TEMPORAL INFORMATION IN NEWSWIRE ARTICLES: AN ANNOTATION SCHEME AND CORPUS STUDY

By

Andrea Setzer

SUBMITTED IN PARTIAL FULFILMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY
AT
UNIVERSITY OF SHEFFIELD
SHEFFIELD, UK
SEPTEMBER 2001

Table of Contents

Abstract

Many natural language processing applications, such as information extraction, question answering, topic detection and tracking, would benefit significantly from the ability to accurately position reported events in time, either relatively with respect to other events or absolutely with respect to calendrical time. However, relatively little work has been done to date on the automatic extraction of temporal information from text.

Before we can progress to automatically position reported events in time, we must gain an understanding of the mechanisms used to do this in language. This understanding can be promoted through the development of an annotation scheme, which allows us to identify the textual expressions conveying events, times and temporal relations in a corpus of 'real' text.

This thesis describes a fine-grained annotation scheme with which we can capture all events, times and temporal relations reported in a text. To aid the application of the scheme to text, a graphical annotation tool has been developed. This tool not only allows easy mark-up of sophisticated temporal annotations, it also contains an interactive, inference-based component supporting the gathering of temporal relations. The annotation scheme and the tool have been evaluated through the construction of a trial corpus during a pilot study. In this study, a group of annotators was supplied with a description of the annotation scheme and asked to apply it to a trial corpus.

The pilot study showed that the annotation scheme was difficult to apply, but is feasible with improvements to the definition of the annotation scheme and the tool. Analysis of the resulting trial corpus also provides preliminary results on the relative extent to which different linguistic mechanisms, explicit and implicit, are used to convey temporal relational information in text.

Acknowledgements

Many people have given me much support throughout this thesis, and I would like to take this opportunity to thank them.

Kalina Bontcheva, Roberta Catizone, Patrick Herring, Diana Maynard, Horacio Saggion and Nick Webb, who spent hours of their free time annotating for me, which was vital for this thesis. My colleagues Roberta Catizone and Nick Webb, who took a great deal of work load off my shoulder, making it possible for me to concentrate on my thesis in the important final stages. My mum, my sisters Bettina and Jane, their respective partners Uschi and Achim, for the love and support they provided me with. A special thank you to my partner John Beranek, for the patience, support and love given. My friends Kalina Bontcheva, Sarah Chambers, Anita Dutta-Keane, Bethan Goodman Jones, Ian Seymour, Andrew Wilcox, Amanda Williams, and Mike Williams for their encouragement and inspiration. The members of the Natural Language Research group who have provided an interesting and inspiring work environment. Mike Holcombe, who had many useful comments. Mark Hepple, especially, who often listened to me, read my work, and had many suggestions.

This thesis could not have been accomplished without my supervisor, Robert Gaizauskas. His encouragement, guidance, patience (especially with my use of the English language), and kindness have given me the motivation to carry on. His level of knowledge, and wit (and sarcasm) were a constant source of inspiration. Thank you, Rob.

Sheffield, September 2001

Andrea Setzer

Chapter 1

Introduction

1.1 The Problem

Many natural language processing applications such as information extraction (IE) (Cowie and Lehnert, 1996; Gaizauskas and Wilks, 1998), question answering (QA) (Voorhees, 2001; Voorhees and Harman, 2001), topic detection and tracking (Fiscus et al., 2001) involve information about temporally located events. IE, for example, typically involves scenarios such as terrorist attacks, management successions events, rocket launches or plane crashes. For all these applications the ability to accurately position key events in time is of great importance. Take the following article from the New York Times as an example.

> SMALL PLANE CRASHES INTO ATLANTIC; NO SURVIVORS FOUND
>
> A small single-engine plane crashed into the Atlantic Ocean about eight miles off New Jersey on Wednesday. The Coast Guard reported finding aircraft debris and a fuel slick, but no bodies or survivors.
>
> The plane, which can carry four people, was seen hitting the water shortly after 11 a.m. by a fisherman, who radioed the Coast Guard, according to Petty Officer Jeff Fenn, a spokesman for the base at Governors Island in New York Harbor.
>
> By midafternoon, several vessels and a helicopter were combing the area about eight miles east of Sea Bright, N.J., and seven miles south of the Ambrose Light,

> the Coast Guard said. The area is 55 miles from the site off Long Island where a TWA 747 crashed one week earlier.
>
> Searchers found the plane's landing gear, seat cushions and other debris, Petty Officer Fenn said. He said the water is about 125 feet deep in the crash area and that much of the wreckage had sunk.
>
> The Coast Guard said the craft had taken off from Allaire Airport in Monmouth County, N.J. The Federal Aviation Administration said the plane was registered to Delaware Environmental Development Service of Wilmington. There was no listing for the company in Wilmington.
>
> 07-24-96

Without knowledge of the temporal relations between the events mentioned in the article, what information would we get from the text? Two planes crashed, debris was found, a plane was seen hitting the water, several vessels and a helicopter were combing the area. We would not be able to answer simple questions like *When did the plane crash?* or *What kind of plane crashed one week earlier?* nor could we draw time graphs like the one shown in figure 1.1.

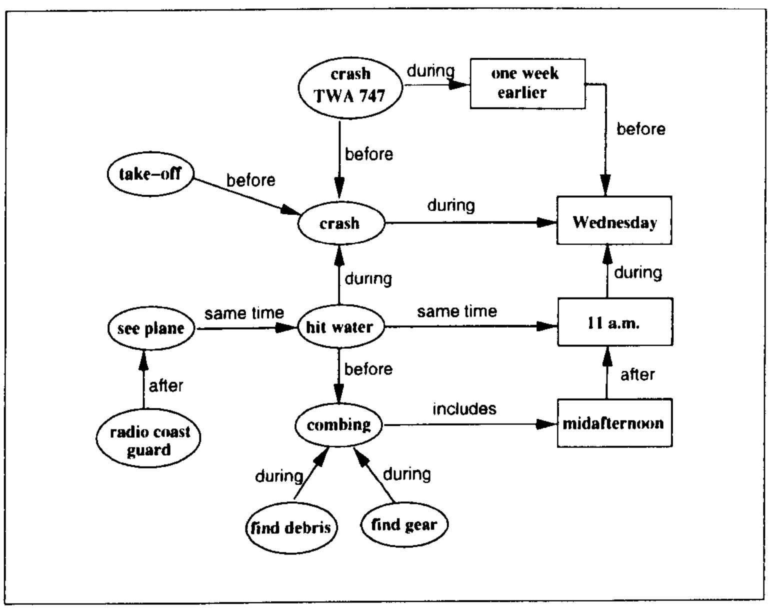

Figure 1.1: Time graph for the introductory example

Another application, which is becoming known as 'people tracking' (Schiffman et al., 2001), creates short biographies of people from information like the following example.

> William C. Walbrecher Jr., an executive at San Francisco-based 1st Nationwide Bank, was named president and chief executive officer of Citadel Holding Corp. and its principal operating unit, Fidelity Federal Bank. The appointment takes effect Nov. 13.
>
> 07-04-97

The 'bibliography' outcome based on this example could look like this:

name:	William C. Walbrecher Jr
position1:	executive
company1:	1st Nationwide Bank
location1:	San Francisco
date1:	until 11/12/1997
position2:	president and chief executive officer
company2:	Citadel Holding Group, Fidelity Federal Bank
location2:	unknown
date2:	from 11/13/1997

This task could not be performed at all without temporal information since bibliographical information without dates is not very useful. We need to know when positions were taken and left and we often need the dateline of the article to interpret temporal expressions like *Nov. 13* and fill in the correct date of *11/13/1997* into the date field.

It is easy for humans to answer the questions mentioned above and to draw a time graph like the one in figure 1.1[1]. But when we want to develop a system to automatically

[1]There may be more than one valid time graph representing the temporal information in a text, we will come back to time graphs in section 2.3.

determine the temporal relations between the events in a text, we have to analyse in detail what it is that enables us humans to effortlessly perform the same task.

A wide variety of techniques is used to convey temporal information in natural language text. Verb tense, narrative sequence, temporal prepositional or adjectival phrases and subordinate conjunctions are obvious and explicit ways. But there are ways of conveying temporal information which are not as obvious. Event co-reference is often used in newswire texts, where events (like the plane crash above) are introduced in the beginning and then come back to introduce more detailed information (the plane being seen at 11 a.m. by a fisherman, for example, provides the reader with more detailed temporal information than given in the beginning of the article). General world knowledge is relied upon as well. The reader is assumed to know about plane crashes and know that a rescue operation follows a crash and not the other way round. Without this world or 'script' knowledge, we would not know that the taking off happened before the crash (the perfective aspect is not a reliable indicator).

Events need to be identified by the system before they can be related. This bears its own difficulty since humans do not always agree on whether a sentence conveys an event or not. In the article above, the sentence *[...] the plane was registered to Delaware Environmental Development Service of Wilmington* could be seen as a referring to a state or an event, for example. Thus, a prior problem is that of defining events.

Even though temporal information is important for most natural language applications, it has not been addressed widely. To date only minimal work has been done in the IE community concerning the extraction of temporal information from text, as evidenced by tasks set in recent Message Understanding Conferences (MUC). The MUC-6 named entity subtask required the identification of absolute time expressions in text (MUC6), and the MUC-7 named entity subtask extended this requirement to include relative time expressions (MUC7), but none of these tasks required placing events in time, or temporally relating events to each other. The MUC-5 and -7 scenario tasks did require participants to assign a calendrical time to certain specified event types (joint venture announcements and rocket launchings, respectively). However, this task was quite limited and the scores were low.

indicating its difficulty. More recently the TIDES Temporal Guidelines (Ferro et al., 2000; Wilson et al., 2001) have been developed – a very thorough set of guidelines for annotating time expressions. These will clearly supersede the MUC named entity guidelines for time expressions, but again are not setting out to annotate events and temporal relations between events or between events and times.

The importance of extracting temporal information from texts, together with the difficulty of the task, suggest that a concerted effort be made to automatically determine the relative order or, if possible, the absolute time of the events reported in the texts. The basis of such a system is a sound conceptual base which defines those features and relations in texts which enables humans to perform this task and an annotation scheme to annotate them. This work aims to contribute to this overall goal by providing such a base.

Both formal discourse approaches and corpus-driven approaches provide frameworks and mechanisms on which a system to extract temporal information from text can be based. A choice has to be made early on, as to which of these approaches to adopt. We have opted for a corpus-driven approach, as will be discussed later on in the thesis.

With an annotation scheme available, annotated corpora can be built, which will yield the benefits normally associated with the construction of such resources: a better understanding of the phenomena of concern, and resources for the training and evaluation of adaptive algorithms to automatically identify features and relations of interest. This will further aid the development of automated systems to, for example, extract the temporal relations between events in a text and also lead to a deeper understanding of temporal phenomena in text.

1.2 Objectives

We develop an annotation scheme which identifies those features in texts which enable the human reader to determine the temporal order and, where possible, the calendrical time of the events reported in them. The basis of this scheme is a conceptual framework which contains the entities and temporal relations needed to perform the task. To validate the annotation scheme, it is applied to a trial corpus by a number of annotators. The resulting annotations are compared against a (human produced) gold standard and the results are reported and analysed.

The genre chosen is that of newswire articles. Much IE work has been done in this genre, which will benefit from the inclusion of temporal information, and newspaper articles are also readily available in machine readable form.

1.3 Thesis Structure and Methodology

The problem we are dealing with is the temporal ordering of events in newswire text. The first part of the thesis describes the background for this problem. Chapter 2 describes methodologies of conceptualising time and the entities involved. How these temporal entities are expressed in natural language is the focus of chapter 3. We have already mentioned in the introduction that we can either adopt a formal approach or a corpus-driven approach. Chapters 4 and 5 look at those in more detail.

The second part of the thesis describes our approach. In chapter 6 we explain the choices we made in the light of the background as well as detailing the conceptual framework we have adopted based on these choices. The annotation scheme we developed, based on the conceptual framework, is described in detail in chapter 7. The development of the conceptual framework, temporal ontology and annotation scheme are closely related and although they have been laid out in a sequential way, the process which led to them was a cyclical one. Repeated cycles of applying the annotation scheme to real text and analysing resulting problems lead to refinement of the scheme and often to discovering different ways of conveying the information to be annotated.

We used the annotation scheme in a pilot study where a group of annotators was supplied with the scheme and asked to apply the annotation scheme to a trial corpus. In particular we were interested in answers to these questions:

- how unambiguous and comprehensive are our temporal annotation guidelines?
- how much genuine disagreement is there about temporal relations in text?
- how burdensome is our annotation procedure? - i.e. is it feasible to think of annotating a corpus of significant size at this level of detail?

The process of annotation and the framework for evaluating the results are described in chapter 8, the annotation guidelines given to the annotators can be found in appendix A. We developed a graphical annotation tool to assist the annotators which also contains an interactive component to aid establishing temporal relations. The results of the pilot study and the answers to above questions - as far as we have been able to determine them - can be found in chapters 9 and 10.

