by temporal signals – see Table 5.10. Part of speech is not a reliable disambiguator of sense, in this case.

5.4.4 *Parallels to Spatial Representations in Natural Language*

Time and space are related and often an event will be positioned in both. Language used for describing time and language used for describing space are often similar, not least in the fact they they both use signals and often even use the same words as signals. Temporal signals relate a pair of temporal intervals, and spatial signals relate a pair of regions. Although not the focus of this chapter, it is useful to note the common and contrasting behaviours of temporal and spatial signals that emerged during investigation.

SpatialML [19] is an annotation scheme for spatial entities and relations in discourse.[3] Among other things it includes elements for annotating relations between spatial entities.

Links in SpatialML may be topological or relative. Topological links include containment, connection and other links from a fixed set based on the RCC8 calculus. SpatialML relative links, on the other hand, express spatial trajectories between locations.

In the revised ACE 2005 SpatialML annotations,[4] 97.5 % of all RLINKs (the SpatialML representation for a relative spatial link) have at least one accompanying textual signal (See Table 5.12). Compared to TimeBank's 11.2 % of TLINKs having a signal, SpatialML relative links are much more likely to use an explicit signal

[3] Although SpatialML has now been superseded by ISO-Space, we are concerned in this section with a SpatialML annotated corpus; there is no ISO-Space equivalent at the time of writing.

[4] LDC catalogue number LDC2011T02.

Table 5.12 Frequency of signal usage for different types of spatial link in the ACE 2005 English SpatialML Annotations Version 2

Link type	SpatialML element	Occurrences	Signalled	Signalling rate (%)
Relative	RLINK	80	78	97.5
Topological	LINK	378	7	1.85

than TimeML temporal relations. This may be because the mechanisms available in language for expressing temporal relations are wider than those for relating spatial entities. For example, to relate events in English, one may choose to use a tense and aspect (which involves inflection or added auxiliaries) instead of adding a signal word. Furthermore, there are three spatial dimensions in which to describe an entity; in contrast, the arrow of time supplied a single unidirectional dimension, which limits range of movements and relations available.

Unlike with relative links, signal usage is lower with topological links. Only 1.85 % of the latter use a signal. This distinction between relative and the temporal equivalent of topological links is not made in TimeML.

This difference in signal usage rate between topological and relative links may be because topological links are used to express relations that we infer from world knowledge and do not lexicalise. In *"A Ugandan village"*, one does not need to explain that the village is in Uganda. Relative links define one region relative to another. The nature of the relation is not easy to discern and so needs to be made explicit.

Because of the dominance of spatio-temporal sense frequencies over other uses of many of the words in this class, work on temporal signals may provide insights for future researchers working on determining spatial labels using spatial signals. This chapter will later (Sect. 5.6.4.3) on show how indications of spatial signal usage help discern temporal from non-temporal candidate signal words.

5.5 Adding Missing Signal Annotations

Given an idea of what signals are and evidence of their utility in temporal relation typing, the next step was to attempt automatic signal annotation. This was a two stage process, first concerned with identifying signal expressions that occur in a temporal sense, and then with determining which pair of events/timexes any given temporal signal co-ordinates. A preliminary approach to finding temporal signal expressions found that the dataset used suffered from low annotation quality, and so after outlining the preliminary approach, this section focuses on how the resources could be (and were) improved.

Upon examination of the non-annotated instances of words that usually occur as a temporal signal (such as *after*) it became evident that TimeBank's signals are under-

annotated. In an effort to boost performance, and as there is evidence of annotation errors in the source data, we revisited the original annotations.

This chapter outlines the signal expression discrimination task only briefly, instead focusing on corpus re-annotation. The next section is dedicated entirely to the discrimination problem.

5.5.1 Preliminary Signal Discrimination

The overall problem is to find expressions in documents that occur as temporal signals (a fuller problem definition is given below, in Sect. 5.6). This was approached by considering all occurrences of expressions from the above closed class of expressions (e.g. candidate signals) and judging, for each instance, whether or not it had a temporal sense. Judgement was performed by a supervised classifier (maximum entropy), trained and evaluated using cross-validation, based on the features listed in Sect. 5.6.4.2.

Failure analysis of this initial approach suggested that the corpus was too poorly annotated to serve either as representative, solid training data for signal discrimination, or for an evaluation set for a signal discrimination approach. Some re-annotation was necessary to improve the quality of the ground truth data. This section relates the approach to, and results of, that re-annotation.

5.5.2 Clarifying Signal Annotation Guidelines

Given that the signal annotations in TimeBank are not of sufficient quality, there are three potential causes for this: annotator fatigue, insufficient annotation guidelines, or a poor definition of signals. As annotator fatigue depends on the method of an individual annotation exercise, and TimeML's signal definition is sufficient, we seek to clarify the annotation guidelines.

To clarify the guidelines, it's important to have a thorough definition of temporal signals. While TimeML's definition is sufficient, this chapter offers an extended definition of temporal signals in Sect. 5.2.

Signal surface forms have a compound structure of a **head** and an optional **qualifier**. The head describes the general action of the signal phrase and may optionally have an attached modifying phrase. Only the head should be annotated.

Example 13 "I arrived long after the party had finished."

In Example 13, the word *after* is annotated, and the qualifier *long* is not. This would be annotated in TimeML something like:

```
I arrived long <SIGNAL>after</SIGNAL> the party had
finished.
```

Further, a temporal signal has two arguments, which are timexes or events which are temporally related. Often both of these are explicit in the text immediately surrounding the signal. However, one may be elsewhere, as an implied argument.

5.5.3 Curation Procedure

The goal is to create a firm ground truth for further investigation. Given the extended definition of a signal and the guideline clarifications just mentioned, this section details the ensuing exercise of hand-curating TimeBank to repair signal annotations.

A subset of signal words was selected for re-annotation. All instances of these words (both as temporal and non-temporal) were re-annotated with TimeML, adding `EVENT`s, `TIMEX3`s and `SIGNAL`s where necessary to create a signalled `TLINK`. We will reference this version of TimeBank with curated signal annotations as **TB-sig**.

Evaluating correct classifications against erroneous reference data will lead to artificially decreased performance. To verify that the training data (which is also evaluation data for cross-validation) is from a correct annotation, negative examples of signal words were checked manually. False negatives are removed by annotating them as TimeML signals, associating them with the appropriate TLINK or adding TLINKs and EVENTs where necessary.

Checking the entire corpus would be an exhaustive exercise. To increase the chance of finding missing annotations while limiting the search space during annotation, potentially high-impact signal words were prioritised. These were drawn from a set of signal phrases that fit the following criteria: (a) more than 10 instances in the corpus, and at least one of: (b) accuracy on positive examples less than 50 % or (c) accuracy on negative examples less than 50 % or (d) below-baseline classification performance. The data from this second pass is in Table 5.13.

5.5.4 Signal Re-Annotation Observations

During curation, some observations were made regarding specific signal expressions. In some cases, these observations led to the suggestion of a feature that may help discriminate temporal and non-temporal uses of a certain expression. This section reports those observations.

Previously

TimeBank contains eight instances of the word *previously* that were not annotated as a signal. Of these, all were being used as temporal signals. The word only takes one event or time as its direct argument, which is placed temporally before an event or time that is in focus. For example:

"X reported a third-quarter loss, citing a previously announced capital restructuring program"

Table 5.13 Signal texts that are hard to discriminate; error reduction performance compared to the most common class ("change") is based on a maximum entropy classifier, trained on TimeBank. tp/fn/fp/tn correspond to counts of true and false positives and negatives

Signal	Count	As sig. (%)	Acc. (%)	Change (%)	tp	fn	fp	tn	+ve acc. (%)
for	621	8.2	92.4	8	18	33	14	556	35.3
by	356	5.6	95.2	15	7	13	4	332	35.0
while	39	23.1	79.5	11	1	8	0	30	11.1
from	366	5.2	94.8	0	2	17	2	345	10.5
when	62	85.5	85.5	0	53	0	9	0	100.0
still	35	11.4	88.6	0	0	4	0	31	0.0
already	32	40.6	56.2	−8	1	12	2	17	7.7
at	311	4.8	94.9	−7	2	13	3	293	13.3
as	271	6.6	93.0	−6	3	15	4	249	16.7
over	59	22.0	71.2	−31	7	6	11	35	53.8
since	31	58.1	48.4	−23	12	6	10	3	66.7
then	23	21.7	73.9	−20	0	5	1	17	0.0
earlier	50	12.0	86.0	−17	0	6	1	43	0.0
before	33	93.9	87.9	−100	29	2	2	0	93.5
previously	19	84.2	68.4	−100	13	3	3	0	81.2
former	16	75.0	50.0	−100	5	7	1	3	41.7

In this sentence, the second argument of *previously* is *"announced"*, which is temporally situated before its first argument (*"reported"*). When *previously* occurs at the top of a paragraph, the temporal element that has focus is either document creation time or, if one has been specified in previous discourse, the time currently in focus.

After

Of the nineteen instances of this word not annotated as temporal, only three were actually non-temporal. The cases that were non-temporal were a different sense of the word. The temporal signals are adverbial, with a temporal function. Two non-temporal cases used a positional sense. The last case was in a phrasal verb *to go after*; *"whether we would go after attorney's fees"*.

Throughout

All the cases of *throughout* not marked as signals were not temporal signals. Four were found in the newswire header, which carries meta-information in a controlled language heavily laden with acronyms and jargon and is not prose.

Early

Three of the negative instances of *early* are possibly not correctly annotated; the other 32 negatives are accurate. Of these three, one has a signal use, in part of a longer signal phrase *"as early as"*. The remaining two cases look like temporal signals. However, they are adjectival and only take one argument; there is no comparison, so we cannot say that the argument event is earlier than anything else. For this reason, they are deemed correctly annotated as non-signals.