1.4 Contribution

We have developed an annotation scheme for temporally ordering events which forms the basis for developing systems which will automatically identify events in a text and establish their relative order and, where possible, their absolute position on a time line. For the moment, this annotation scheme is aimed at newswire texts but it can easily be adapted to other genres.

We also developed a graphical annotation tool to assist the annotation to apply the scheme. The tool also contains an interactive component which aids the gathering of temporal relations.

The annotation scheme as well as the tool were validated in a pilot study we conducted. The pilot study resulted in an annotated trial corpus. The results of the study and the temporal features of the corpus were analysed and we identified problems and improvements.

Chapter 2

Conceptualising Time

When developing ontologies we have to begin by considering what the ideas are we want to represent. We are dealing with time and what happens or obtains in time, so we need to define a model for the entities we suppose exist or are relevant for our purposes. Usually, *time* is modelled using *instants* and/or *intervals*; *states* and *events* are what obtains/happens in time. The modelling of these three entities is strongly interrelated and will be described in section 2.1. Once we have established a model for time, events and states, we can turn to the temporal relations holding between them (section 2.2).

We adopt a naive philosophical view of the world. We assume the following:

1. Events happen and states pertain in *reality*. A plane crashing into the Atlantic ocean on the 3rd of September 2001 is an example.

2. These events and states are described or represented by *propositions*, often in first order logic or other variants of logic. The example above may be described by

   crash(plane,Atlantic ocean), time(crash,3.9.2001).

3. The same proposition can be expressed by several different *sentences*, for example in different languages: *A plane crashed into the Atlantic ocean on Monday the 3rd of September 2001* or *Letzten Montag stürzte ein Flugzeug in den Atlantischen Ozean.*

In short, sentences express propositions which themselves describe reality; with this we have three levels of abstraction. The same levels of abstraction hold for times too: different time expressions can express the same time concept, which refers to a 'time in reality'. For example, the expressions *last Monday*, *yesterday* and *September 3rd* can all be represented by date(03092001) (in the appropriate context). The time described is September 3rd, 2001. We will sometimes use a shortcut and say that a sentence expresses an event or state or that a time expression refers to a time, skipping the propositional/conceptual level for reasons of brevity.

2.1 Time, Events and States

When modelling time and what happens or obtains during time, there are, broadly speaking, two different avenues we can take. We can either first establish a temporal framework, a model of time itself and then superimpose the different types of change that can occur in time (like events and states). Time does then exist independently of change, in what Galton (1995) calls *independent-time models*. The alternative is to first establish a model for change and then to define the temporal structure in terms of this model. Time only exists depending on the notion of change with which is is defined. These models are called *dependent-time models*. Philosophically, the latter model seems more satisfactory since we seem to experience change and events directly, but not time.

Independent from the model one chooses, there are different ways of classifying events and states as well as times. We will described these, respectively, in sections 2.1.1 and 2.1.2.

2.1.1 Classifying Events

The propositions which are conveyed by sentences in context can, according to Vendler (1967), be classified into *aspectual categories* (also called *aspectual types*, *Aktionsarten* or *situation types*). Aspectual here refers to the intrinsic temporal properties of the verb and not to the perfective or progressive aspect of verbs (more about this in section 3.4.1). These aspectual categories can be broadly classed into ***events*** ("a thing that happens") and ***states*** ("the existing condition or position of a person or thing"), both quotations are taken from the Oxford Compact English Dictionary (Thompson, 1996).

As we can see from the rather vague dictionary entries, defining what exactly events and states are is a difficult undertaking. We will nevertheless attempt to informally define them here. A ***state*** is a relation between entities or the holding of an attribute of an entity which, while capable of change, is ongoing over a time span; often without a defined beginning or end. Typically, a change of state constitutes an event. An example of a sentence expressing a state is *Elvis is alive.* An ***event*** is intuitively something that happens, with a defined beginning and end, like the event conveyed by *a small single-engine plane crashed into the Atlantic ocean.*

Propositions expressing ***states*** typically describe indefinitely extending 'states of affairs' (Moens and Steedman, 1988) or conditions and are constant throughout their duration. On the sentence level, they combine happily with the perfect and are (almost) the only propositions that can be expressed in the English language using the simple present tense (the exception being performatives like *I name this ship Walther*). Examples of sentences expressing states are:

(1) *Elvis is alive.*

(2) *I have been happy.*

(3) *I know geography.*

(4) *I love Jazz.*

Events have a defined beginning and end and a behaviour in time. Sentences conveying events combine happily with the progressive form. Examples for such sentences are:

(5) *He is running.*

(6) *Keats wrote a sonnet.*

Events, or dynamic situations can be distinguished further. The most common classification goes back to Vendler (1967), who included states and three types of events in his classification. He separated events on the grounds of duration vs. punctuality and conclusiveness vs. non-conclusiveness[1]. Vendler only recognised three out of the four possible combinations. A fourth option, non-conclusive punctual situations, were introduced by Miller and Johnson-Laird (1976) (according to (Steedman, 1997)). We will briefly discuss the four classes below, a summarisation can be found in table 2.1. The terminology is adopted from Moens and Steedman (1988), who, amongst others, have extended Vendler's work.

Culminations (called *achievements* by Vendler) are viewed as punctual or instantaneous and they are "accompanied by a transition to a new state of the world" (Moens and Steedman, 1988). The new state is referred to as the *consequent state*. Sentences expressing culminations combine happily with the perfect and examples are:

(7) *Sally reached the top.*

(8) *Harry ran a mile.*

(9) *Walther has written a song.*

Another type of punctual expression typically not associated with a consequent state is called *point*, a category not recognised by Vendler. "A point is an event (not necessarily an instantaneous one) that is viewed as an indivisible whole and whose consequences are not an issue in the discourse" (Moens and Steedman, 1988). Points differ from other events also in that sentences conveying them do not combine with the perfect. Examples of expressions conveying points are:

(10) *John hiccuped.*

(11) *Mary winked.*

[1]A durative event is, for example, *building a house* whereas a typical punctual event is *recognising something.* A conclusive event has an intrinsic termination, (*running a mile*) whereas a non-conclusive event does not have this property (*running*)

(12) *#John has hiccuped.*[2]

The third type of expressions are those describing events which are extended in time but which do not have a particular conclusion or culmination point (*non-conclusive* events). This aspectual category is called *process*. Vendler drew attention to the fact that sentences expressing processes (or *activities* in his terminology) can be combined with *for*-adverbials but not with *in*-adverbials. They also combine with the progressive.

(13) *Jack ran.*

(14) *Hillary climbed for several hours.*

(15) *#Hillary climbed in several hours.*

(16) *Jack is running.*

The last class of expressions are those describing events which are extended in time but which also have a particular culmination with which a consequent state is associated. This aspectual class is called a *culminated process*. Sentences conveying a culminated process combine readily with *in*-adverbials but not with *for*-adverbials, as the following examples show. Vendler called the type of event *accomplishment*.

(17) *Hillary climbed to the top.*

(18) *Hillary climbed to the top in an hour.*

(19) *#Hillary climbed to the top for an hour.*

The propositions conveyed in a natural language sentence can be distinguished on a small number of dimensions. On the highest level, there are states and events. Events can be subclassified on just two dimensions, one based on the contrast between punctuality and temporal extension and the other dimension based on the association with consequent states. Table 2.1, taken from Moens and Steedman (1988), summarises this subcategorisation.

[2]Following convention, we use a # to mark examples that are anomalous, unless supported by rather unusual contexts.

	events		states
	atomic	extended	
+conseq	**CULMINATION** recognise, spot, win the race	**CULMINATED PROCESS** build a house, eat a sandwich	understand,love, know, resemble
-conseq	**POINT** hiccup, tap, wink	**PROCESS** run, swim, walk, play the piano	

Table 2.1: Event classification (from Moens and Steedman, 1988)

Moens and Steedman stress the importance of recognising that aspectual classes are properties of sentences seen in context; they are not attached to certain verbs. Linguistic devices such as tense, aspect and temporal or aspectual adverbials can transform entities of one type into another. For example, the argument of the progressive auxiliary is expected to be a process. If used with, for example, a punctual type (*Harry is hiccuping*), then the proposition conveyed will be reinterpreted as a process, in this case an iteration of the basic event. Not all transitions are valid, for example combining a perfect auxiliary with something other than the culmination it demands, can lead to an anomalous expression: *#the clock has ticked.*

The permissible transitions, called ***type coercions***, between aspectual classes are strongly related to what Moens and Steedman call the ***nucleus*** - an event structure containing "a culmination, a preparatory process and a consequent state", see figure 2.1. Figure 2.2 shows the nucleus associated with the expression *climbing Mt Everest.*

Events can be classified in more than one way, although the Vendler based approaches are the most common. Halliday (1994), for example, introduced the following event classification. In his "Functional Grammar" he sees clauses as describing ***processes*** (not to be

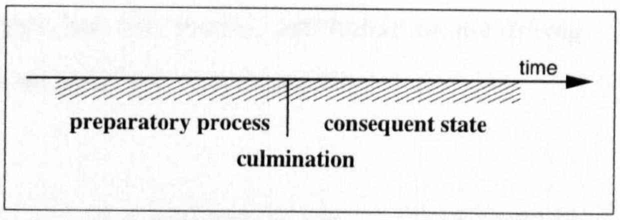

Figure 2.1: Event nucleus (adapted from Moens and Steedman, 1988)

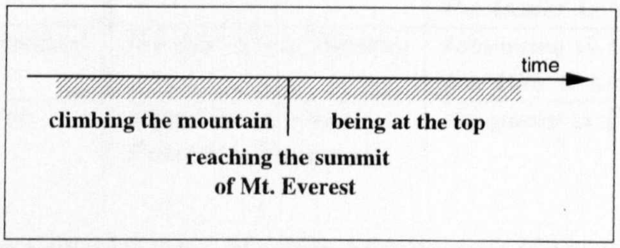

Figure 2.2: Example for an event nucleus (adapted from Moens and Steedman, 1988)

confused with Moens and Steedman's use of the term). Halliday claims that "our most powerful conception of reality is that it consists of 'going-ons': of doing, happening, feeling, being. These going-ons are sorted out in the semantic system of language, and expressed through the grammar of the clause" (Halliday, 1994, pg. 101).

Halliday distinguishes the following types of process (quotations are taken from Halliday (1994)).

Material processes or processes of *doing* are conveyed by sentences like *the lion ran* or *the lion caught the tourist.* They involve an obligatory ACTOR ("who 'does' something" – the lion) and an optional GOAL ("the one to which the process is extended" – the tourist).

Mental processes or processes of *sensing* are for example *I don't like Marmite* or *I believe you.* They typically involve an obligatory SENSER and a PHENOMENON ("that which is 'sensed'"). The phenomenon is obligatory but does not need to be explicit, as in *Jane cannot see*, where the something that Jane cannot see is implied. Processes of *perception* (*seeing, hearing,...*), *affection* (*liking, fearing,...*) and *cognition* (*thinking, knowing,...*) are subtypes of mental processes.

Relational processes or processes of *being* are one of three types: *intensive, circumstantial* or *possessive*. Each type has two modes, *attributive* or *identifying*. Table 2.2, taken from Halliday (1994), gives an overview and examples.

	attributive mode	identifying mode
intensive	*Sarah is wise*	*Tom is the leader; the leader is Tom*
circumstantial	*the fair is on Tuesday*	*tomorrow is the 10th; the 10th is tomorrow*
possessive	*Peter has a piano Peter's is the piano*	*the piano is Peter's;*

Table 2.2: The principal types of relational processes (from Halliday, 1994)

In the attributive mode, the participants are ATTRIBUTE (*wise*) and CARRIER (*Sarah*), whereas in the identifying mood they are IDENTIFIED (*Peter*) and IDENTIFIER (*piano*).

Other process types are *behavioural processes* (*dreaming, breathing, smiling, coughing* ...), *verbal processes* (*saying,* ...) and *existential processes* (*existing,* ...).

2.1.2 Classifying Time

When constructing models for time, either instants or intervals (or both) can be chosen as primitive, and used to construct the other concept. We can only give a summary here, see Galton (1995) for a good overview.

Instant Based Structures

When we chose instants as the primitive units of times then we need to make clear what we think instants really are. We can think of them as atoms or building blocks of time, which means they are indivisible and very small. Taken to the extreme this would mean that instants are durationless, but durationless instants lead to the puzzle of how temporal duration can arise from durationless primitives. The obvious alternative is to insist that every instant has a finite duration but not have a lower bound on the duration.

Time can be seen as linear or as branching. The most important branching model is forward branching time where each point has a unique past but more than one future. Temporal order is treated as having a tree-like structure, very similar to the chronicles described in McDermott (1982). Figure 2.3 illustrates forward branching time.

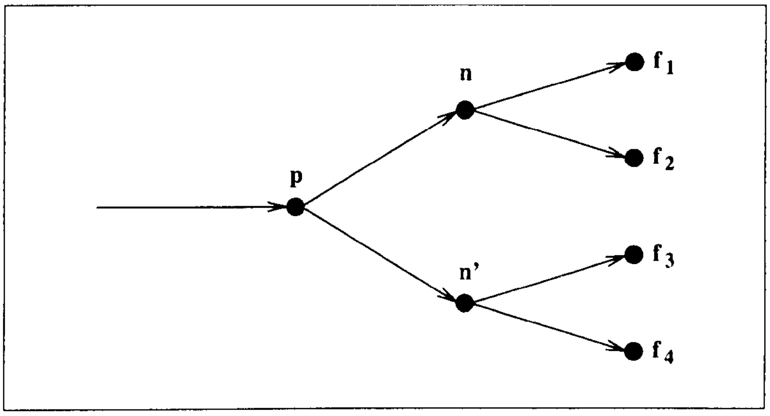

Figure 2.3: A Branching time structure (from Galton, 1995)

We can construct intervals from instants in two ways. We can either identify an interval with an ordered pair of instants (t_1, t_2), the interval being 'between t_1 and t_2 (such that $t_1 < t_2$)' or we can identify an interval as a convex set of instants, for example $\{t|t_1 < t < t_2\}$. The latter option has several advantages, for example, a natural notion of subinterval (a subset which is also an interval) and definitions of 13 irreducible interval-interval relations (Allen, 1983) and 5 irreducible instant-interval relations (Vilain, 1982) (see also section 2.2).

It has been argued that it would be both philosophically and computationally more satisfactory to dispense with instants as primitives altogether and use intervals as primitives instead, because:

1. empirically, we cannot distinguish between an event taking a very short time and a truly instantaneous event, so nothing is lost if we see all events as taking time and if we use intervals as temporal primitives;
2. "instants are an artefact of chronometry – they only arise out of our habit of assigning

numbers to times, together with the assumption that this assignment can be refined to arbitrary degrees of precision." (Galton, 1995)

Interval Based Structures

With intervals as primitives, a strict partial ordering is imposed on intervals with the transitive and asymmetric ordering relation $t < t'$ ("t wholly precedes t'"). With this ordering, 13 irreducible relations between arbitrary pairs of intervals can be defined, similar to Allen (1984)'s relations (see figure 2.4).

An operation to construct new intervals is needed, for example *join*, which takes two intervals as arguments and produces a new interval which begins where the earlier one begins and ends where the later one ends. Instants are constructed from intervals by defining them as the point at which two intervals meet. See Galton (1995) for more information on the difficulties of this process.

Allen (1983) introduced an important feature of reasoning about intervals the composition of interval-interval relations. He produced a look-up table with all 169 possible combinations, which aids automatically reasoning about intervals. This was further expanded by Freska (1992), who introduced the notion of 'conceptual neighbourhood'.

Most temporal frameworks include both instants and intervals in their ontology and those who do not, seem to be heavily criticised. Allen (1984), for example, developed a framework based on intervals alone. He gave two reasons for this. First, times can be decomposed into subtimes; whenever we look at an occurrence there is always a more detailed causal explanation if one cares, and is able to look for it. Thus, times seem to correspond to intervals on the real line. Second, time points in addition to intervals are not introduced because they are not necessary and, more important, they present difficulties with the semantics of Allen's logic. Allen was criticised by Galton (1990), mainly because Allen cannot represent a property being true at an instant without it being true throughout an interval. This is needed to represent, for example, a continuously moving object which is at a point in space only for an instant.

2.2 Temporal Relations

When one has decided on a temporal ontology, one can can turn to the temporal relations that hold between the entities in the ontology.

For *independent-time models*, we get 13 irreducible interval-interval relations, as introduced by Hamblin (1969) and Allen (1983). The 13 relations are the relations shown in figure 2.4 and their inverses.

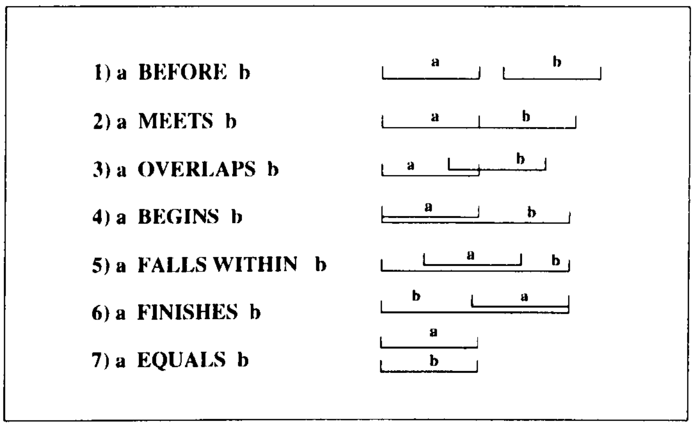

Figure 2.4: Interval-interval relations (from Galton, 1995)

Between instants and intervals, the 5 irreducible relations, shown in figure 2.5, can hold between an instant t and an interval a (Vilain, 1982).

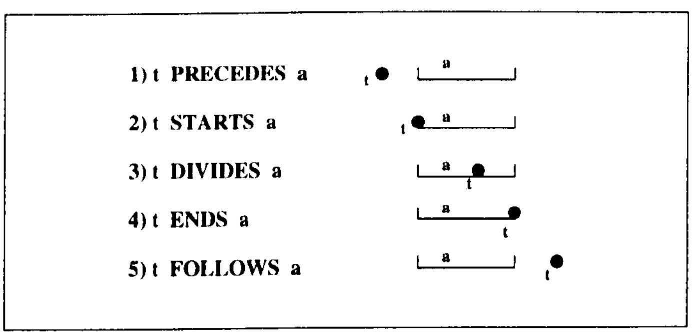

Figure 2.5: Instant-interval relations (from Galton, 1995)

These relations are independent from the choice of instants or intervals as temporal primitives, although the way the relations are defined slightly varies. For more information see Galton (1995).

For ***dependent-time*** models, where states and events come first and times are defined in terms of these, we look at the temporal relations between events or rather between what Galton (1995) calls ***event-tokens***, which are reifications of event-types. An example taken from Galton (1995) is: *Antony's ascent of Helvellyn in 1975* is an event-token of *Antony climbs Hellvellyn, Antony climbs a mountain, Antony spends some time walking uphill* and various others.

We can now define temporal relations between event-tokens by means of binary (or higher-ary) predicates. The only binary predicates holding between punctual events are SIMULTANEITY and SUCCESSION. For durative events we get the same 13 relations that hold between intervals and intervals (see figure 2.4), EQUAL being read as IS SIMULTANEOUS WITH. Between states and events, we can, for example, define the relation *while(s,e)*, which expresses that the event e occurred while the state s obtained.

inclin(t_i, t_j)	t_i is a time included in the time t_j
incl(t_i, t_j)	t_i is a time that includes the time t_j
begin(t_i, t_j)	t_i is a time whose beginning overlaps the time t_j
end(t_i, t_j)	t_i is a time whose ending overlaps the time t_j
after(t_i, t_j, D)	t_i is a time after the time t_j with a temporal distance D, expressed as a duration
before(t_i, t_j, D)	t_i is a time before the time t_j with a temporal distance D, expressed as a duration
relpos(X, T, t_i, t_j)	t_i is the X^{th} time of type T after the time t_j (or before, if $X < 0$, then it is $-X^{th}$ time)
extend(t_i, t_j, t_k)	t_i is the time period starting at time t_j and ending at time t_k

Table 2.3: Gagnon and Lapalme's temporal relations

The 13 interval-interval relations shown in figure 2.4 have been criticised by Gagnon and

Lapalme (1996) on the grounds that they are too precise for natural language processing purposes. They introduced a different set of more underspecified temporal relations which resemble more closely the way temporal localisation is dealt with in natural language. Gagnon and Lapalme claim that their relations are as expressive as the relations described above and that both are just two different ways of expressing the same temporal relations. Table 2.3 shows a slightly simplified account of Gagnon and Lapalme's relations.

2.3 Visualising Events, Time and, Temporal Relations

This section will talk about how events, times, and the temporal relations holding between them can be visualised.

The ideal situation would be to be able to place all events and times mentioned in a newswire article on a time line, i.e. to be able to associate a calendrical date with each entity. But we often do not have the information necessary to do this accurately. It might only be possible to place an event or time in time relative to another event, as in *the pilot ejected prior to the crash.* Figure 2.6 shows an attempt to put the *eject* event on a time line in relation to the *crash* event (assuming it happened in June 2000).

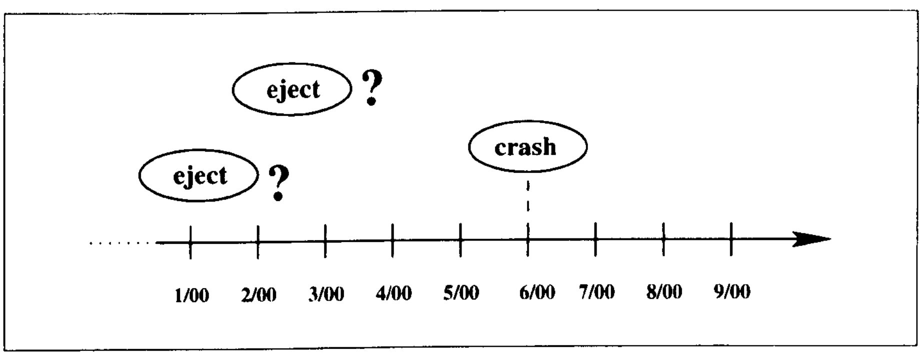

Figure 2.6: Time line example 1.

The problem is that we have to put the *eject* event somewhere on the time line which will inevitably lead to it being associated with the calendrical time it is being placed above, even though it should only be seen in relation to the *crash* event.

Another problem is that we might know that a group of events, e.g. B, C, and D, can be placed relative to another event, e.g. A, but we do not know their temporal relationship to each other. Figure 2.7 shows an attempt to do this.

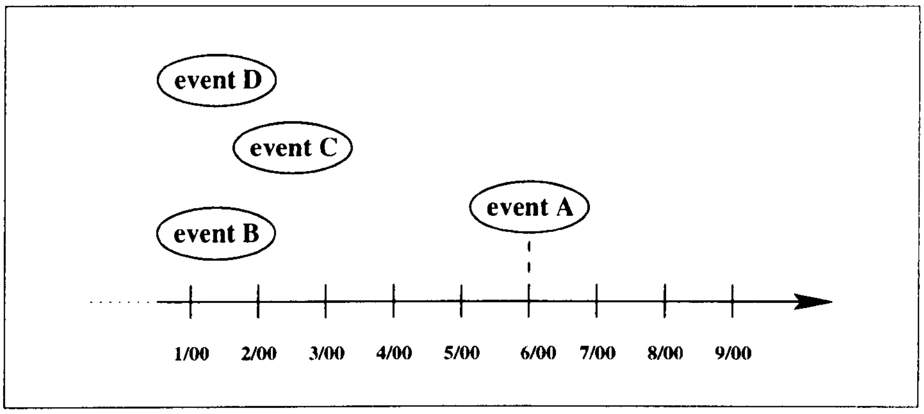

Figure 2.7: Time line example 2.

Not only do we have the same problem as before, we also have to place the events B, C and D relative to each other which gives the impression that we know their relationship to each other.

The problem is that time line representations force commitment to temporal detail that is often not warranted by the text. Hence we have chosen to use time graphs rather than time lines. Although we have not fully developed an approach for time graphs, the example above could look like the time graphs shown in figure 2.8.

2.4 Discussion

In this chapter, we have described different ways of conceptualising time. Classifications for events, states, times, and temporal relations have been introduced and discussed, as well as a method of visualising these entities. This chapter, together with chapter 3 about how

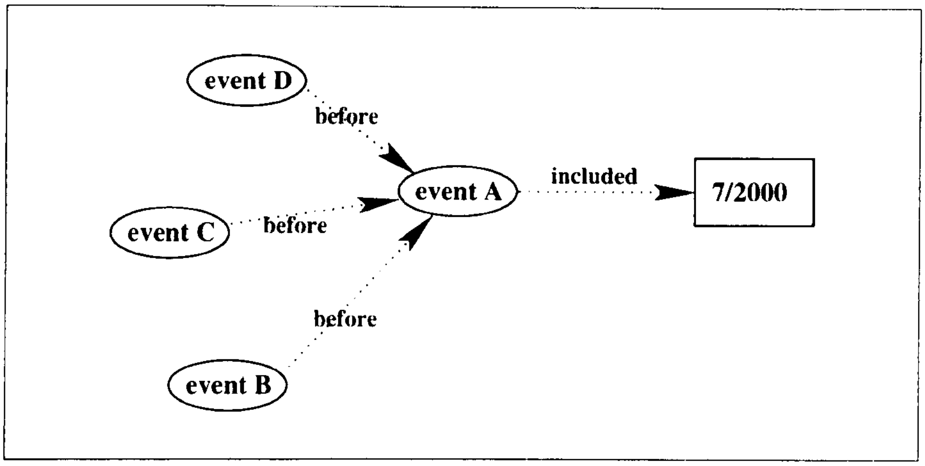

Figure 2.8: Time line example 3.

temporal information is conveyed in language, chapter 4 about formal discourse approaches, and chapter 5 about corpus-driven approaches, form the background information on which the annotation scheme described in chapter 7 is based.