When

There are 35 annotated and 27 non-annotated occurrences of this phrase. It indicates either an overlap between intervals, or a point relation that matches an interval's start. Twenty-three of the twenty-seven non-annotated occurrences are used as temporal signals. Two of the remaining four are in negated phrases and not used to link an interval pair. for example, *"did not say when the reported attempt occurred"*. The other two are used in context setting phrases, e.g. *"we think he is someone who is capable of rational judgements when it comes to power"* (where *when it comes to* occurs in the sense of *"with regard to"*), which are not temporal in nature.

While

The cases of *while* that have not been annotated as a signal – the majority class, 33 to 6 – are often used in a contrastive sense. This does suggest that the connected events have some overlap, often between statives. For example, *"But while the two Slavic neighbours see themselves as natural partners, their relations since the breakup of the Soviet Union have been bedeviled"*. As two states described in the same sentences are likely to temporally overlap and any events or times outside or bounding these states will be related to the state, it is unlikely that any contribution to `TLINK` annotation would be made by linking the two states with a "roughly simultaneous" relation; the closest suitable label is TempEval's OVERLAP relation [20].

Example 14 "nor can the government easily back down on promised protection for a privatized company while it proceeds with …"

The cases of *while* that were not of this sense were easier to annotate. Sometimes it was used as a temporal expression; *"for a while"*. Other times, it was not used in a contrastive sense, but instead modal – see Example 14. The four cases of non-contrastive usage were annotated as temporal signals.

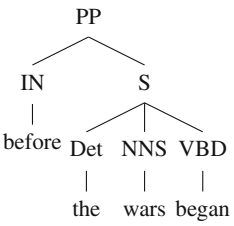

Fig. 5.3 An example of the common syntactic surroundings of a *before* signal

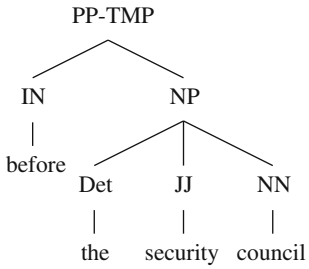

Fig. 5.4 Typical mis-interpretation of a spatial (e.g. non-temporal) usage of *before*. The whole sentence was: *"The procedures are due to go before the Security Council next week."*

Before

Three of the ten negative examples are correctly annotated. They are *before* in the spatial sense of "in front of" (as in *"The procedures are to go before the Security Council next week"*) and also a logical before that does not link instantiated or specific events (*"before taxes"*). The remaining seven unannotated examples of the word are all temporal signals. These directly precede either an NP describing a nominalised event, or directly precede a subordinate clause (e.g. `(IN before, S)` – see Fig. 5.3).

Both cases of *before* that were not temporal signals were parsed and function tagged as if they were.[5] They were given the structure `(PP-TMP, (IN before) ...)` as shown in Fig. 5.4.

Until

All fourteen non-annotated instances of *until* should have been annotated as temporal signals. This word suggests a TimeML IBEFORE relation, unless qualified otherwise by something like "not until" or "at least until".

Already

There were thirteen positive examples of *already*. All of the non-annotated examples had a non-temporal sense as per our description of temporal signals. The word tends to be used for emphasis, but can also suggest a broad "BEFORE DCT" position, which goes without saying for any past and present tensed events. As *already* can be removed without changing the temporal links present in a sentence, no further examples of this were annotated beyond the thirteen present in TimeBank.

Meanwhile

This word tends to refer to a reference or event time introduced earlier in discourse, often from the same sentence. As well as a temporal sense, it can have a contrastive "despite"-like meaning. It is often used to link state-class events, which are difficult to link unless one of their bounds is specific (see Example 15). In this case, it is

[5] Using the PTB trained Stanford Parser and the Blaheta function tagger; see Sect. 5.6.3.1.

not possible to describe the nature of the relation between the start and endpoints of either event interval, and so *meanwhile* suggests some kind of temporal overlap but nothing more. Sometimes *meanwhile* is used with no previous temporal reference. In these cases, the implicit argument is DCT. Five of the ten non-annotated *meanwhile*s were temporal signals.

Example 15 Obama was president. Meanwhile, I was a musician.

Again

This word shows recurrence and is always used for this purpose where it occurs in TimeBank not annotated as a temporal signal. No instances of *"again"* were annotated.

Former

This word indicates a state that persisted before DCT or current speech time and has now finished. Generally the construction that is found is an NP, which contains an optional determiner, followed by *former* and then a substituent NP which may be annotated as an `EVENT` of class `STATE`. This configuration suggests a `TLINK` that places the event BEFORE the state's utterance.

Example 16 *"The San Francisco sewage plant was named in honour of former President Bush."*

In Example 16, there is a STATE-class event – *President* – that at one time has applied to the named entity *Bush*. The signal expression *former* indicates that this state terminated BEFORE the time of the sentence's utterance.

Three-quarters of the non-annotated instances of *former* in TimeBank are temporal signals. An example non-temporal occurrence is shown in Fig. 5.5

Recently

Although *recently* is a temporal adverb, it cannot be applied to posterior-tensed verbs (using Reichenbach's tense nomenclature [21]). In the corpus, these are only seen in reported speech or of verbal events that happened before DCT. *Recently* adds a qualitative distance between event and utterance time, but is of reduced use when we can already use tense information.

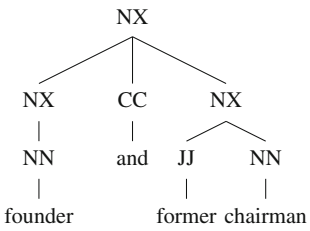

Fig. 5.5 Example of a non-annotated signal (*former*) from TimeBank's `wsj_0778.tml`

The phrase "Until recently" appears awkward when cast as a temporal signal but can be interpreted as "BEFORE DCT", with the interval's endpoint being close to DCT. In this case, recently functions as a temporal expression, not a signal.

Only one of the non-annotated *recently*s in TimeBank is a temporal signal. The exception, *"More recently"*, includes a comparative and is annotated as a `TIMEX3`; both this phrase and, e.g., *"less recently"* suggest a relation to a previously-mentioned (and in-focus) past event. As a result, we posit that *recently* on its own behaves as an abstract temporal point best annotated as a timex (as seen in the behaviour of *"until recently"* – *until* is the signal here, *recently* a TIMEX3 of value `PAST_REF`). Structures such as *[comparative] recently* may be interpreted as a qualified temporal signal, as they convey information about the relative ordering of the event that they dominate vent compared with a previously mentioned interval.

5.5.5 TB-Sig Summary

Upon examination of the non-annotated instances of words that often occur as a temporal signal (such as *after*) it became evident that TimeBank's signals are under-annotated. As we are certain of some annotation errors in the source data, we revisited the original annotations. A subset of signal words was selected for re-annotation. This set consisted of signals that were ambiguous (occurred temporally close to 50 % of the time) or that we expected, based on informal observations, would yield a number of missed temporal annotations. All temporal instances of these words were re-annotated with TimeML, adding `EVENT`s, `TIMEX3`s and `TLINK`s where necessary to create a signalled `TLINK`.

A single annotator checked the source documents and annotated 69 extra signals, as well as adding 34 events, 1 temporal expression and 48 extra temporal links. This left 712 SIGNALs that support `TLINK`s and 780 `TLINK`s that use a signal, with 54 signals being used by more than one `TLINK`. No events, timexes or signals were removed.

A summary of frequent candidate signal expressions is given in Table 5.14. The corpus is available via http://derczynski.com/sheffield/. Given this new, curated ground truth for temporal signal annotation, we are now ready to begin approach automatic signal annotation: firstly distinguishing temporal from non-temporal candidate expressions, and then linking signal expressions with the interval annotations that they co-ordinate.

Table 5.14 Frequency of candidate signal expressions in TimeBank and TB-sig. We include counts of how often these occur as signal expressions both before and after manual curation

Expression	Count in corpus	As signal	Proportion as signals (%)	After curation	Proportion (%)
in	1214	161	13.3		
after	72	56	77.8	66	91.7
for	621	52	8.4		
if	65	37	56.9		
when	62	35	56.5	56	90.3
on	344	33	9.6		
until	36	25	69.4	36	100.0
before	33	23	69.7	30	90.9
by	356	20	5.6		
from	366	19	5.2		
since	31	17	54.8	18	58.1
through	69	15	21.7		
as	271	14	5.2		
over	59	14	23.7		
already	32	13	40.6	13	40.6
ended	21	13	61.9		
during	19	13	68.4		
at	311	11	3.5		
previously	19	11	57.9	16	84.2
within	23	8	34.8		
s	10	8	80.0		
later	15	7	46.7		
earlier	50	6	12.0		
while	39	6	15.4	9	23.1
then	23	5	21.7		
once	15	5	33.3		
still	35	4	11.4		
following	15	4	26.7		
meanwhile	14	4	28.6	9	64.3
at the same time	6	4	66.7		
to	1600	3	0.2		
into	63	3	4.8		
follows	4	3	75.0		
subsequently	3	3	100.0		
followed	10	2	20.0	4	40.0
former	16	0	0.0	12	75.0

5.6 Signal Discrimination

The words and phrases that can act as temporal signals do not always convey a temporal relation. Some may indicate possession, or a spatial relation (see Sect. 5.4.4). If we are to automatically annotate signals, we need to develop a method for choosing which words and phrases in a discourse are temporal signals. This task, of finding temporal signal phrases, is called temporal signal **discrimination**.

This section begins with a problem definition and description of the method we adopted to address the problem. An automatic signal discrimination technique is trained using TimeML annotations. Finally, we present results showing automatic accuracy near or above gold-standard corpus IAA.