Chapter 3

Time and Language

The task we address in this thesis is that of ordering events in time and in the chapter 2 we have discussed the possible kinds of conceptual frameworks and models that can be developed. Independent of the framework chosen, it has to contain events, states, times and temporal relations since these are paramount to the task. In this chapter, we will talk about how these entities are conveyed in natural language. This is part of the basis on which our annotation scheme (detailed in chapter 7) has evolved.

In sections 3.1, 3.2 and 3.3, we will describe, respectively, how events, states, and times are expressed in natural language. The temporal relations that can hold between events and events or between events and times are the subject of section 3.4. Temporal relations also exist between times and times, but since this is only rarely expressed directly in text, we will not go into the mechanism here.

We believe that genre affects which mechanisms are used to convey temporal information, and in section 3.5 we will briefly describe the idiosyncrasies (as far as they affect temporal ordering) of the genre this thesis is aimed at - newswire articles.

3.1 Expressing Events

Most events are expressed by finite clauses and finite verbs can be thought of as indicators of events, as in the following examples.

(20) *The action came after an F-14A fighter crashed today in the Atlantic.*

(21) *The aircraft was introduced to the fleet in the mid 1970s.*

(22) *Aviation safety and engineering experts will review the data.*

(23) *And on Jan. 29, an F-14A crashed into houses in Nashville after takeoff, killing the two crew members and three people on the ground.*

(24) *The National Transportation Safety Board is borrowing a Boeing 737 from Seattle's Museum of Flight.*

Finite verbs are not the only indicators; events can be expressed by nominalisations (examples (25) to (27)) and non-finite verbs (examples (27) to (29)) as well.

(25) *The cause of today's crash is undetermined.*

(26) *Two Government Flying Service helicopters joined Macau Marine Police in the search for the Sikorsky 76.*

(27) *And on Jan. 29, an F-14A crashed into houses in Nashville after takeoff, killing the two crew members and three people on the ground.*

(28) *... apparently returning from an oil rig ...*

(29) *when he crashed an F-14A jet fighter in Nashville in January, killing himself, a fellow officer and three people on the ground.*

3.1.1 Problematic Cases

In the previous section we described how finite verbs, nominalisations and non-finite verbs can be employed to express events. Unfortunately, the presence of these do not guarantee that an event is represented, as the following examples show.

(30) *The area* <u>*is*</u> *55 miles from the site off Long Island.*

(31) *The writ* <u>*is*</u> *for "damages, interest and costs" of seven passengers who died* ...

Expressions like these seem to describe attributes or states rather than events (more about states in the next section). If states conveyed by finite clauses using the verb *be* were the only exception then we could still define how events are expressed in purely syntactical terms. But events are more complicated than that, because what we perceive as an event can change with the domain or time span a newswire article covers. Consider the following two situations.

1. A newspaper article in April 2001 about a demonstration in Berlin against the transport of nuclear waste to other countries. This article might mention that Berlin is the capital of Germany.
2. A newspaper article about the political development of Germany over the last 200 years, describing among other things when Berlin was the capital of Germany.

In the first example, the fact that Berlin is the German capital *includes* the time span the article covers (one day in April 2001). Should we be asked to create a time graph of the article, like the one shown in the introduction (figure 1.1), we most certainly would want to put the events during the demonstration on it. But we might not want to put Berlin being the capital on the time graph, instead perceiving this fact as a state. In the second article, however, the situation would be different. Berlin being the capital of Germany from 1991 onwards, especially since it used to be the capital before, would most certainly be part of the time graph and is *included* in the time span the article covers (200 years).

Thus, depending on the article and the *granularity* of the events in the text, the same clause can convey an event or a state. Syntactic criteria alone are not enough and we need to employ semantic criteria as well.

Even if what is conveyed by, for example, a finite clause is temporally included in the time span of the article, it can still be difficult to decide whether an event is expressed or not. It seems to be the *kind* of verb, the 'semantic class' it belongs to, that causes the problem illustrated in examples (32) and (33).

(32) *They were partly persuaded by a surprising discovery found ...*

(33) *The circles of glass that cover many of the cockpit dials, and even a light bulb above a staircase that led to the plane's upper deck, had somehow survived the crash intact.*

Would one want to put the *persuaded* and *survived* event on a time graph? And if yes, could they be anchored in time (a prerequisite of placing an event on a time graph)? Sometimes it is not even clear how many examples are mentioned. Does example (34) refer to one or two events? Multiple events can be referred to in one expression, as example (35) shows.

(34) *Two cars crashed today on the M25.*

(35) *... while it investigates three recent F-14 crashes.*

While we were analysing articles, more abstract and philosophical questions arose, indicating the level of complexity involved once we start to define what events are. For example, is one team's win the same as another team's loss? Do examples (36) and (37) represent the same event?

(36) *Sheffield Wednesday won 3:0 against Arsenal.*

(37) *Arsenal lost 0:3 against Sheffield Wednesday.*

3.2 Expressing States

The core indicator of a state is a finite verb expressing a relation between entities (example (38)) or a copula signalling the possession of an attribute by an entity (example (39)). Of course not all copulas express states. But definitional sentences which assert properties or relations which cannot change without the subject ceasing to be what it is, can also be seen as stative. For example, sentences taking generics as subjects (example (40)), or giving information about class attributes (example (41)) or describing intrinsic properties (example (42)) do not refer to states but are perceived as stative.

(38) *The piano belongs to John.*

(39) *Elvis is alive.*

(40) *Altimeters are instruments for ...*

(41) *The plane, which can carry four people ...*

(42) *The area is 55 miles wide.*

As will be explained in section 6.5, we have defined states to be able to exclude them from being annotated.

3.3 Expressing Times

Most time expressions are simple, they are temporal prepositional or adverbial phrases as examples (43) to (48) show.

(43) *The action came after an F-14A fighter crashed <u>today</u> in the Atlantic.*

(44) *<u>Sunday</u>, an F-14D crashed into the Pacific Ocean off southern California, killing its two crew members.*

(45) *And on <u>Jan. 29</u>, an F-14A crashed into houses in Nashville.*

(46) *The aircraft was introduced to the fleet in the <u>1970s</u>.*

(47) *By <u>midafternoon</u>, several vessels and a helicopter were combing the area about eight miles east of Sea Bright, N.J.*

(48) *... which was signed on <u>October 12, 1929</u>.*

Holidays like *Easter Sunday* or *All Saints Day* are also time expressions. Most time expressions refer to times relative to the date of the article. Example (48) is one of the few where the date given is an absolute calendar date. But time expressions can be more complex as examples (49) and (50) show.

(49) *The sound occurred about 5 minutes and 47 seconds after takeoff.*

(50) *The captain "questioned" the sound 17 seconds after hearing it.*

In these examples, the underlined clause is the time expression. An interval of time (*17 seconds*) is related (*after*) to an event (*hearing it*) and the expression refers to the point at the end of the interval.

We distinguish between *referring* and *non-referring* time expressions. Referring time expressions refer to a time in reality, so that a calendar date can be associated with the expression, at least in principle. We include those that are referring to a fictional calendar, as might be found in stories. We also say that a time expression can be 'anchored in time'. All examples so far have been examples of referring time expressions.

Non-referring time expressions cannot be anchored in time and they cannot be associated with a calendar date. Examples (51) to (54) show non-referring time expressions.

(51) ... *while it investigates three recent F-14 crashes.*

(52) *The Navy has ordered a 72-hour safety standdown for all F-14 aircraft.*

(53) *The Navy ordered its Northrop Grumman Corp. F-14s out of the skies for three days* ...

(54) *The Archers omnibus is on Sundays.*

3.4 Expressing Temporal Relations

Temporal relations can be expressed by either relating events to times or by relating one event to another. The former will be described in sections 3.4.1 and 3.4.2 and the latter in section 3.4.3.

3.4.1 Tense, Aspect and Modality

The correspondence between the form of the verb and what we understand as time is expressed using *tense*. The verbal action can be in progress or completed, which is conveyed by *aspect*, and it can also be related by *mood* to conditions such as certainty, obligation, necessity and possibility. These three systems are interrelated, for example the expression of the future is closely related to mood and the the expression of present or past time cannot be done separately from aspect.

Tense is the most fundamental of these three mechanisms. One of the most influential works in this area is Reichenbach's 'Elements of Symbolic Logic' (Reichenbach, 1947). His most important contribution is the introduction of a *reference point*. He argued that the tense system does not just predicate over two times ('now' and 'then'), but over three. He called these three points S (speech point), R (reference point) and E (event point). The event point can be thought of as the temporal extension of the proposition itself; the speech point is the time of the utterance and the reference point is "the time (or situation or context) that we are talking about" (Reichenbach, 1947).

English can be seen as having three tenses: past, present and future. In all three cases, the event point and the reference point coincide, shown in figure 3.1.

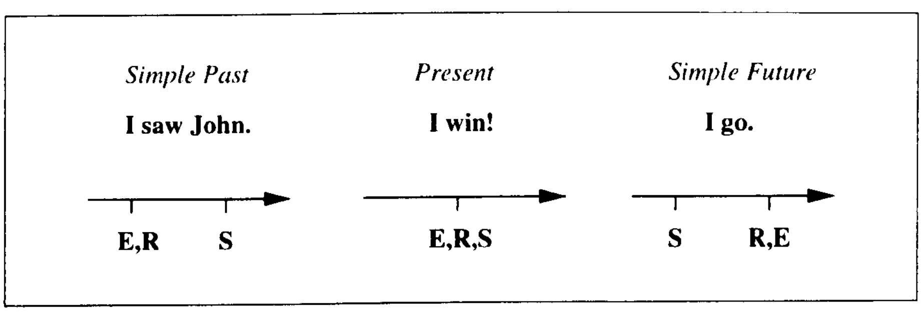

Figure 3.1: The tenses (from Steedman, 1997)

The importance of the reference time becomes clear when Reichenbach's theory is applied to show the differences between the past perfect, simple past and present prefect, all of which Reichenbach regarded as tenses proper (see figure 3.2). "The most important insight here is that the simple past is used to make a statement about a past time, whereas the perfect is used to make a statement about the present [...]" (Steedman, 1997).

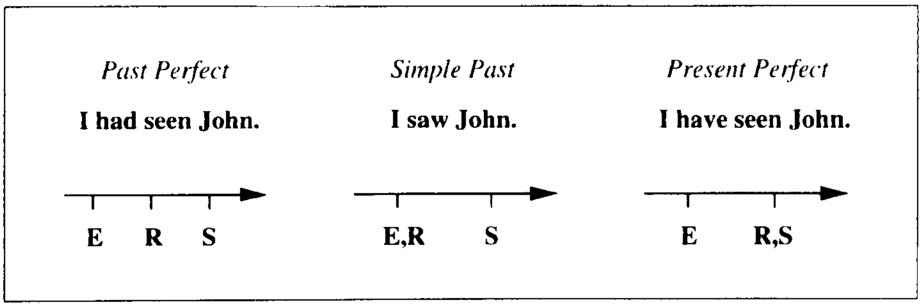

Figure 3.2: Past vs. perfect (from Steedman, 1997)

When we look at the Reichenbach approach in connection with the second system to convey temporal relations – aspect – then we can see a clear connection to the event classes based on Vendler (1967) and Moens and Steedman (1988) (see chapter 2 and especially 2.1). In the case of the perfect, the reference point R lies within a consequent state derived by the event E. This corresponds to the event class *achievement* or *culmination*. In the case of the progressive, the reference point lies within a progressive state, again derived from the event E; an *activity* or *process*. In either case, E does not figure directly in the representation shown in figures 3.3 and 3.4. In fact, the position of E relative to S and R is not fully determined by the perfect and the progressive (and is therefore shown in brackets in both figures).

Modality is the last of the three systems. Modal verbs in English can be classed into two groups. *Epistemic* modality is concerned with necessity, possibility or predictability as in the following examples (taken from Steedman (1997)).

(55) *It must have died.*

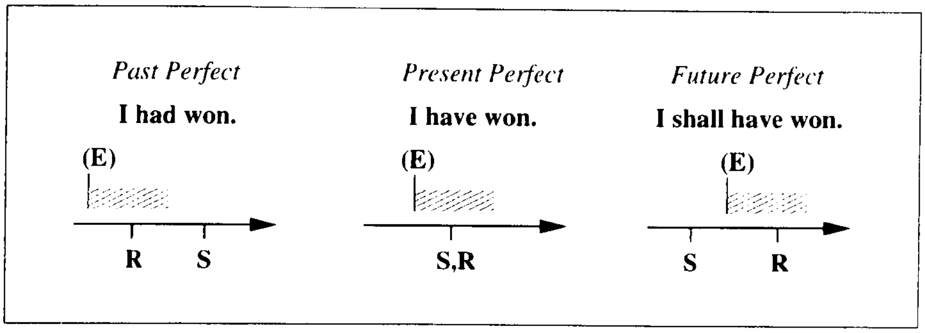

Figure 3.3: The perfect (from Steedman, 1997)

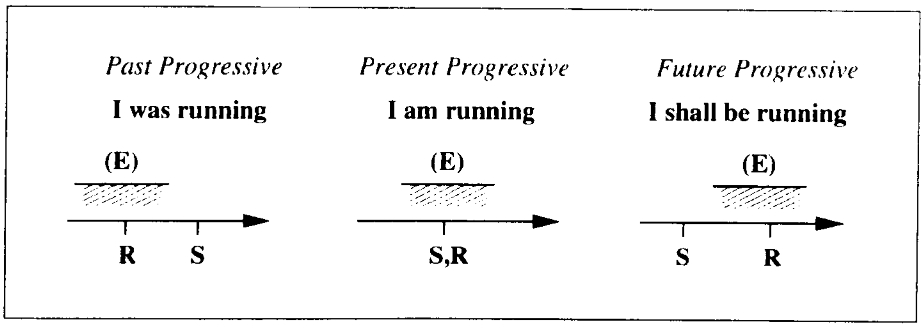

Figure 3.4: The progressive (from Steedman, 1997)

(56) *That will be the mailman.*

(57) *She may be weary.*

Deontic modals, on the other hand, are concerned with the feasibility and permissibility of the core proposition and ability and obligation of the agent, as in the following examples (again taken from Steedman (1997)).

(58) *You must sit down.*

(59) *You may smoke.*

(60) *I can do the Boogaloo.*

Epistemic modals are strongly related to the forward branching time model mentioned in section 2.1.2 and shown in figure 2.3, because of their involvement with necessity and possibility.

3.4.2 Prepositional Phrases and Adverbials

Events can be located in time by using temporal prepositional phrases, adverbials and adverbial phrases or deictic noun phrases (this section is based mainly on Quirk and Greenbaum (1973), from which the examples are taken.

Prepositional Phrases

- AT, ON, IN, DURING

  The preposition AT is used for points of time, mainly clock-time (*at ten o'clock, at noon, ...*), but also idiomatically for holiday periods (*at Christmas*) and for phrases such as *at night, at that time* etc. ON is used when referring to days (*on Monday, on May first, ...*), whereas IN and DURING are used to indicate periods of time (*in the evening, in August, ...*).

- FOR, OVER, ALL THROUGH,THROUGHOUT

  FOR expresses duration, as in *we camped here for the summer* and also in idiomatic phrases like *for good.* OVER, ALL THROUGH and THROUGHOUT have a similar durational meaning, as does FOR ... TO (*we camped here from June to September*). All these prepositions have a locative meaning which is transferred to duration (apart from FOR).

  FOR can be omitted (*we stayed there (for) three months*), and is always omitted if the phrase begins with *all*, such as *all day* or *all week.*

- BEFORE, AFTER, SINCE, UNTIL

  The prepositions in this part are used almost exclusively as prepositions of time. They are often followed by a temporal noun phrase (*before next week*). UNTIL is used to either specify a terminal point with positive predications (*we slept until midnight*) or a commencement point with negative predications (*we didn't sleep until midnight*).

- BETWEEN, BY, UP TO

  These are other prepositions of time, as in *I'll phone you between lunch and three o'clock; up to last week, I hadn't received a reply.* "BY specifies a commencement point" (Quirk and Greenbaum, 1973) (*by that time, he was exhausted*) and hence cannot be used together with verbs of durative meaning.

If adjuncts contain the deictic words *last, next, this* or *that*, or quantifying words like *some* or *every*, or nouns which have *last, next* or *this* as an element, then the preposition of time is always absent, as in:

(61) *I saw him last Thursday*

(62) *Plums are more plentiful this year*

(63) *Every summer she returns to her childhood home*

Prepositions can be ambiguous with regard to the temporal feature they indicate (position on the time axis, absolute duration, etc.). Bree and Feddag (1996) extracted those sentences from the Brown corpus (Kucera and Francis, 1967), which contain one of the prepositions *at, on, through, throughout, by, during, in, within, over, for* and *ago*. They analysed 90% of these sentences and, based on their analysis, they provide a set of rules to determine the temporal feature intended. The rules are based on the preposition itself, the determiner and the class of the noun of its prepositional phrase. A test of these rules on a different set of data is left as future work (the authors do not mention this set may be the remaining 10% of the sentences extracted).

Adverbials

Time adjuncts can be classed into the following three main semantic classes.

- Time *when* adjuncts, mostly adverbs, usually serve as a response to a *when* question (*when did he arrive?*). They can denote a point of time (common adverbs include *again, just, late, now, nowadays, presently, then* and *today*) or a boundary of time (common adverbs include *afterwards, before, eventually, formerly, just, lately, momentarily, previously, presently, recently, since* and *then*). Examples for time points are (64) and (65) and examples for boundaries are (66) and (67).

(64) *I was in New York <u>last year</u> and am <u>now</u> living in Boston.*

(65) *I'm <u>just</u> finishing my homework.*

(66) *I haven't any time at the moment but I'll see you <u>soon</u>.*

(67) *Take a drink and <u>then</u> go to bed.*

- Time ***duration*** adjuncts, which normally serve as a response to *(for) how long* questions (*how long are you staying for?*), can also be divided into two groups. They either denote length of time (common adverbs include *long, momentarily, permanently* and *temporarily*) or they denote duration from some preceding point of time (common adverbs include *lately, recently* and *since*). An examples for the former is (68), and examples for the latter are (69) and (70).

(68) *I'll go walking <u>for a few hours</u>.*

(69) *I have bought new walking boots, <u>since</u> I have done the 'five dales walk' with you.*

(70) *I have done two walks in the Peak District <u>lately</u>.*

- Time ***frequency*** adjuncts mostly serve as a response to a *how often* question (*how often do you wash your car?*). They are usually adverb phrases or noun phrases and can be divided into two classes: those naming the times by which the frequency is measured directly (***definite frequency***) and those who do not (***indefinite frequency***). Each subclass can be subdivided further, but we will not go into detail here (see Quirk and Greenbaum, 1973). Examples of definite frequency are shown in examples (71) to (73) and (74) to (76) are examples of indefinite frequency.

(71) *Committee meetings take place <u>weekly</u>.*

(72) *I have been in Singapore <u>once</u>.*

(73) *He <u>again</u> demanded a refund.*

(74) *We <u>normally</u> don't go to bed before midnight.*

(75) *I have often told him to relax more.*

(76) *I have been in this office <u>on several occasions</u>.*

3.4.3 Event-Event Relations

The techniques described in sections 3.4.1 and 3.4.2 are used to express the temporal relations that can hold between events and times. Temporal relations between events and events, when explicitly expressed in text, are the focus of this section.

Event-event relations are usually expressed by using temporal conjunctions such as *before* and *after*. In addition, *ing*-clauses without a subject can also express temporal relations, see examples (77) and (78) (from Quirk and Greenbaum, 1973).

(77) *Nearing the entrance, I shook hands with my acquaintances.*

(78) *The stranger, having discarded his jacket, moved threateningly towards me.*

The interpretation of temporal conjunction is linked to the event classes described in chapter 2 and we will give an analysis here, which is based on Ritchie (1979). Ritchie claims that one of the factors contributing to the interpretation of temporal clauses is whether the clause describes an 'ongoing' or 'instantaneous' event. The categories in the Vendler/Steedman classification system, which Ritchie adopts, cannot be separated neatly into 'ongoing' and 'instantaneous' and so Ritchie introduces the two new categories COMPLETED and CONTINUING for his analysis, see table 3.1.

COMPLETED	CONTINUING
all ACHIEVEMENTS	progressive verb forms
all ACCOMPLISHMENTS	*be* + complement clauses
ACTIONS without repeated interpretation	STATIVES
ACTIVITIES like *sleep*	'Semi-statives' like *live*
ACTIVITIES with a specified time duration	habitual/repeated interpretations (usually of ACTIONS
	ACTIVITIES like *run*

Table 3.1: COMPLETED and CONTINUING classes (Ritchie, 1979)

We will now briefly describe the most common types of temporal clauses (examples are taken from Ritchie (1979), some are slightly adapted).

- BEFORE

  The start of the process in the *before*-clause is used to indicate a time called the END-LIMIT - with both COMPLETED and CONTINUING clause types. With a COMPLETED clause, the entire process precedes the END-LIMIT (example (79)) and with a CONTINUING clause only the start of the process precedes the time given by the *before*-clause (example (80)).

  (79) *The car broke down before we left London.*

  (80) *We were cooking before the ceiling collapsed.*

- AFTER

  A COMPLETED clause indicates the end of the process conveyed as a time point (example (81)) whereas the time point referred to by a CONTINUING clause is the start of the process (example (82)). This time point is called the START-LIMIT.

  (81) *After he left*

  (82) *After they were talking to him*

  With a CONTINUING main clause, the end of that process follows the START-LIMIT of the bound clause (example (83)), but with a COMPLETED main clause, that whole process follows the START-LIMIT (example (84)).

  (83) *We were here after he left.*

  (84) *We arrived after he left.*

- WHILE

  Independent of whether they are COMPLETED or CONTINUING, *while*-clauses describe an INTERVAL.

  (85) *While you picked up the luggage*

  (86) *While you were picking up the luggage*

If the main clause is CONTINUING, i.e. describes an INTERVAL, then the bound clause interval is a subinterval of the main clause interval (example (87)). A COMPLETED main clause describes a POINT which is within the interval of the bound clause (example (88)).

(87) *The sun was shining while you were outside.*

(88) *He left while you were outside.*

- UNTIL

  Both COMPLETED and CONTINUING *until*-clauses define a single time point which is given by the start of the process described. The difference between *before* and *until*-clauses is that *before*-clauses give only a limit whereas *until*-clauses specify an exact end-point for the main clause process (called END-LIMIT). Only CONTINUING main clauses are valid and their process continues right to the END-POINT given by the bound clause.

  (89) *He ran until he was past the bridge.*

  (90) *It was sunny until we left the country.*

- AS SOON AS

  Similarly to *until* and *before*, *as soon as* differs from *after* in that *after* merely gives a limit whereas *as soon as* defines the exact point a process starts – the START-POINT. This START-POINT is the end of the bound process for a COMPLETED *as soon as*-clause (example (91)) and the start of the bound process for a CONTINUING *as soon as*-clause (example (92)). Independent from the type of the main clause, the main clause process begins exactly at the START-POINT.

  (91) *The raccoon climbed the tree as soon as it saw us.*

  (92) *John went swimming as soon as the water was warm.*

- AS LONG AS

  An *as long as*-clause describes an interval during which a process goes on. Only CONTINUING main clauses are valid and the bound clause "must be open to some kind of durative interpretation" (Ritchie, 1979). The bound clause interval is a subinterval of the main clause interval.

(93) *He lived in London as long as he could afford it.*

(94) *The alligators stayed away as long as Frank splashed the oar in the water.*

- WHEN

  The most basic connective is also the hardest one to describe. A COMPLETED *when*-clause describes a POINT (example (95)) and a CONTINUING *when*-clause describes an INTERVAL (example (96)).

  (95) *When I opened the box ...*

  (96) *When I was opening the box ...*

  Depending on the type of the main clause, there are four possible interpretations, shown in table 3.2.

When-clause	main clause	relationship
point x	point y	x approx. coincides with y
point x	interval i	x contained in i
interval i	point x	x contained in i
interval i	interval j	i is a subinterval of j

Table 3.2: *When*-clause relationships (from Ritchie, 1979)

3.4.4 Implicitly Expressed Relations

The methods of conveying temporal relations that we have described so far have all been *explicitly* expressed, for example by using temporal conjunctions. But temporal relations can be also expressed implicitly and we will give a brief overview here of several methods of doing so.

Events can be referred to more than once, which is referred to as *event co-reference*. This is similar to co-reference between persons or inanimate objects. Identical events happen at the same time and inferences can be drawn from this (more about this later on).

Subevents are referred to as well in text. The sentence *We decorated the kitchen yesterday* describes an event that has many subevents, for example painting a particular wall or painting the skirting boards. A sentence like *I painted the east wall and Nell painted the skirting boards* refers to two subevents, which are temporally included in the 'decorating event' and therefore happened 'yesterday' as well.

Causality is not strictly a temporal relation holding between events but it has temporal consequences. It can be explicitly expressed, as in *John because Harry puched him* but it need not be. The sentences *John fell. Harry pushed him* describes the same events and the same causal relationship between them. The temporal consequence is that cause precedes effect and thus the 'decorating events' happens after the 'exploding'.