5.6.1 Problem Definition

The temporal signal discrimination problem is as follows: Given a closed class of signal words or phrases and a discourse annotated with times and events, identify the temporal signals. This task resembles word sense disambiguation [22, 23], in that given a word or phrase that may have multiple senses and its context, we have to determine if the active sense in context is a temporal one.

5.6.2 Method

The approach taken to automatic temporal signal discrimination is a supervised learning one.

We agreed a corpus and a set of words that could occur as signals. Next, we determined a set of feature variables that describe a word in context. After this we described each occurrence of a potential signal phrase in the corpus as a feature vector. Each instance was assigned a binary classification: positive if it is TimeML-annotated as a signal that is associated with a `TLINK`, or negative otherwise. Finally, we trained a classifier with these instances and evaluated its performance.

5.6.3 Discrimination Feature Extraction

As well as surface features from TimeML, syntactic features were used as part of feature extraction for signal discrimination.

5.6.3.1 Parsing and Other Syntactic Annotation

Syntactic information is likely to be of use in the signal discrimination task. Lapata [24] had some measure of success at learning a temporal relation classifier using sentences that contained signals, with syntactic information as a core part of their feature set. Their work used the BLLIP corpus,[6] which contains around 30 million words from Wall Street Journal articles and constituent parses generated by the Charniak parser [25].

To attempt to partially replicate this source information, we parsed the text of the TimeBank corpus. Note that TB-sig and TimeBank differ only in the annotations that they make over text; the actual words in both corpora are the same, and in the same order. To do this, we removed markup from each document and separated the remaining discourse into sentences using the Punkt sentence tokeniser [26], as part of CAVaT preprocessing [11]. Each sentence was then word-tokenised using NLTK's treebank tokeniser.[7] To maintain word alignment consistency with the non-parsed text stored in CAVaT, we needed a parser that accepted external tokenisation. We chose the Stanford parser [27] for generation of constituent parses.

In addition to constituent parses, the BLLIP corpus includes **function tags**. These are optional labels [28] attached to nodes in a constituent tree. Function tags extend a constituent tag by providing additional information about the role it plays in a sentence. They exist in three main groups; syntactic, semantic and topical [29]. Of direct interest to us is the `-TMP` tag, which indicates temporal function. An example of this tag is given in Fig. 5.6, where the first children of an `SBAR-TMP` node comprise a temporal signal.

Early work on function tag assignment in conjunction with the Charniak parser was performed by Blaheta and Charniak [30]. Their approach found that choosing whether or not to assign any tag was a significant and difficult component of the task. Thus, evaluations are split into "with-null" and "no-null" figures, where with-null refers to tag assignment accuracy including the assignment of no tag to untagged constituents and no-null is the proportion of correctly-tagged constituents excluding non-tagged nodes. We refer to no-null performance figures when discussing taggers. The initial Blaheta tagger had an F-measure of 67.8 % on the semantic form/function category, which includes the `TMP` tag.

We would like to use a function tagger with good `TMP` tagging performance. This involved selecting the right tagger. Of these, Musillo [31] simultaneously parsed and tagged text using a Simple Synchrony Parser and an extended tag set. This generated lower results than Blaheta's original attempt though this was improved to provide a marginal increase using input sentences annotated by an SVM tagger. Blaheta's final tagger [32] improved semantic tagging to 83.4 % F-measure, which was comparable to later work in which overall tagging performance increased [33, 34]. As the final Blaheta tagger is freely available and openly distributed, we used this to augment our constituency parser (the Stanford parser [27]).

[6] LDC catalogue number LDC2000T43.

[7] See http://www.nltk.org/ for more information on this package.

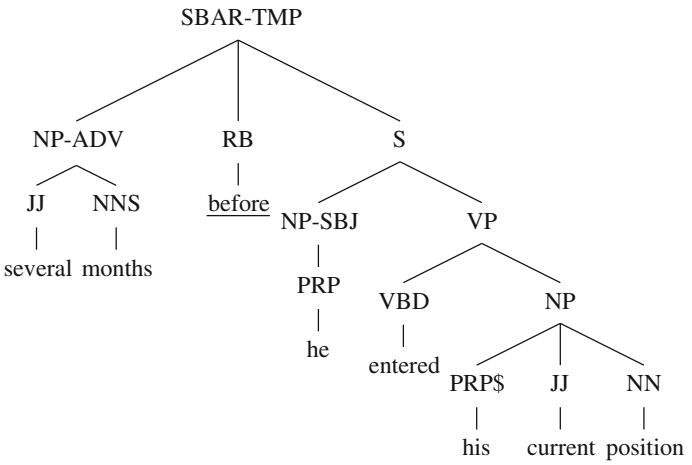

Fig. 5.6 Example of an SBAR-TMP where the first child is a signal qualifier (*several months*) and the second child the signal word itself (*before*)

We only treated as positive examples signals that were associated with a `TLINK`. Signals that only provided information regarding event cardinality, or to subordinate or aspectual links, were ignored. Signals with text not in our closed class of signal words and phrases were ignored.

5.6.3.2 Basic Feature Set

Our initial features were both syntactic and lexical; a list of them is given below. Lexical and TimeML-based features were extracted directly from a CAVaT database constructed from TimeBank [11]. We use NLTK's built-in Maximum Entropy classifier.

a. Part-of-speech from PTB tagset [35]. (`sig_pos`)
b. Function tag from Blaheta tagger; if there is more than one and the set includes TMP, assign TMP, otherwise assign the first listed. (`sig_ftag`)
c. Constituent label and function tag of parent node in parse tree (two features). (`parent_pos, parent_ftag`)
d. Constituent label and function tag of grandparent node in parse tree (two features). (`gparent_pos, gparent_ftag`)
e. Is there any node with the TMP function tag between this token and the parse tree root? (`tmplabel_in_path`)
f. Signal text. (`text`)
g. Text of next token in sentence (if there is one). (`next_token`)
h. Text of previous token in sentence (if there is one). (`previous_token`)
i. Is there a `TIMEX3` in the *n* following tokens? (`timex_in_n_after`)

j. Is there an EVENT in the *n* following tokens? (event_in_n_after)
k. Is there a TIMEX3 in the *n* preceding tokens? (timex_in_n_before)
l. Is there an EVENT in the *n* preceding tokens? (event_in_n_before)
m. The Stanford dependency relation of the candidate word to its parent. ()

In our work, $n = 2$ for the interval proximity features, based on an informed guess after looking at the data. The optimal value, depending on direction of context and type of interval (event vs. timex) search for, is left to future work.

There are 102 entries in our closed class of signal words/phrases; this set is kept constant throughout all experiments. In TimeBank there are 7 014 mentions of the members of this set, including both temporal and non-temporal mentions.

5.6.3.3 Extended Feature Set

Curation of signals, as detailed in Sect. 5.5, led to some direct observations about specific signal words. These observations in some cases suggested specific sources of signal discrimination information thar could potentially be translated to features. From the observations above, the new features that could be added were:

n. Flag to see if signal text is in a verb group (*before, after*) (in_verb_group)
o. Flag to see if a token at the top of a paragraph (*previously*)
p. Flags to see if the preceding or following word(s) are part of a verb group (*after*) (following / preceding_in_verb_group)
q. What is the highest-level subtree that begins at the next token (*before*) (following_subtree)
r. What is the highest-level subtree that ends at the preceding token (preceding_subtree)
s. PoS of the next token and previous token (*before*, *after*) (following/preceding_pos)
t. PoS of the next event within *n* tokens (*before*, *former*) (next_event_pos)
u. Type (TimeML class) of the next event within *n* tokens (*former*, *meanwhile*) (next_event_class)
v. TimeML Tense and aspect of the next event within n tokens (*already*) (next_event_tense / aspect)
w. NP begins at next token? (*former*) (np_next)
x. Is the preceding token a comparative, i.e., is it one of JJR or RBR? (*recently*) (preceding_comparative)

All of these were implemented and added as features, except the paragraph-top feature (due to a lack of a reliable document segmentation tool). In addition, we removed some noisy features that seemed to be causing overfitting within our sparse data set; the offset of the word within its sentence and the preceding & following token texts. We used the full constituent tag of subtrees for the preceding_subtree and following_subtree features, including.

Table 5.15 Comparison of the effect that decomposing values of the preceding_subtree and following_subtree features has, using our extended feature set and TimeBank data. Error reduction compared to classifier MCC baseline

Features	NBayes	MaxEnt	ID3
Full subtree labels	−1.32	19.4	**25.4**
Just constituent tag	−2.31	19.7	**21.6**
Separate constituent and function tag	−4.28	19.9	**24.2**

5.6.3.4 Multivalent Tags

In a minority of cases, constituents and terminals were assigned multiple function tags. For example, values such as PRD-TPC-NOM or TMP-SBJ would be appended. Noticing that these instances were assigned high weights by a Naïve Bayes classifier, we measured error reduction on multiple variations of subtree tag feature representations. Results are shown in Table 5.15. It was found that reducing data sparsity by providing two separate features per subtree (for constituent tag and function tag) provided best overall performance for MaxEnt discriminators, but ID3 benefited most from the feature extraction that gave the sparsest values – full subtree labels.

5.6.3.5 Choice of Learning Algorithm

Signal discrimination is a binary classification problem: is a given word or phrase a temporal signal or not? We have constrained the set of words we attempt to classify by defining a closed class of signal words and described a set of features with which we will represent candidate words and context. We now need to choose a binary classification algorithm. We use a Naïve Bayes classifier, decision trees, a maximum-entropy classifier and adaptive boosting.

For rapid learning and quick feedback, we worked with the Naïve Bayes classifier. Naïve Bayes models are computationally cheap to learn. Its inductive bias includes the independence assumption – that all features are independent from each other. This is not true in our case, given the heavily interdependent nature of most of our features: well-formed syntactic structures are inherently constrained by grammar and the values of many of our features depend on syntax at multiple places in the same sentence or paragraph. For example, the parts of speech of any given token has some bearing on the part of speech of the following one, and these are again not independent of the parse tree of the sentence in which they occur. We also use a decision tree classifiers, which do not have this particular bias and are computationally quick to learn, but do not always cope well with noise. ID3 and C4.5 types are used. C4.5 attempts to deal with noise in training data by performing pruning on the tree after construction [36].

We also evaluate performance of our feature set with a maximum entropy classifier. This regression-based model assumes low collinearity between features, which is a less constraining assumption than that of the Naïve Bayes classifier, though problems may arise if we use highly-correlated features. Finally, we use adaptive boosting with decision stumps [37, 38], which is constrained to binary classification and can yield high-performance results. Adaptive boosting reduces the impact of the typically computationally intensive SVM-learning process and typically displays little overfitting, which is helpful with smaller datasets such as ours.

Performance was improved by removing features that have a high number of values (for example, the text of the token after a signal). We suspect this is due to them leading to overfitting.

5.6.4 Discrimination Evaluation

We have described how we trained a classifier using cross-validation. We evaluated performance using a held-out evaluation set, and determined scores by counting correct classifications and measuring both percentage of correctly classified instances and also the error-reduction compared to a baseline.

5.6.4.1 Baselines

To evaluate the performance of our approaches, it is useful to describe some simple annotation methods as baselines. A summary of our baselines is given in Table 5.16 and we explain each of them below.

One simple baseline is to find the most common classification and assign this to all instances. In our corpus, instances of phrases from our list of potential signals are used non-temporally nearly all the time (out of 6 091 instances of potential signal phrases, only 688 are annotated as being temporal signals in TimeBank – 11.3 %) and so our most common case is to classify everything as not being a temporal signal, regardless of the signal text.

We also use baselines that mark all words found in the signal phrase list as temporal signals if they have a part-of-speech tag of RB or IN, according to NLTK's built-in

Table 5.16 Performance of four constituent-tag based baselines over TimeBank

Baseline	Accuracy (%)	Accuracy on positives (%)
Most common class	86.7	0.0
Baseline: Part-of-speech is IN	25.6	**81.2**
Baseline: Part-of-speech is WRB	86.9	5.77
Baseline: Parent is SBAR-TMP	**87.0**	9.88
Baseline: Parent function is -TMP	84.5	72.7

maximum entropy tagger. Values are quoted for overall classification accuracy, as well as accuracy on positive examples (the minority of our training data).

Most Common Class

The training set is confined to just signal annotations in TimeBank/TB-sig, that are also in the closed class of signal expressions detailed above in Table 5.9. This introduces an inherent performance cap to the overall approach, but assumes no knowledge of whichever corpus is being used as the evaluation set. Of 4 576 training instances, 3 969 are negative (non-temporal) and 607 are positive (having a temporal meaning). The most-common-class is negative and if we assign this label to all mentions of members of the set, classifier accuracy is 86.7 % but no signals are identified (giving an effective F1 of zero if we imagine this as a signal recognition task); not a very informative baseline.

Class Member and Signal Word Tag

Of all leaf labels, IN and WRB have the highest proportion of signals (Table 5.10). To this end, we have two simple baselines, where we count a word as a temporal signal if its constituent tag is IN or WRB and it is found in the closed class of signals. Performance for these is given in Table 5.16. For IN, we have 25.6 % overall accuracy, correctly identifying text that is a temporal signal 81.2 % of the time. For WRB, we achieve 86.9 % accuracy, but only 5.77 % on the positive examples.

Parent Is SBAR-TMP

As mentioned in Sect. 5.4.2.1, one might expect an a SBAR-TMP subtree to begin with a temporal signal and also contain one of the signal's arguments (see also Fig. 5.6). As we can use our closed class of signal words to differentiate signal head, signal qualifier and event/timex argument, we can look for leaves where the parent is SBAR with TMP in its function tags. This is our SBAR-TMP baseline, that performs at 87.0 % accuracy overall, with 9.88 % on positives – better than WRB, but still poor.

Parent Has Temporal Function

Limiting ourselves to just signals in subtrees labelled SBAR may be a short-sighted manoeuvre. We added a baseline that labels signal candidates as temporal if their parent has a temporal function label. This baseline achieves classification accuracy of 84.5 % and a 72.7 % accuracy on the positive examples; see Table 5.16.

Table 5.17 Signal discrimination performance on the plain TimeBank corpus. Error reduction is measured relative to the "parent has temporal function" baseline. Evaluated with 5-fold cross validation and 1 000 iterations of adaptive boosting

Measure	Accuracy	Accuracy (+ve)	Error reduction	Error reduction (+ve)
Naïve Bayes	88.6	**78.4**	26.5	**20.9**
Maximum Entropy	89.5	56.0	32.3	−61.2
ID3	90.5	65.6	38.7	−26.0
C4.5	90.4	60.1	38.1	−46.2
AdaBoost	**90.7**	59.8	**40.0**	−47.3

5.6.4.2 Performance

With our original feature set and based on pre-curation data (e.g. TimeBank v1.2), we achieved a 40 % error reduction in signal discrimination relative to a competitive baseline, as seen in Table 5.17. For the general annotation task, naïve Bayes performed best, with good error reduction overall (26.5 %) and a similar improvement in recognition of positive examples (20.9 %), something that other classifiers did not perform so well with.

With the original feature set, models learned over TB-sig data performed as shown in Table 5.18. Performance using the extended feature set is detailed in Table 5.19, again based on TB-sig.

Our extra annotations introduce new signal instances for the extra terms that we have annotated, reducing the baseline to 85.2 % accuracy (677 positives, compared to 607 before re-annotation) from 86.7 % before – see Table 5.18. Performance using TB-sig is overall better (compared to Table 5.17), which we attribute to having a better-stated problem and less misleading data. Error reduction rate is now over 40 %, with overall accuracy just under 92 % and up to 75 % on the positive examples. This is better than performance on the original TimeBank data and comparable to the IAA figure of 0.77 for TimeBank's initial `SIGNAL` annotation. C4.5 performs particularly well, reaching near-highest error reduction rate and good accuracy on positive examples.

The extended feature set, however, does not improve performance in the majority of cases, despite having been generated as part of a rational investigation. Analysis and further work is required to improve upon these signal discrimination results.

Table 5.18 Signal discrimination performance on the curated corpus. Error reduction is measured relative to performance. Results are for 5-fold cross validation. Adaptive boosting used 1 000 iterations

Measure	Accuracy	Acc. (+ve)	Error reduc.	Error reduc. (+ve)
Most common class	85.2	0	n/a	n/a
Baseline: IN	25.4	77.1	–	–
Baseline: RB	86.3	8.3	–	–
Baseline: SBAR-TMP	86.1	10.8	–	–
Baseline: Temporal parent	84.5	70.0	–	–
Simple features				
Naïve Bayes	89.3	**78.7**	31.0	29.0
Maximum Entropy	88.2	51.3	23.9	−62.3
ID3	91.7	69.6	46.5	−1.3
C4.5	**92.1**	73.0	49.0	10.0
AdaBoost	91.9	70.5	47.7	1.7
Extended features				
Naïve Bayes	87.0	**81.4**	16.1	38.0
Maximum Entropy	88.1	50.1	23.2	−66.3
ID3	91.1	68.7	42.6	−4.3
C4.5	91.7	75.0	46.5	16.7
AdaBoost	**91.8**	69.3	47.1	−2.3

Table 5.19 Signal discrimination performance on the TimeBank corpus, with an extended feature set. Error reduction is measured relative to most-common-class ("not a signal") performance. Evaluated with 5-fold cross validation and 1 000 iterations of adaptive boosting

Measure	Accuracy	Accuracy (+ve)	Error reduction
Extended features			
Naïve Bayes	86.1	80.9	−4.28
Maximum Entropy	89.4	55.4	19.9
ID3	89.9	59.8	24.2
C4.5	90.4	60.8	27.8
AdaBoost	**90.6**	**59.3**	**29.3**

5.6.4.3 Useful Features

A sample post-classification analysis of feature weights – using TB-sig and the extended feature set – is presented in Table 5.20, taken from the last of five cross-validation passes. This is from the construction of a model using the whole signal-labelled corpus with a naïve Bayes classifier. The text of the signal is a particularly strong indicator for some of the features that occur much more often as temporal signals than not. We can also see that wh-adverb signals and wh-adverb phrases that contain the candidate signal expression are strong indicators of temporal meanings (features signal_label, parent_label and ending_subtree_label); this may be because of words such as *when* having only temporal senses. A timex or a past-tensed event occurring after the signal is also an indicator of it being temporal (timex_in_2_after). When the parent constituent or the largest constituent beginning at this point has a temporal function, then a candidate word is more likely to be temporal (parent_function, starting_subtree_function). The -TMP function tag helps to indicate a temporal signal when it dominates the candidate signal word (tmpfunction_in_path). Being followed by a dollar amount suggests that a candidate is not temporal (following_label = $) – for example, in a non-temporal use, *"Shares closed at $ 50"*; the high weight of this attribute-value pair is likely influenced by the high proportion of financial reporting in TimeBank, which takes a significant part of its text from the Wall Street Journal.

Words and phrases that are within a syntactical structure that has a spatial function (e.g. -LOC) contra-indicate a temporal meaning. This is aligned with the observation that members of our class of signal words often have both temporal and spatial meanings. Further, an adjacent structure with a spatial function (-EXT or -LOC) suggests a temporal function in a candidate word. This suggests collocation based approaches may not correctly discriminate temporal and non-temporal signals; syntactic parsing is required, in order to detect these functional nuances. Having NX (indicating the head of a complex NP) as a parent at can indicate a signal; this could be in cases where we have a signal before a nominalised event, such as in *"before the explosion"*. Finally, preceding a verb may be an indication of a temporal signal; this reflects the signal's adverbial nature.