Another method of expressing temporal relations is *narrative sequence*, as in *John entered the room. He picked up a book and sat down in front of the fire.* Here, the reader naturally interprets that the 'entering' happened before the 'picking up' which in turn happened before the 'sitting down'. The sequence in which these events are reported reflects the way in which the events happened.

Temporal relations can also be inferred. We have already done so in the examples before. When we talked about subevents and that the 'painting events' were temporally included in the 'decorating', we quite naturally inferred that the 'painting' must have also happened 'yesterday'. Another example is the following: *John went to the V&A before he went to the Tate. Afterwards he had a meal in a Chinese restaurant.* The 'V&A event' happened before the 'Tate event' which in turn happened before the 'meal event'. We infer from this that the 'V&A event' also happened before the 'meal'. Or, to say it in a different way, if A happens before B and B happens before C, then A also happens before C. These are two simple examples of many possible inference rules. More about these rules will follow in section 7.3.

These are only a few examples of how events can be implicitly related, giving a brief glimpse into the complex area of implicitly expressed relations.

We will now turn to the characteristics of newswire texts and explain which kind of methods this genre favours, before concluding this chapter in section 3.6.

3.5 Characteristics of the Genre Newswire Texts

In the previous sections of this chapter, we described several mechanisms with which temporal information and especially temporal relations can be conveyed. Which of these mechanisms are used and to what extent, is partially predicted by the genre. Based on Bell (1991) and Bell (1998), we will describe the characteristics of newspaper reporting, a genre unto itself, and briefly explain which mechanisms it favours. The experiments in chapter 9 will contribute to the task of quantifying the proportions of mechanisms used.

Consider again the example we gave in the introduction (repeated here for reasons of clarity).

> SMALL PLANE CRASHES INTO ATLANTIC; NO SURVIVORS FOUND
>
> A small single-engine plane crashed into the Atlantic Ocean about eight miles off New Jersey on Wednesday. The Coast Guard reported finding aircraft debris and a fuel slick, but no bodies or survivors.
>
> The plane, which can carry four people, was seen hitting the water shortly after 11 a.m. by a fisherman, who radioed the Coast Guard, according to Petty Officer Jeff Fenn, a spokesman for the base at Governors Island in New York Harbor.
>
> By midafternoon, several vessels and a helicopter were combing the area about eight miles east of Sea Bright, N.J., and seven miles south of the Ambrose Light, the Coast Guard said. The area is 55 miles from the site off Long Island where a TWA 747 crashed one week earlier.
>
> Searchers found the plane's landing gear, seat cushions and other debris, Petty Officer Fenn said. He said the water is about 125 feet deep in the crash area and that much of the wreckage had sunk.

> The Coast Guard said the craft had taken off from Allaire Airport in Monmouth County, N.J. The Federal Aviation Administration said the plane was registered to Delaware Environmental Development Service of Wilmington. There was no listing for the company in Wilmington.
>
> 07-24-96

The most important characteristics of newspaper articles are the following. Firstly, newspaper articles typically exhibit a non-chronological time structure, which results from obeying news values rather than ordinary narrative norms. In the article above, the first event mentioned is the plane crash which is the most important one, the one the article is about. Chronologically, the plane taking off from Allaire airport happened before the crash, but is not mentioned until the end. Secondly, newspaper articles frequently adopt the so-called 'instalment method' by which an event is introduced and returned to in more detail two or more times later in the text. When the plane crash is mentioned first, only the minimum of information is given. In the second paragraph, more details are reported, like a more precise time and that the plane was seen by a fisherman. The search for survivors is treated similarly. First the most important facts are given – that no bodies or survivors had been found. Later on, more details are given. Thirdly, and strongly related to this second point, news articles tend to follow the 'inverted pyramid style' which means that all main points are made in the beginning and then the article processes through decreasingly important information. The most important points are all made in the very beginning, the first paragraph - the headline even. A plane crashed Wednesday and no survivors were found. And the last paragraph makes the least important points, where the plane was registered, for example.

One of the reasons for employing these methods is that the reader has to be able to stop at any point and abandon the article and still know what happened. Finer details might be missed but the most important points have to be made in the very beginning.

From this it is clear that different linguistic mechanisms may be used to convey temporal information in newswire articles than in other types of narrative. In particular, event co-reference plays an important role whereas aspect, in the sense of Aktionsarten, or lexical aspect (see section 2.1.1), does not play a large role, mainly because the short paragraphs are not necessarily connected.

Another characteristic of newspaper articles is that they do not only report events, they also report what is called *attributions* in this genre. The source, and with that the reliability, of the information, is important and often included in the text. This is called attribution; examples (97) and (98) show the attribution underlined.

(97) *<u>The Coast Guard said</u> the craft had taken off from Allaire Airport in Monmouth County, N.J.*

(98) *<u>Aeroflot general manager for Hong Kong Vassili Tkatchenko said</u> on Tuesday he was unaware the writ had been filed.*

The relevance for the temporal analysis of the article becomes clear when we look at example (98). The 'reporting event' (*[...] Vassili Tkatchenko said*) happens at a particular time (*Tuesday*) and the 'reported event' (*he was unaware the writ had been filed*) must have happened before. This gives a more accurate temporal location of the 'reported event' then indicated by the past tense of the verb, which would have located the 'reported event' before the date of the article only.

3.6 Discussion

In this chapter, we have described mechanisms that are used to convey events, states, times and temporal relations in natural language. We also introduced the genre newswire texts in more detail and explained how this genre affects which of the described mechanisms are used in newswire texts. The insights into the mechanisms and the genre are very influential for the evolution of the annotation scheme described in chapter 7.

Chapter 4

Formal Discourse Approaches

In the introduction, we discussed two different approaches one can take towards analysing temporal information in text – formal discourse approaches and corpus-driven approaches. In this chapter, we will discuss the merits of formal discourse approaches and their suitability for the goal – the development of a system to automatically locate the events in a newswire text in time.

It became clear very early during this research that a large proportion of the temporal information is not explicitly expressed, but has to be inferred. Moving towards an approach which offers the means to draw these inferences, as formal discourse approaches do, can be an advantage. This chapter will first give an overview of the three most important formal approaches developed so far, and then discuss their suitability in section 4.4.

Discourse Representation Theory (section 4.1) is the furthest developed and most commonly used formalism to represent the semantics of multi-sentence natural language texts, including temporal information, and to reason about it. Ter Meulen (1995) introduced a different way of representing temporal information using so-called *dynamic aspect trees*; her method is described in section 4.2. Moulin (1997) comes from a more AI-oriented background, and he uses conceptual graphs to represent the temporal information contained in text – see section 4.3.

4.1 Discourse Representation Theory

Kamp and Reyle (1993) have developed a theory of representing the semantics of coherent multi-sentence discourse: Discourse Representation Theory (DRT). The key idea is that each new sentence S of a discourse is interpreted in the context provided by the sentences preceding it. This means that the context is updated with the contribution made by S. This 'updating' includes the linking of anaphoric references. New as well is the idea that the same structure serves as content and context; these were kept apart in older approaches. But this 'double duty' leads to special constraints on the logical form. This idea is implemented in form of interpretation rules which identify the semantic contribution to the context.

Important for the updating is the introduction of elements, called reference markers or discourse markers, which serve as antecedents to anaphoric expressions. These referents play a key part in the context structures used (Discourse Representation Structures or DRS).

Following van Eijck and Kamp (1997), we use their simple examples to illustrate the central concepts of DRT. Consider example (99).

(99) *A* man^1 *entered.* He_1 *smiled.*

In predicate logic, this could be represented as

(100) $\exists x\ (man(x) \wedge entered(x) \wedge smiled(x))$

As already mentioned, in DRT, the interpretation of the first sentence results in an initial DRS which is then updated by interpreting the second sentence. The resulting context (or DRS) essentially represents the same truth values as (100) does. The initial structure or context must not only capture the existential interpretation of *a man entered*, it must also be capable of acting as a context for the interpretation of *he smiled* and transforming the initial DRS into a DRS that satisfies the truth conditions of (100). The DRS for the first sentence of example (99) can be seen in figure 4.1.

x
man x entered x

Figure 4.1: Example DRS

The interpretation of the noun phrase (NP) *a man* leads to the introduction of the reference marker x and the two conditions *man* x and *entered* x. Reference markers serve as antecedents for anaphoric NPs when the context is used in the interpretation of the next sentence. In the example, x is available as the antecedent for the pronoun *he* and the discourse marker introduced for *he* can be linked to x. Interpreting the second sentence of example (99) results in the transformed DRS shown in figure 4.2.

x y
man x entered x x = y smiled y

Figure 4.2: Transformed DRS

Generally speaking, a DRS consists of

1. a set of reference markers, the *universe* of the DRS; and
2. the *condition set* of the DRS

All reference markers get an existential interpretation and are available as antecedents to pronouns and other anaphoric expressions. The interpretation of a sentence S in a context C of DRS K results in a new DRS K', which in addition to the context represented by K also contains the content of S (as interpreted with respect to K).

Not all information has the logical form of an existentially qualified conjunction. For example, consider (101), for which a conditional interpretation is needed.

(101) *If a man enters, he smiles.*

The antecedent of a conditional describes a situation (*a man enters*) which must also satisfy the information specified in the consequent (*he smiles*). Within DRT, the interpretation of the consequent uses the interpretation of the antecedent as a context, similar to a sentence S being interpreted within the context provided by the previous discourse. The representation of (101) (or any other conditional) must also represent the conditional connection between antecedent and consequent. The DRS for (101) is given in figure 4.3.

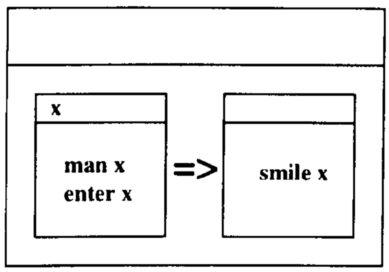

Figure 4.3: DRS for a conditional

An important principle in connection with conditions of the form $K \Rightarrow K'$ is that reference markers in the ***main universe*** of the DRS get an **existential** interpretation whereas reference markers on the left-hand side of the ***subordinate universe*** of an $\Rightarrow$-condition get a **universal** interpretation (note that the right-hand side does not contain a discourse marker). An important difference between discourse markers in the main and in the subordinate universe is that only the former are accessible as antecedents to subsequent pronouns. Kamp and Eijck give the discourses in examples (102) and (103) to illustrate this point.

(102) *A man came in. He smiled. He was holding a flower in his right hand.*

(103) *If a man comes in, he smiles. ?He is holding a flower in his right hand.*

The second *he* in (102) is not problematic whereas the second *he* in (103) is. This is reflected by the fact that the reference marker for *a man* in a DRS for example (102) belongs to the main universe (and is accessible) but in the DRS for (103) the reference marker for *a man* belongs to the subordinate universe.

Universally quantified sentences such as in example (104), are treated very similarly to conditional sentences.

(104) *Every man who meets a nice woman*^{1} *smiles at her*$_1$.

The DRS for example (104) can be seen in figure 4.4.

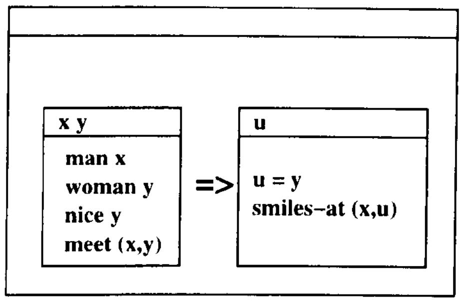

Figure 4.4: DRS for a universally qualified sentence

Negations are dealt with in DRT by 'negating' a subordinate DRS; see figure 4.5 which shows the DRS for example (105).

(105) *Someone did not smile.*

We will now shortly describe how DRSs are merged in order to update an existing DRS. If we have a DRS for the first sentence of example (106) and a DRS for the second example (both shown in figure 4.6) then the problem is that the discourse marker x occurs twice and needs to renamed. After the renaming takes place, both the universes and the conditions sets are merged and the resulting DRS is also shown in figure 4.6. This is only a very short

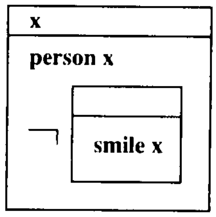

Figure 4.5: DRS for a negated sentence

description of the most basic case of merging, for more details see Kamp and Reyle (1993); van Eijck and Kamp (1997).

(106) *The man entered. The woman smiled.*

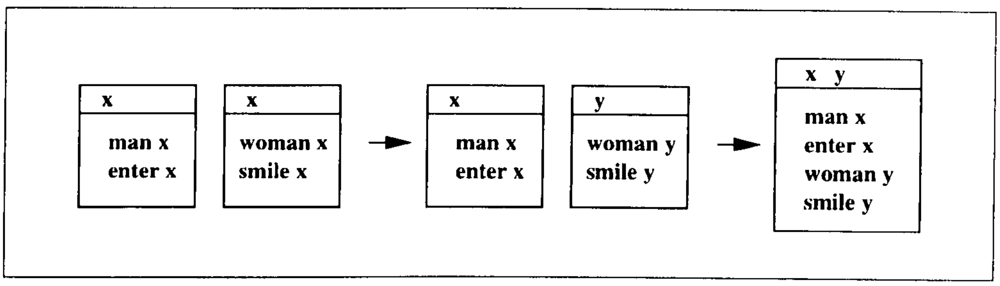

Figure 4.6: Merging two DRSs

DRT has been critisised for not being 'compositional'. This can be taken in two ways. The first is that DRT fails to provide the means for direct compositional semantics for the natural language fragments it is applied to. For the original formulation of DRT (Kamp and Reyle, 1993) this criticism is justified and this has been addressed through the dynamic reformulation of DRT in van Eijck and Kamp (1997). The second criticism pertains to the formalism of DRT itself, an objection which is groundless (according to the authors) because DRT is as compositional as standard predicate logic.

Tense and aspect have also been addressed, however briefly, within DRT. Verbs are given an event argument ($smile(e_2, x)$) and conditions for the temporal interval during which the event takes place ($t(e_2)$) are introduced into the condition set. An element n (*now*) refers to the time of utterance. The set of temporal intervals is ordered by precedence $<$ and temporal inclusion $\sqsubseteq$. Figure 4.7 shows a DRS for example (107). The events in example (107) are naturally understood as sequential with the entering preceding the smiling.

(107) *A man entered the White Hart. He smiled.*

x y z e1 e2
man x
y = White Hart
enter(e1, x, y)
t(e1) < n
z = x
smile(e2, z)
t(e1) < t(e2)
t(e2) < n

Figure 4.7: Temporal reference

The order in which sentences appear is only one of several factors which determine the temporal relation. ***Aspect*** is another such factor, for example changing the non-progressive *smiled* in example (107) to the progressive *was smiling* would yield a different temporal interpretation. Stative verbs like *like* also give rise to similar interpretations. Rhetorical relations between sentences, ***explanation*** for example, carry certain implications for the temporal order as well. For instance, what is explained has to happen before the explanation.

The central idea behind the treatment of time in DRT goes back to Reichenbach (1947) and the interpretation of times involves relating events to a reference point. The theory is, at the moment, limited to sentences where the immediately preceding sentence supplies the reference point. How this reference point is used depends on whether the sentence is of a stative or non-stative character (i.e. whether it is a state or an event).

The treatment of temporal reference in van Eijck and Kamp (1997) is seen as being a hint of what a fully fledged account of tense and aspect for a language like English might be.

4.2 Representing Time in Natural Language

Ter Meulen's book comprises three important parts: a systematic and detailed account of how temporal information in texts or discourses is used to reason about the flow of time; a powerful dynamic representation system (dynamic because it changes while a text or discourse is processed); and a 'semantic toolkit to get from one to the other'. This toolkit contains algorithms and rules for the core ideas of dynamic interpretation and is intended as a starting point for computational interpretation.

The choice of verbal tense and aspect, temporal adverbials, prior context and general world knowledge all play an important role in determining how events described in a text are related in time. Temporal adverbials explicitly describe such temporal relations. Prior context obviously matters to interpretation and ter Meulen requires that semantic rules reflect how a sentence expresses different temporal information in different contexts. This is fundamental for her study of situated inferences. General world knowledge can also be of importance but is abandoned because "a semantic theory for natural language should not have to venture into the realm of physics or common sense knowledge about causal structure" (ter Meulen, 1995, pg. 3). Ter Meulen concentrates on verbal tense and aspect, as they are important indicators of how described events are temporally related.

Verbal tense only partly indicates in which order the events took place, only puts a weak constraint on the temporal relations. The information given by aspect is far more crucial and will be reviewed later in this section.

Dynamic aspect trees (DATs) are used to represent temporal dependencies and they are also the means of situated inferences. We will come back to DATs and reasoning with DATs later in this section.

Aspectual Classes Ter Meulen's view on aspectual classes is different from the classical Vendlerian one but is based on the same four aspectual classes that were introduced in section 2.1.1. The focus is on the flow of information, on how change is described and encoded in text, and on how this flow is controlled by aspectual classes.

Ter Meulen first of all distinguishes between stative information and three modes of active flow control. We will first describe active flow control and then stative information later on.

Ter Meulen labels the aspectual classes in a more mnemonic way, so *Holes* for example correspond to activities. If a sentence is interpreted as describing a *hole*, then the next sentence is interpreted as describing a temporal part of the activity, "as if the information it conveys flows through a *hole*" (ter Meulen, 1995, pg. 7). This means, that the starting point of the *hole* must precede the starting point of the next event. But the next event can go on longer than the *hole*.

(108) *Water spread over the floor. The children stood there watching.*

Filters correspond to accomplishments. They create a choice for the next sentence to be interpreted as either describing a later event, or an event temporally included in the *filter*, or happening simultaneously. An example for an event being included is:

(109) *Mary climbed Mont Blanc. She walked up to the base camp.*

The start of a *filter* must precede the start of the event(s) described in the next clause(s), but it finishes after these events finish. *Holes* give similar information, for example: *Jane drank some water. She ate some chips.* The only information available is that the *hole* starts before the event described in the second sentence starts, nothing is known about when the events finish.

Plugs correspond to achievements. If a sentence is interpreted as describing a *plug*, information about what happened at the same time is blocked, the temporal focus is redirected and subsequent sentences are interpreted as describing later events. Events described as *plugs* are 'atomic', conceptually instantaneous and the internal structure is not accessible for further information. Example:

(110) *Jane arrived. She went straight to her room.*

Everything said about *holes, plugs* and *filters* was said disregarding the influence of context and background. Either of those can override general semantic principles.

DATs The information about events in an episode is represented in *dynamic aspect trees* (DATs). They will be explained by means of example (111); the DAT is given in figure 4.8.

(111) *Jane felt ill. She sat down to decipher the message, and looked at her watch. She sighed. It was not even noon yet. (ter Meulen, 1995)*

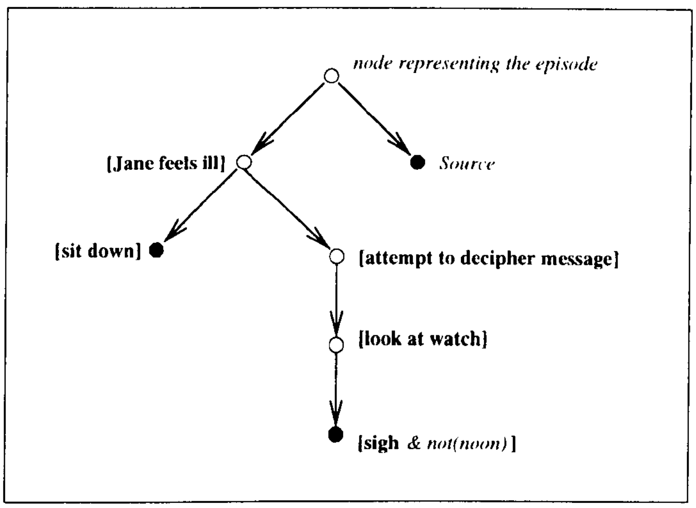

Figure 4.8: DAT for example (111)

The event of issuing information (orally or otherwise) is represented within the DAT by a node uniquely labelled *source*. This is always the rightmost node, only dominated by the unique root node representing the entire evaluated discourse or text.

Since we are dealing only with past and perfect tenses, all events precede the utterance and are represented by nodes branching to the left. Downward arrows indicate temporal inclusion of events and the left-to-right order between paths of labelled nodes (also called *chronoscopes*) represent the flow of time.

In the example, *Jane felt ill* is a *hole* (represented by an open node) and all the following events are temporally included; they are represented by daughter nodes. *She sat down* is a *plug* (represented by a closed node) and the following *She attempts to decipher the message* is represented by a sister node. Now we have a choice in interpreting the attempt as a *hole* or as a *plug*, depending on whether Jane's sighing is interpreted as happening after she looked at her watch or while she is looking at her watch. Two different DATs would be needed for the two interpretations, the DAT shown in figure 4.8 shows the 'hole' interpretation.

The stative information *It was not even noon yet* is not represented by a node but by a so-called *sticker* attached to the *current* node (the last constructed node). In fact, only simple past tense clauses describing events introduce new nodes and make the DAT grow, all stative information is represented by stickers.

More information, including information about the syntax and semantics of DATs and about rules for representing aspectual classes and updating DATs, can be found in ter Meulen (1995).

Reasoning with DATs A short example should give an impression of how reasoning with DATs works. The example, two inferences (where * marks an invalid inference) and the DAT (figure 4.9) follow, and can be found in (ter Meulen, 1995, pg. 52).

(112) *The car hit the fence. The driver was killed. The police arrived.*

(113) a. *The car hit the fence before the driver was killed.*

b. * *The car hit the fence after the driver was killed.*

The temporal adverbial *before* requires that the node representing the car hitting the fence is to the left of its sister node representing the driver getting killed. Similarly, the adverbial *after* would require the two nodes the other way round, the 'killing' node to the

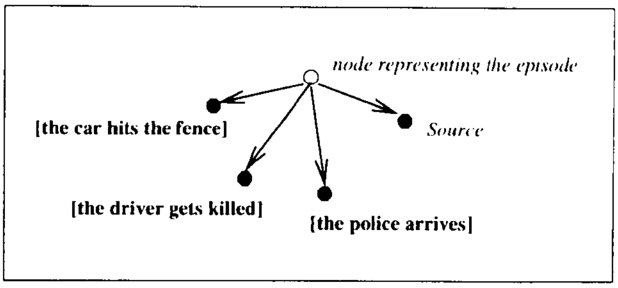

Figure 4.9: DAT for example *The car hit the fence ...*

left of the 'hitting' node. Since this is not the case, *The car hit the fence after the driver was killed* is an invalid inference.

When natural language is processed, the information is accumulated and inferences we draw are based on this accumulation. DATs provide us with a representation of the accumulated information and they facilitate simple search algorithms, e.g. for verifying a perfect tense conclusion (p. 54 in ter Meulen's book).

4.3 Representing Temporal Information Using Conceptual Graphs

Moulin (1997) extends Sowa's 'Conceptual Graph Theory' (Sowa, 1984) in order to represent semantic and pragmatic mechanisms used in expressing time in natural language. Although conceptual Graphs (CGs) provide the means to represent not only complex sentences with relative clauses but also anaphora and indexicals, they lack structures for representing verb tenses. Nor can they characterise words reported or represent temporal references (such as *last year* or *two days before he arrived*).

"In order to interpret deictic forms, it is necessary to know who the speaker and hearer are, and the time and place where the speech act was performed. Verb tenses are also used and understood with respect by speakers with respect to their temporal location and to the temporal localisation of situations referenced by the speech acts that are performed. To represent these pragmatic properties of discourse, we need to introduce the context of utterance of speech acts and references to speakers who perform them." (Moulin, 1997)

The approach described in (Moulin, 1997) addresses exactly these issues. We will follow Moulin's structure and first give a very brief overview of Sowa's Conceptual Graph Theory, then describe Moulin's own approach.

4.3.1 Conceptual Graph Theory

A *conceptual graph* (CG) is a semantic network which contains *concepts* and *conceptual relations*. It is a finite, bipartite and connected graph and the nodes are either *concepts* or *conceptual relations*. *Concepts* represent objects of the application domain, which have a referent like *John*, *Mary* or *apple* and a type like *human* or *animal*. *Concepts* are specified between square brackets in the linear notation or as boxes in the graphical representation. *Conceptual relations* are elementary links between *concepts*, usually binary. They often correspond to semantic cases like AGENT or PATIENT and are represented between brackets in the linear notation, or as circles in the graphical representation. Figure 4.10 shows both the linear and the graphical representation of a CG for example (114).

(114) *John buys an apple.*

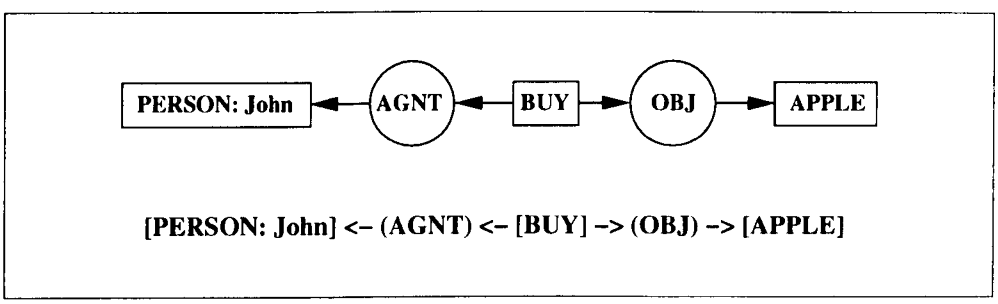

Figure 4.10: Conceptual graph I

Sowa provides an elegant way of representing anaphoric references. They are denoted by a $*$ following a variable name and co-referring *concepts* have the same variable name. In example (115), both *John* and *he* are co-referring and *apple* and *it*. Figure 4.11 shows the corresponding CG.