5.6.5 Discrimination on Unseen Data

Up to this point, evaluation has used cross-validation over TimeBank. Our error analysis led to the inclusion of features based on the data that is also part of the evaluation set. To check performance on previously unseen data, a further experiment was performed is as follows. We trained a signal discriminator and associator based on all of TimeBank + the extra signal annotations. The closed class is increased to include all phrases marked as signals in TimeBank. This way, TimeBank is only the training data.

Table 5.20 Sample features useful for signal discrimination, based on our curated TimeBank data, TB-sig

Feature	Value	Indication	Weight
text	until	True	131.5
text	before	True	70.0
text	after	True	56.9
signal_label	WRB	True	49.6
parent_label	WHADVP	True	49.5
ending_subtree_label	WRB	True	48.5
text	when	True	48.3
text	previously	True	26.2
text	former	True	15.4
grandparent_label	SBAR	True	13.9
text	during	True	11.5
following_subtree_function	-LGS	False	9.7
text	meanwhile	True	9.6
timex_in_2_after	True	True	9.0
text	since	True	7.6
preceding_subtree_label	S	True	7.2
starting_subtree_function	-LOC	False	7.1
following_label	$	False	7.0
starting_subtree_label	SBAR	True	6.6
parent_function	-LOC	False	6.4
following_subtree_label	VBN	True	6.3
starting_subtree_function	-TMP	True	6.2
following_label	PRP	True	6.1
grandparent_label	NX	True	5.7
starting_subtree_label	NX	True	5.7
preceding_label	JJS	True	5.6
following_subtree_label	VB	True	5.6
text	thereafter	True	5.6
next_event_tense	PAST	True	5.4
parent_function	-TMP	True	5.3
parent_label	SBAR	True	5.3
text	later	True	4.9
tmpfunction_in_path	True	True	4.1
preceding_subtree_function	-LGS	True	4.1
preceding_subtree_function	-EXT	True	4.1
following_subtree_function	-PRD	True	4.1
starting_subtree_function	-TPC	True	4.1
grandparent_label	SINV	True	4.1
following_subtree_label	.	False	4.0
following_label	.	False	4.0

Table 5.21 Characteristics of the N45 section of the AQUAINT TimeML corpus, before and after signal curation

Feature	Pre-curation	Post-curation
Documents	15	
Tokens	7099	
Signals	96	114
TLINKs	1048	1062
Events	1060	1060
Timexes	154	156

Table 5.22 Performance of a TB-sig trained signal discriminator on unseen data

Method	Accuracy (%)	Precision (%)	Recall/acc. on positives (%)
Parent –TMP baseline	84.5	–	70.0
MaxEnt model	93.6	83.0	78.3

As the final model was developed based partially on observations of TimeBank, it is not suitable to evaluate the final model on this corpus also. A previously unseen set, taken from the AQUAINT corpus (Sect. A.2.2), now forms the evaluation set. The N45 section of the AQUAINT corpus was curated to verify its signal annotations, and then signal discrimination was evaluated over this subcorpus based on a model trained on the entirety of TB-sig. The relevant statistics regarding this evaluation corpus are presented in Table 5.21.

Signal discrimination is measured in two ways. Firstly, classification accuracy shows how many of the candidate signal words were correctly labelled as signals or not-signals. Secondly, the overall performance of the association approach at annotating signals in any given document is described in terms of precision and recall. This takes into account how well the entire approach described above (including the signal words list described in Table 5.9, but not also including those found in TimeBank) does when given the task of identifying temporal signals in an arbitrary text. The augmented AQ/N45 annotations form the gold standard. The "parent has temporal function" baseline (Sect. 5.6.4.1) is used for comparison. Results are presented in Table 5.22. This compares well with the performance on (seen) TB-sig data (Table 5.17).

5.6.6 Summary

In this section, we have explored the task of signal discrimination. We discovered that TimeBank's signal annotations are incomplete. To remedy this, we have proposed augmentations to the TimeML annotation standards and re-annotated a portion of the

corpus. We have also defined a set of features that can describe a temporal signal in context and constrained our search space to just words and phrases in a closed class of signal words. As a result, we have been able to train a classifier to detect temporal signals at near-IAA accuracy.

5.7 Signal Association

Temporal signals connect one or more interval pairs and describe the nature of the temporal relation between the pair. This section describes an investigation into how to find the arguments of a temporal signal, thus associating the two arguments. We refer to this task as **signal association**.

In order to fully annotate temporal signals, we need to determine which arguments they co-ordinate. To this end, the task of determining which times or events are coordinated by a temporal signal is examined as the subject of this section.

5.7.1 Problem Definition

When performing temporal annotation, one needs to identify events and times and can then connect them with temporal links, perhaps using an associated signal. In fact, every time that a temporal signal is annotated, there must be a temporal link present. The signal association problem is: Given text with signal, event and timex annotations, determine which pair of events/times are associated by each signal phrase.

5.7.2 Method

A supervised learning approach is taken to finding which intervals a given signal co-ordinates. TB-Sig is used as the dataset for feature extraction. Two approaches are explored, detailed below. These use a largely common feature set, extracting a number of features for each interval considered and a further set of features describing the signal.

To generate training data given a signal, we will describe events and timexes within the scope of that signal using our feature set. Although any two intervals in a document could be linked by a given signal, the number of intervals or interval pairings one must search through could be large if the entire document is used as potential signal scope. For this reason, scope must be constrained, at a possible performance loss. Given candid examination of the signals in the corpus, the scope of the signal is taken to be the signal's sentence and also enough previous sentences

to include at least two intervals, as well as a DCT timex if present. We are attempting to determine which intervals are associated with the signal.

The goal is to learn a binary function, that can indicate whether or not an association supporting a TLINK exists in a given situation. A TLINK associates two intervals (timex or event) and may specify the type of temporal relation between them. We have tried two approaches to this signal association task; one where we examine ⟨interval, signal⟩ tuples and another where we examine ⟨interval-pair, signal⟩ tuples. The gold standard corpus, TimeBank, provides the positive examples. For each signal, there may be up to five valid TLINKs, each shown as an interval pair (see earlier Table 5.8).

For the single interval approach, we train a binary classifier to learn if an interval and signal are linked and then choose the two best candidate intervals for a signal, using classifier confidence to rank similarly-classified intervals. For the interval pair approach, for each signal we examine possible combinations of intervals and create a vector of features based on relations between the intervals and the given signal.

5.7.2.1 Single Interval Approach

In this section, we describe a signal association approach where individual intervals are ranked by their relation to the signal and the top two intervals are deemed to be associated.

Positive training examples came from intervals associated in a gold standard annotation. Negative training examples were taken to be all temporal intervals in the same sentence as the signal that were not associated with the signal. We used cross-validation to learn classifiers and recorded the prediction and confidence of the classifier for each entry in the evaluation fold. After this, for each signal, a list of candidate intervals was determined. The two intervals related to the signal were those classified as related with highest classifier confidence, or if fewer than two positive classifications were made, up to two are taken from lowest-confidence unrelated classifications. That is, for each signal, intervals are ranked in descending order of confidence; the goal is to find the two most likely intervals, and associate them in a TLINK backed by the given signal. Priority is established in this order:

1. High-confidence and classified as related
2. Low-confidence and classified as related
3. Low-confidence and classified as unrelated
4. High-confidence and classified as unrelated

The top two are then associated with a signal. This approach is limited to only detect one pair of intervals per signal.

5.7.2.2 Interval Pair Approach

In contrast to our previous approach, we tried to identify whole ⟨interval-pair, signal⟩ 3-tuples as either a signalled `TLINK` or not. This produced a majority of negative examples. We instead only considered intervals where both arguments fell inside a sliding window of sentences, to reduce the heavy skew in training data. A boolean feature describing whether the intervals were in the same sentence was added to our set, as well as two sets of interval-signal relation features and general signal features as described earlier.

5.7.2.3 Surface and Constituent-Parse Features

For the signal association tasks, we used the following surface and constituent-parse features as input to a binary classifier. Constituent parse information comes from running the Stanford Parser [27] over discourse sentences, the bounds of which are determined using the Punkt tokeniser [26] implementation in NLTK. The features describe a single interval/signal pair. We use the same definition of syntactic dominance as [24]; that is, an interval (e.g. event or timex) is syntactically dominated by a signal if the interval's annotated lexicalisation is found within a parse subtree where the first (leftmost) word of the parse subtree is the signal. Dominance features are included based on their success in signal linking in [24], where dominance was described as the V_L feature.

- Is this interval the textually nearest after the signal?
- Is this interval the textually nearest before the signal?
- Does the signal syntactically dominate the interval?
- Signal text (lower case)
- Signal part of speech
- Token distance of interval from signal
- Interval/signal textual order
- Is there a comma between the interval and signal?
- Is the interval in the same sentence as the signal?
- Is the interval DCT or a DCT reference?
- Interval type (TimeML `EVENT` class or `TIMEX3` type), total 11 values
- If an event, its TimeML-annotated tense

5.7.2.4 Dependency Parse Features

We use the Stanford dependency parser [39] to return dependency graphs of our PoS-tagged, parsed and function labelled sentences. By default, the dependency parser ignores some words that we consider to be signal words, moving information about removed words in relationships. We configured it to never ignore words. The features that we extracted from sentence dependency parses were:

- Length of path from interval to root
- Is the signal a child of the interval?
- Is the signal a direct parent of the interval?
- Is the interval the tree root? (e.g., the head event/time)
- Is the interval directly related to the signal with an `advmod` or `advcl` relation?
- Does the interval modify the root directly? (e.g., is the interval a direct ancestor of the root, regardless of relation type)
- Does the signal modify the interval directly? (e.g., is the signal a direct ancestor of the interval)
- What relation does the interval have to its parent?
- If the signal is a child of the interval, what is the relationship type?

5.7.3 Dataset

Examining some of the instances of temporal relations in TimeBank which have an attached signal, there were often clear syntactic relations between signals and their arguments (which are also the temporal relation's arguments). Almost all signals co-ordinated two intervals in the same sentence as the signal (Table 5.23). In the cases where they did not, one of three situations prevailed. Firstly, the signal was the first token in the sentence and the argument outside of the sentence was either referenced by a temporal pronoun (as in e.g. *"After **that**, the situation improved."*). Secondly, one argument is an event or time that has remained the temporal focus in discourse at the point where the signal is found, even after new sentences have been introduced. Thirdly, the signal will relate DCT with an interval in its sentence.

5.7.3.1 Closure

Some supervised approaches that deal with temporal relations chose to use closure to generate extra training data. We have deliberately chosen not to include temporal

Table 5.23 Distribution of sentence distance between intervals linked by a signal, for TB-sig. A special case is made for those that link to document creation time or one of its co-referents, as it often persists as a reference point through the length of a discourse

Distance	Count
DCT	40
0	682
1	43
2	16
3	3
4	3
5+	0

links generated through closure [40] in our examples. Temporal closure typically generates more links than were in the original annotation by at least an order of magnitude. The generated links tend to be between intervals not directly related in text – e.g. lacking textual proximity or clear discourse relations. As with many binary classification models, the negative examples that enable our classifiers to learn the most precise decision boundaries are those that closely resemble positives. Entities only linked through a chain of four or five annotated `TLINK`s, with low textual or syntactic proximity, will not be in this set. We do however use windowing approaches to permit some of these wide-ranging negative examples into the training.

5.7.3.2 Detecting Document Creation Time

Document creation time (**DCT**) refers to the instant at which a discourse was created. In the case of newswire articles this is often included in the article metadata, or as a deictic temporal expression at the beginning of the first sentence, which describes day and month (e.g. *"KABUL, August 21 – ..."*). Other times, it may be possible to extract this date automatically [41]. The document creation time persists throughout a discourse as an antecedent temporal point that may be referred to by temporal expressions or, in some cases, signals. As we have seen some signals that work like this (e.g. *afterwards*), it may be useful to include a boolean feature indicating whether or not a timex represents DCT.

TimeML-annotated data is used to determine whether a given timex is DCT or DCT-equivalent. Our algorithm is as follows, given a candidate `TIMEX3` element:

1. if `functionInDocument` = `CREATION_TIME` ⇒ return **true**
2. if `functionInDocument` = `PUBLICATION_TIME` ⇒ return **true**
3. `most-frequent-anchor` ← the most frequent non-null value of `anchor TimeID` in this document's `TIMEX3` annotations
4. if `sentence-number` $< j$ and `timex_id` = `most_frequent_anchor` ⇒ return **true**
5. else return **false**

That is, we first look for explicit annotation markers that declare this timex to be a creation time reference. Failing that, if the timex is near the beginning of the document and also the timex most-often used as an anchoring point for other timexes, we mark it as DCT-referring. With $j = 2$, this heuristic is accurate for all of TimeBank.

5.7.4 Automatic Association Evaluation

As both approaches rely on a binary classifier, the first evaluation measure given is classifier accuracy. This shows the proportion of accurate binary decisions made by the classifier based on model learned from training data. The error reduction that the

Table 5.24 Performance at the signal:interval association task, with 5-fold cross validation. The classifier performance baseline is most-common-class, which was 64.1 % not-related for TimeBank and 64.0 % not-related for the signal-augmented version

Corpus	Classifier	Accuracy	Err. reduc	Full (%)	Partial (%)	Failure (%)
TimeBank	MaxEnt	**85.2**	58.7	**64.2**	34.5	1.25
	NBayes	82.5	51.1	57.2	41.2	1.53
	ID3	78.4	39.8	42.1	52.1	5.85
TB-sig	MaxEnt	**84.8**	57.9	**61.5**	37.6	0.897
	NBayes	82.2	50.5	56.3	41.9	1.79
	ID3	79.6	43.4	40.9	54.4	4.74

classifier's model provides over a most-common-class baseline is also given. The single-interval approach and interval-pair approaches are structurally different and can be further evaluated in separate ways, which are detailed below, as well as results.

5.7.4.1 Single-Interval

We recognised three possible states of signal annotation. A **full match** occurs when both signal arguments are correctly found, when just one argument is correct we have a **partial match** and when both associated arguments are incorrect there is a **failure**. Results of classifier performance and signal annotation success can be found in Table 5.24. Full matches are the only cases we should consider as successes; anything else is not correct, though partial successes (where one argument is correctly associated) are shown to give insight into how problematic the non-full matches were. As can be seen from the data, even in cases where there was not a full argument match, it was almost always the case that at least one interval was correctly associated – that is to say, partial matches were orders of magnitude more common than failures.

5.7.4.2 Interval-Pair

Results for the interval-pair:signal approach are given in Table 5.25. The "Acc (+ve)" column represents the classifier accuracy on examples labelled as positive in the gold standard, as opposed to the proportion of the instances labelled as positive that were matched the gold standard annotations. The best classifiers are those that achieve a high error reduction while maintaining good classification accuracy on positive examples.

For most Naïve Bayes classifier results, there were was a low false negative and a high true positive rate, but also an overbearing false positive rate. For example, with $n = 2$ there were 1371 true positives and only 65 false negatives, which is good, but 4513 false positives, meaning that the classifier output was not particularly useful. Less than one quarter of interval-pair:signal associations would be accurate.

Table 5.25 Performance at the signal:interval-pair association task, with 5-fold cross validation. The baseline is most-common-class, which was "no link" in all cases. The sentence window for negative examples is the signal's sentence plus the n prior sentences

Corpus	Classifier	Accuracy	Err. reduction (%)	Acc. (+ve)
TimeBank n = 0, baseline 89.6	NBayes	94.0	41.8	91.4
	ID3	97.7	77.3	84.7
	MaxEnt	92.5	28.0	43.7
TimeBank n = 1, baseline 96.6	NBayes	93.6	−89.4	93.9
	ID3	99.3	**79.9**	84.0
	MaxEnt	97.1	13.9	43.6
TimeBank n = 2, baseline 98.3	NBayes	94.7	−219	**95.5**
	ID3	**99.4**	62.1	68.7
	MaxEnt	84.9	-804	39.3
TB-sig n = 0, baseline 89.7	NBayes	94.1	42.8	90.8
	ID3	97.4	74.8	84.8
	MaxEnt	92.2	23.6	41.6
TB-sig n = 1, baseline 96.7	NBayes	93.4	−100	93.2
	ID3	99.3	**78.0**	83.5i
	MaxEnt	97.1	12.3	44.5
TB-sig n = 2, baseline 98.4	NBayes	94.7	−229	**94.7**
	ID3	**99.1**	42.7	46.8
	MaxEnt	84.9	−832	38.8

Table 5.26 Confusion matrix for signal association performance with a MaxEnt classifier on TimeBank with a window including the signal sentence and two preceding ones

	Prediction	
Class	True	False
True	564	872
False	12,110	72,192

Table 5.26 shows the confusion matrix of the worst-performing attempt. It detects a large number of false positives.

Using windowing for candidate interval selection with $n = 2$, 0.38 % of signal arguments lie out of the window (see Table 5.27) and are therefore not correctly associable with this approach – an acceptably small amount. With $n = 0$, this unassociable proportion rises to 4.13 %. We found that increasing n led to worse classifier performance and a value of $n = 1$ provided a good trade-off.

Table 5.27 Distribution of sentence distance between intervals and signal that links them. A special case is made for those that link to document creation time or one of its co-referents, as in Table 5.23

Distance	Count
DCT	41
0	1468
1	43
2	16
3	3
4	3
5+	0

Performance is worst with $n = 2$. We can achieve a good classification accuracy on a test set that includes cross-sentence links even if we only consider same-sentence intervals for the generation of negative examples (i.e. $n = 0$). We can also see that decision trees, which do not follow the independence assumption, perform consistently well, although do worse as n increases.

5.7.4.3 Evaluating on Previously Unseen Data

To test association on its own, a classifier is trained on TB-sig and evaluated on the augmented AQ/N45 data (a TimeML subcorpus introduced in Sect. 5.6.5). The interval pair annotation method is used, as it performs best on prior TimeML data (Sect. 5.7.4.2). The results are shown in Table 5.28.

This is satisfactory performance, with a strong error reduction of 58 % beyond the baseline.

5.7.5 Association Summary

Our aim was to find a method of automatically associating a temporal signal with a pair of intervals, given a partially annotated text. We tried two approaches. The first ranked ⟨interval, signal⟩ tuples and treated the top two as linked. The second treated ⟨interval-pair, signal⟩ tuples as atomic units.

Table 5.28 Performing of a TB-sig trained signal associator on unseen data

Method	Accuracy (%)	Error reduction (%)	Acc. on positives (%)
Most common class (not related)	91.96	–	0.00
ID model (n = 1)	96.60	57.72	84.93

It is important to achieve a good error reduction rate and also to have good predictive accuracy on positive examples. Both of these metrics need to have high values for a classifier to be useful in annotation. We found that although the ranked single-interval approach achieved decent results, treating interval pairs as atomic units worked better. We achieved 78.0 % error reduction over the most-common-class baseline, at 96.7 % predictive accuracy and 83.5 % accuracy on the positive examples.

5.8 Overall Signal Annotation

The overall motivation for signal extraction is to improve automatic temporal relation typing. We have independently determined that signals are useful for TLINK typing (Sect. 5.3) and that we can extract and associate signals automatically (Sects. 5.6 and 5.7). To show that automatic extraction is useful in support of the relation typing task, we took a gold-standard TimeML corpus (the AQUAINT TimeML corpus) and removed all its signal annotations. Performance of an automatic TLINK labeller was then compared when there are no signal annotations and when signal annotations have been automatically added using the above methods.

The same unseen corpus (a signal-augmented version of the N45 section of AQUAINT TimeML corpus) was used for evaluation of discrimination and association, as introduced in Sect. 5.6.5.

5.8.1 Joint Annotation Task

To measure combined performance, the signal annotations suggested in the discrimination step are used as the basis for association. Note that because the set of TLINKs identified in a document's annotation may not be a temporal closure of that document (see Sect. 3.3.2), it is possible to correctly detect a pair of events that are in fact linked via a signal but for the TLINK not to be present in the gold standard. For this reason, the performance scores are minimums. We hypothesise that despite a lack of guidance regarding which TLINKs must be defined in order to create a complete or valid TimeML annotation, annotators are likely to add explicit TLINK annotations where the temporal relation is suggested explicitly (e.g. with a signal). Therefore the number of unannotated signalled TLINKs should be small.

The corpus used was the augmented N45 dataset, stripped of TLINK and SIGNAL annotations (leaving TIMEX3s and EVENTs). The method was to first attempt automatic signal discrimination over the corpus (training on all of TB-sig using the basic feature set), and then perform automatic signal association (using the interval-pair approach). The resulting SIGNAL and TLINK annotations were then compared to the augmented N45 annotations.

Table 5.29 Details of the joint approach to signal annotation. Although the augmented N45 corpus only contained 136 signals, our approach found 424. This table breaks down that 424

Signal/TLINK associations	Count	Proportion (%)
In N45	136	–
Found	336	–
Found, both args in N45	88	26.2
Signal in N45, new TLINK assoc	216	64.3
Found based on new signals	32	9.5

Results are summarised in Table 5.29. In total, compared to the 136 signalled TLINKs in the augmented AQ/N45 data, 336 interval pairs (e.g. TLINK suggestions) were suggested based on the automatically annotated signals. A total of 64.7 % of the 136 TLINKs were found correctly automatically. Only 26.2 % of associated interval pairs (88 out of 424) were found in the gold standard; 248 were not there. A minority of 9.5 % (32) of pairs found were based on signals not in the gold standard. This leaves 64.3 % (216) automatically generated instances of signal associations with interval pairs not mentioned in the gold standard.

Upon manual inspection, many of these false positives based on existing signals appear to be supported in the text, but are not annotated in the gold standard, which in many cases contains only a minimal annotation, and certainly never constitutes a closure. Take the following cases, for example, taken from NYT19990505.0443.tml in the signal-augmented corpus and edited slightly for brevity:

Example 17 `A jogger <EVENT eid="e64">observed</EVENT> Kopp's car <SIGNAL sid="s7">at</SIGNAL><TIMEX3 tid="t10">6a.m.</TIMEX3>near Slepian's home <TIMEX3 tid="t11">10 days</TIMEX3> <SIGNAL sid="s8">before</SIGNAL> the <EVENT eid="e65">murder</EVENT>, and, <EVENT eid="e66">curious</EVENT> why a stranger would be <EVENT eid="e67">parked</EVENT> there so early, <EVENT eid="e68">wrote</EVENT> down the license plate number.`

In this section, our approach found the links listed in Table 5.30 (in this example, event eids and instance eiids have a 1:1 mapping, so ei65 corresponds to event e65).

Table 5.30 Sample signals and arguments found in N45

Signal ID	Argument 1	textbfArgument 2	In GS?
s8	ei64	ei65	Yes
s8	ei65	ei66	No
s8	ei65	ei67	No
s8	ei65	ei68	No
s8	ei65	t1	No
s8	ei65	t11	Yes

Many of the links suggested but not annotated are in fact correct from the text. For example; signal s8 (*before*) is said to describe the temporal relationship between ei65 *murder* and *curious*, which it does, as well as e.g. ei65 *murder* and ei68 *wrote*, which is also a correct description of that temporal relationship. However, these relations are not in the gold standard annotation (despite being correct interpretations of the text) and so they present as false positives. Because manual examination of all the false positives to detect errors of this kind would be time consuming, the 26.2 % figure that comes from automatic evaluation must be seen as a lower bound.

For a more concrete evaluation, one can constrain the set of signal associations considered to that described by TLINKs in the document. That is, we assume that events and timexes are known, and also that interval pairs (as in TLINK arguments) have been identified, and that the remaining tasks in a document's TimeML annotation are signal annotation and then TLINK relation type assignment. To this end, one only considers pairs of intervals that are also found in the gold standard. Thus, the evaluation problem is constrained somewhat, excluding the implicit temporal relation identification stage the initial evaluation includes. Therefore, this is referred to as the "constrained joint approach". It is implemented by, instead of using a window to choose interval pairings for consideration, using the pairing suggested in each of the annotated TLINKs.

In this case, there are 136 gold standard entities again. Result are given in Table 5.31. The system finds 99 signalled interval pairs that have arguments corresponding to a TLINK in the gold standard. Of these 99, 88.9 % (88) are correct annotations (e.g. precision is 88.9 %); the remaining 11 are spurious. This gives a recall of 64.7 % and F1 of 74.9 %. We describe these with F1 and not the Matthews correlation coefficient often associated with evaluating binary classifiers because the set of true negatives is very large in this case but not very interesting, and F1 does not take them into account.

In summary, using no signal information from the gold standard and simply relying on models for signal annotation, we achieve a 74.9 % F1 rate for the overall joint task of identifying temporal signal expressions and linking each expression found to a pair of intervals that it temporally co-ordinates.

Table 5.31 Details of the constrained joint approach to signal annotation

Signal/TLINK associations	Count	Proportion (%)
In N45	136	–
Found	99	–
Found, both args in N45	88	88.9
Signal in N45, new TLINK assoc	0	0.00
Found based on new signals	11	11.1

5.8.2 Combined Signal Annotation and Relation Typing

We know that signals are helpful in informing TLINK labelling. We also know that we can automatically annotate signals, to a reasonable degree of accuracy. It remains to be seen whether this degree of accuracy is sufficient for automatically-created signal annotations that are of overall help in TLINK labelling. It may be that the TLINK labelling information provided by signals is offset by imperfect automatic signal annotation, or that false positives in signal annotation provide misleading and counter-productive information to TLINK labelling.

In this section, experiments are reported whose aim is to determine whether automatic signal annotation has an impact on the overall task of TLINK labelling. We take the N45 section of the AQUAINT corpus as the dataset. It is curated to add missing signals, intervals and associations (details in Table 5.32). Two experiments are conducted. The first, a baseline, is over the manually signal-augmented version of the N45 docs (AQN45-sig) using a link labelling model trained on TB-sig, including no signal-specific features. This ignores temporal signals and represents the situation where a gold standard annotation is performed and a model learned without any signal information, and evaluated over unseen data. The second experiment uses TB-sig to learn models for signalled and non-signalled TLINKs, using the signal features described in Sect. 5.3.1, and then evaluates the performance of these models at labelling their respective parts of the automatically signal annotated version of N45 described in Sect. 5.8.1. This represents the scenario of having already annotated events, timexes and pairing intervals, then doing automatic signal annotation on unseen data, and evaluates how helpful these signal annotations are for TLINK labelling. We exclude new TLINKs identified in the course of automatic signal association, as we have no gold standard the relation type of these. The version of N45 with automatically generated signal annotations is referred to as AQN45-auto.

The distribution of interval pair types and TLINKs in the training data, TB-sig, is shown in Table 5.33. Similar data for evaluation corpora is in Table 5.32.

Table 5.32 TLINK stats over corpora used for extrinsic evaluation

Corpus	TLINKs	Non-signalled	Signalled	Signal %
AQN45	1 048	932	116	11.1 %
AQN45-sig	1 062	915	147	13.8 %

Table 5.33 Training dataset sizes from TB-sig used for signal annotation models

Interval types	Non-signalled	Signalled
Event-event	3 179	343
Event-time	2 299	529
Time-time	126	14

Table 5.34 TLINK labelling accuracy over corpora used for extrinsic evaluation. The baseline is the overall most-common-class for TLINKs in the training data (TB-sig). Interval text features are not included. There were no timex-timex links. The difference between the first two rows shows the impact that this total asignal discrimination and association approach has on TLINK labelling accuracy

Corpus	Subset of links	Event-Event (%)	Event-Time (%)	Overall (%)	Baseline (%)
AQUAINT N45 plain	All	44.0	56.4	55.8	28.9
AQN45-auto	All	62.0	58.4	58.6	28.9
AQN45-auto	Unsignalled	50.0	58.6	58.5	28.4
AQN45-auto	Only signalled	66.7	56.8	59.2	32.0
AQN45-sig	Only signalled	70.5	72.2	71.64	32.8

It can be seen that TLINKing based on automatic signal annotations, detailed in the second row (AQN45-auto/all) of Table 5.34, performs better than TLINKing with no signal information (the first row). The approach is therefore effective.

However, signalled TLINKs in the gold standard are still labelled substantially better than when automatic signal annotations are used (compare the fourth and fifth rows). Event-event links tend to draw particular benefit from signal annotations (see second and third columns), and this is still the case with automatic signal annotations; 66.7 % accuracy was achieved on the signalled event-event links, and 70.5 % using gold-standard links, compared to only 44.0 % labelling accuracy without any signal information. Overall, event-event temporal relation typing performance on this dataset increased from 44.0 % accuracy ignoring signals to 62.0 % when using automatically annotated signals – an 18.0 % performance increase, or 32.1 % error reduction.

The N45 part of the AQUAINT corpus unfortunately has a much lower event-event: event-timex TLINK ratio than TimeBank, with only 50 event-event versus 1 012 event-time links (4.71 % of the whole). For comparison, TB-sig has 2 828 event-time links to 3 522 event-event; event-event comprise 55.5 % of links. The bias in N45 has therefore led to an underestimate of the extra impact that signal information has on general event-event labelling. Nonetheless, the results confirm the efficiacy of the automatic signal extraction method, and show an overall 2.8 % absolute improvement in TLINK labelling over data without signals.

5.9 Chapter Summary

Temporal signals are an important source of information for temporal relations.

This chapter presented a principled investigation into temporal signals and the role they play in relating and ordering events and times within discourse.

It first presented a linguistic account for temporal signals, followed by a demonstration of their utility in the relation typing task, with a prototype supervised learning approach to temporal relation typing with signals that achieved error reduction of 53 % compared to the same system without signal information.

Given this strong motivation for exploring signals, a corpus analysis of temporal signals was conducted, examining an existing TimeML-annotated corpus. This was followed by a brief attempt at automatic temporal signal annotation which quickly revealed insufficient quality in signal annotations. As a result, the corpus was re-annotated with extra signals, including the events, timexes and temporal relations that the new signals required. This resource is made publicly available, as TB-sig.

Having a strong corpus, an approach for automatic signal annotation could be developed. This was taken as a two-part task. Firstly, as many signal expressions are polysemous, one must determine which occurrences of candidate signal words occur having a temporal sense. This was achieved with 83.0 % precision. Secondly, given a signal, one must determine which temporal intervals it co-ordinates. Two approaches to this problem were addressed – one considering intervals one at a time and ranking them, then assuming that the top two are linked, and another considering each possible pair of intervals. The interval pair approach worked best, achieving 83.5 % precision.

Having developed both stages of the signal annotation mechanism, these were evaluated jointly against a new gold-standard signal corpus derived from the AQUAINT TimeML corpus. With the least-constrained, hardest evaluation technique, 64.7 % of the gold-standard annotations were found automatically by the discrimination/association system proposed in this chapter.

Finally, with a full signal annotation system developed, the impact of automatic signal annotation on the overall task of temporal relation typing was evaluated. Results were positive. Adding automatic signal annotations and then feature representations of these automatically-found signals improved the absolute performance of a temporal relation type classifier by 18 % for event-event links and 2.0 % for event-time links.

In summary, we showed that temporal signals were useful in temporal relation typing, and developed approached for automatically annotating them, which performed well enough to give a net performance increase in the temporal relation typing task.

References

1. Hitzeman, J.: Semantic partition and the ambiguity of sentences containing temporal adverbials. Nat. Lang. Seman. **5**(2), 87–100 (1997)
2. Ho-Dac, L., Péry-Woodley, M.: Temporal adverbials and discourse segmentation revisited. In: Multidisciplinary Approaches to Discourse (2008)
3. Bestgen, Y., Vonk, W.: Temporal adverbials as segmentation markers in discourse comprehension. J. Mem. Lang. **42**(1), 74–87 (1999)
4. Brée, D., Smit, R.: Temporal relations. J. Seman. **5**(4), 345 (1986)

5. Brée, D., Feddag, A., Pratt, I.: Towards a formalization of the semantics of some temporal prepositions. Time Soc. **2**(2), 219 (1993)
6. Schlüter, N.: Temporal specification of the present perfect: a corpus-based study. Lang. Comput. **36**(1), 307–315 (2001)
7. Vlach, F.: Temporal adverbials, tenses and the perfect. Linguist. Philos. **16**(3), 231–283 (1993)
8. Hitzeman, J.: Text type and the position of a temporal adverbial within the sentence. In: Proceedings of the 2005 international conference on Annotating, extracting and reasoning about time and events, pp. 29–40. Springer (2005)
9. Derczynski, L., Gaizauskas, R.: A corpus-based study of temporal signals. In: Proceedings of the Corpus Linguistics conference (2011)
10. Setzer, A., Gaizauskas, R.: Annotating events and temporal information in newswire texts. In: Proceedings of the Second International Conference On Language Resources And Evaluation (LREC-2000), Athens, Greece, vol. 31 (2000)
11. Derczynski, L., Gaizauskas, R.: Analysing temporally annotated corpora with CAVaT. In: Proceedings of the Language Resources and Evaluation Conference, pp. 398–404 (2010)
12. Derczynski, L., Gaizauskas, R.: Using signals to improve automatic classification of temporal relations. In: Proceedings of the ESSLLI StuS (2010)
13. Derczynski, L., Gaizauskas, R.: Temporal signals help label temporal relations. In: Proceedings of the annual meeting of the Association for Computational Linguistics. Association for Computational Linguistics (2013)
14. Mani, I., Verhagen, M., Wellner, B., Lee, C., Pustejovsky, J.: Machine learning of temporal relations. In: Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics, p. 760. Association for Computational Linguistics (2006)
15. Bethard, S., Martin, J., Klingenstein, S.: Timelines from text: identification of syntactic temporal relations. In: Proceedings of the International Conference on Semantic Computing, pp. 11–18 (2007)
16. Zipf, G.: The Psycho-biology of Language. Houghton-Mifflin, Boston (1935)
17. Quirk, R., Greenbaum, S., Leech, G., Svartvik, J., Crystal, D.: A Comprehensive Grammar of the English Language, vol. 1. Longman, New York (1985)
18. Dorr, B., Gaasterland, T.: Summarization-inspired temporal-relation extraction: tense-pair templates and treebank-3 analysis. Technical Report. CS-TR-4844, University of Maryland, College Park, MD, USA (2006)
19. Mani, I., Hitzeman, J., Richer, J., Harris, D., Quimby, R., Wellner, B.: SpatialML: annotation scheme, corpora, and tools. In: Proceedings of LREC, vol. 8 (2008)
20. Verhagen, M., Saurí, R., Caselli, T., Pustejovsky, J.: SemEval-2010 task 13: TempEval-2. In: Proceedings of the 5th International Workshop on Semantic Evaluation, pp. 57–62. Association for Computational Linguistics (2010)
21. Reichenbach, H.: The tenses of verbs. Elements of Symbolic Logic. Dover Publications, New York (1947)
22. Stevenson, M., Wilks, Y.: Word sense disambiguation. The Oxford Handbook of Computational Linguistics, pp. 249–265. Oxford University Press, Oxford (2005)
23. Navigli, R.: Word sense disambiguation: a survey. ACM Comput. Surv. **41**(2), 1–69 (2009)
24. Lapata, M., Lascarides, A.: Learning sentence-internal temporal relations. J. Artif. Intell. Res. **27**(1), 85–117 (2006)
25. Charniak, E.: A maximum-entropy-inspired parser. In: Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference, pp. 132–139. Morgan Kaufmann Publishers Inc. (2000)
26. Kiss, T., Strunk, J.: Unsupervised multilingual sentence boundary detection. Comput. Linguist. **32**(4), 485–525 (2006)
27. Klein, D., Manning, C.: Fast exact inference with a factored model for natural language parsing. Adv. Neural Inf. Process. Syst. **15**, 3–10 (2003)
28. Marcus, M., Kim, G., Marcinkiewicz, M., MacIntyre, R., Bies, A., Ferguson, M., Katz, K., Schasberger, B.: The Penn Treebank: annotating predicate argument structure. In: Proceedings

of the workshop on Human Language Technology, pp. 114–119. Association for Computational Linguistics (1994)
29. Bies, A., Ferguson, M., Katz, K., MacIntyre, R., Tredinnick, V., Kim, G., Marcinkiewicz, M., Schasberger, B.: Bracketing guidelines for Treebank II style Penn Treebank project. University of Pennsylvania (1995)
30. Blaheta, D., Charniak, E.: Assigning function tags to parsed text. In: Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference, p. 240. Morgan Kaufmann Publishers Inc. (2000)
31. Musillo, G., Merlo, P.: Assigning function labels to unparsed text. In: Proceedings of RANLP'05 (2005)
32. Blaheta, D.: Function tagging. Ph.D. thesis, Department of Computer Science, Brown University (2004)
33. Gabbard, R., Marcus, M., Kulick, S.: Fully parsing the Penn Treebank. In: Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics, pp. 184–191. Association for Computational Linguistics (2006)
34. Lintean, M., Rus, V.: Naive bayes and decision trees for function tagging. In: FLAIRS Conference, pp. 604–609 (2007)
35. Marcus, M., Marcinkiewicz, M., Santorini, B.: Building a large annotated corpus of English: The Penn Treebank. Comput. Linguist. **19**(2), 330 (1993)
36. Quinlan, J.: C4. 5: Programs for Machine Learning. Morgan Kaufmann, San Mateo (1993)
37. Freund, Y., Schapire, R.: A decision-theoretic generalization of on-line learning and an application to boosting. J. Comput. Syst. Sci. **55**(1), 119–139 (1997)
38. Freund, Y., Schapire, R.: Experiments with a new boosting algorithm. In: Machine Learning: Proceedings of the Thirteenth International Conference, pp. 148–156 (1996)
39. De Marneffe, M., MacCartney, B., Manning, C.: Generating typed dependency parses from phrase structure parses. In: Proceedings of the International Conference on Language Resources and Evaluation (2006)
40. Verhagen, M.: Times Between The Lines. Ph.D. thesis, Brandeis University (2004)
41. Kanhabua, N., Nørvåg, K.: Improving temporal language models for determining time of non-timestamped documents. In: Research and Advanced Technology for Digital Libraries, pp. 358–370. Springer (2008)