(115) *John buys an apple and he gives it to Mary.*

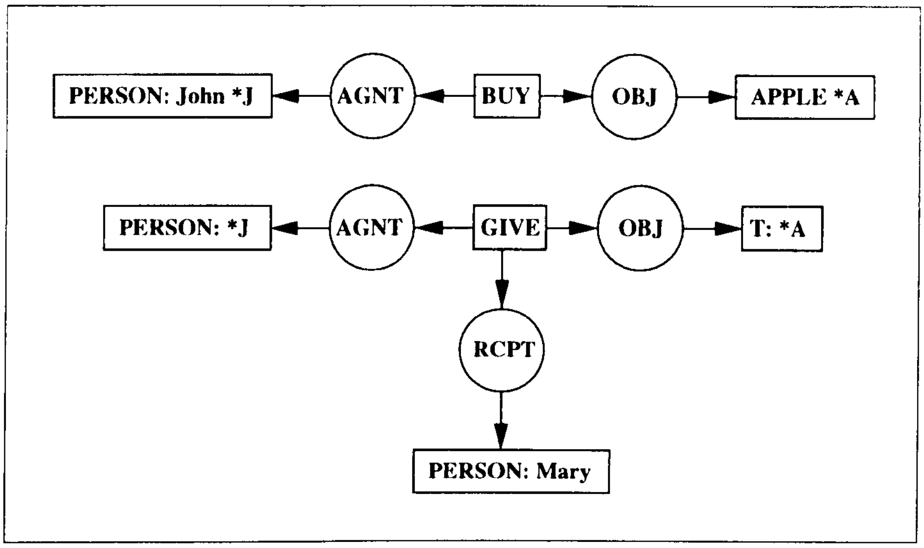

Figure 4.11: Conceptual graph II

Indexicals are denoted by using the symbol # but Sowa did not provide the means to specify the correspondence between indexicals and the ***concepts*** they refer to. A sentence like *I love this apple* can be expressed as:

(116) [PERSON: #I] <- (AGNT) <- [LOVE] -> (OBJ) -> [APPLE: #This]

Sowa introduces so-called ***proposition nodes*** to represent more complex sentences, like examples (117) and (118) and figure 4.12.

(117) *Every farmer who owns a donkey beats it.*

(118) *John thinks that Mary is ill.*

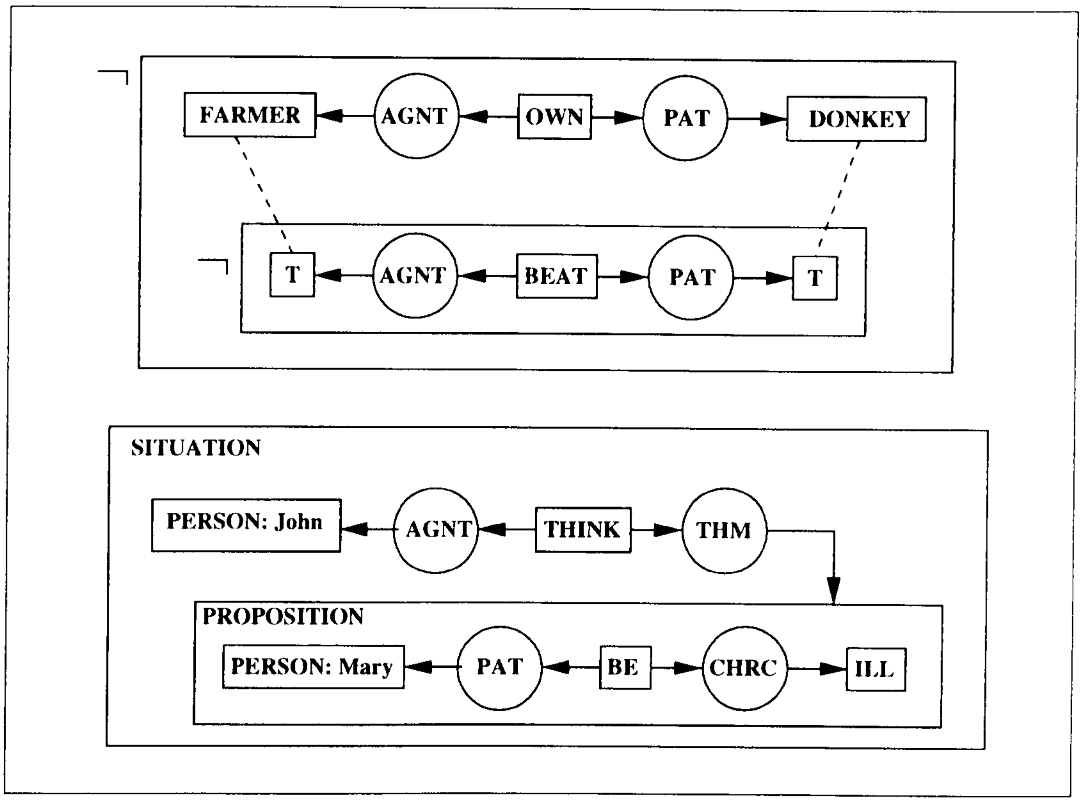

Figure 4.12: Conceptual graphs III

Sowa defines basic and more advanced operators for *conceptual graphs*. The *copy* operator makes an exact copy of a given graph, the *join* operator takes two existing graphs that have a common concept and produces a new graph. The *maximal join* creates a new graph from two existing ones, u and v. The new graph contains all the *concepts* and *relations* from u and v after suppressing duplicates and specialising the appropriate *concepts*. *Generalisation* is similar, but the new graph is more general than u and v. This operation is useful for discovering new *concepts*.

Moulin raises several problems when using Sowa's approach to represent temporal knowledge.

- *Concepts* and *relations* to describe the properties of time points exist, but they are not applicable to time intervals.
- The *relations* for past and future are not precise enough or they are ambiguous. It is

not possible to distinguish between *John opened the door* <u>*and*</u> *called Mary* and *John opened the door* <u>*when*</u> *he called Mary*. Aspectual properties play an important role determining the temporal properties of a situation. The time intervals during which the situations take place need to be considered in order to describe the aspectual properties of these situations.

- A formalism is needed to represent temporal knowledge independently from the surface form of the sentence expressing this knowledge.

4.3.2 Moulin's Framework

To address these problems, Moulin developed a framework to provide a semantic representation. He distinguishes two levels of representation of the information contained in a discourse:

1. The ***conceptual level*** describes the temporal entities (objects, situations, perspectives etc.) and the temporal relations describing the situations in a discourse.
2. The ***linguistic level*** contains all the linguistic information needed to represent the speech acts which produced the discourse.

The ontology of the conceptual level assumes the world is "composed of temporal situations in which agents act physically, mentally or illocutionarily (through the performance of speech acts)." (Moulin, 1997)

Time interval, temporal situation, temporal object and temporal relation are the basic notions on which the temporal framework is based. A ***temporal situation*** is associated with a ***time interval*** which denotes its temporal location on, for example, a time axis. A temporal situation can be an event, a state or a process. ***Temporal relations*** relate time intervals which are associated with temporal situations. A ***temporal object*** is used to refer to a temporal situation as a whole (as opposed to the semantic description described by the temporal situation) and is thus associated with a time interval. Graphically, a temporal situation is represented by a rectangle comprising two parts. The rectangle itself represents the time interval, the upper part contains the situation description (COMPLETED PROCESS for example (119)) and the parameters of the situation time interval. The lower part represents the propositional content (using CGs). Figure 4.13 shows the graphical representation of the temporal situation described in example (119).

(119) *John stayed in Toronto for two weeks from January 1st until January 16 1995.*

COMPLETED-PROCESS:cp1 ;BT:Jan 1 1995; ET: Jan 16 1995; TS: date; DU: 2; DS: week
[PERSON: John] <- (AGNT) <- [STAY] -> (LOC) -> [CITY:Toronto]

Figure 4.13: Temporal Situation for example (119)

We will now briefly describe the basic notions mentioned above. Figure 4.14 shows the graphical representation of the temporal situations described in example (120) and associated objects.

(120) *The day that John arrived in Toronto*

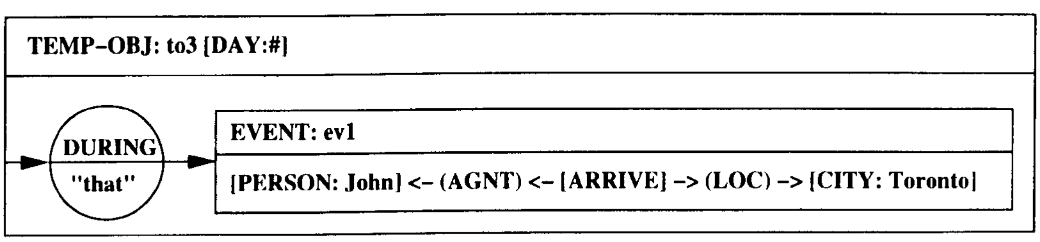

Figure 4.14: Temporal Object for example (120)

The situation type can be an *event* (denoting a change or discontinuity with respect to a static background), a *state* (a state of affairs persisting with no change) or a *process* (representing a change from an initial static situation to a final static situation). The full taxonomy can be seen in (Moulin, 1997, pg. 237). The *temporal relations* used are a slightly extended form of Allen (1983)'s relations. They can be expressed in natural language using temporal conjunctions such as *while* or *when*. Graphically, *temporal relations* are represented by a circle comprising two parts. The upper part contains the temporal relation type and the lower part contains relevant parameters (when instantiated in the discourse). For example, the linguistic term expressing the lap parameter *2 days* for the relation *AFTER* can be a relevant parameter. Rectangles represent temporal situations as well as the associated *time intervals*. *Temporal concepts*, such as *day*, *week* or *year*, are characterised by a *time interval* and can be related by *temporal relations*. Thus temporal reasoning can be carried out on the time intervals associated with temporal objects and temporal situations. *Temporal concepts* are represented graphically by a rectangle which represents the associated time interval and is composed of two parts. The upper part contains the object description as well as relevant parameters. The lower part contains the object propositional content, which is the temporal situation or object that is related to the embedding object by a relation. In figure 4.14, the temporal object *the day* is related to the situation *John arrives in Toronto* by the relation *during*.

The narrator sets the *main time coordinate system* in a discourse, which sets a temporal reference point for localising the temporal situations or objects. The narrative is told from the narrator's perspective. When reporting words from another agent, this perspective changes to that of the other agent. This is reflected by the fact that the verb tense changes in order to accommodate the new perspective. When speakers refer to specific points in time using dates or indexicals like *last year* or *tomorrow* then they are referring to a *secondary time coordinate system* positioned relative to temporal perspective. Moulin introduces graphical representation to include these time coordinate systems, we omit details here for reasons of brevity. See Moulin (1997) for more detail.

Moulin interprets a sentence not only from the point of view of temporal structure but also as the result of an illocutionary act performed by an agent communicating with other agents. Each temporal structure is augmented with parameters describing the illocutionary act, such as *verb mode* (indicative or subjunctive, for example) or *illocutionary point* (directive or assertive etc.). In example (121), *Mary told John* is explicitly stated in the text and the expression style is direct. *Send a letter to your parents* is the directive illocutionary act performed by Mary, the verb mode being imperative. Figure 4.15 shows the graphical representation.

(121) *Mary told John: "Send a letter to your parents!".*

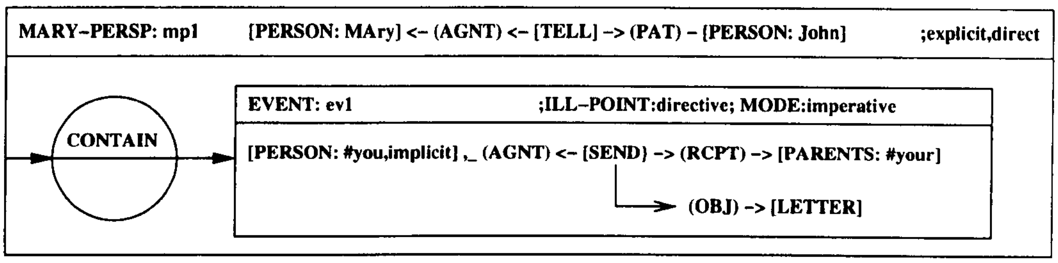

Figure 4.15: Temporal situation for example (121)

4.4 Discussion

The goal motivating the current research is to identify events in newswire texts and to establish their relative order or, where possible, their calendrical date (see section 1.1). As we have mentioned in the beginning of this chapter, being able to draw inferences based on the temporal information explicitly expressed in text, is important part of any system aimed at temporally locating events in time.

The formal discourse approaches, which were introduced in the previous sections, provide the means for the inferences needed. If we could process the texts and derive one of the three representations described, then we are likely to be able to use the framework provided to infer the relative order between events or their position in time.

But the following problems arise:

1. The cost of arriving at an appropriate representation for the formal approaches is high. Both Moulin and DRT require a full semantic interpretation of *all* the context as well as a syntactic analysis of the text. We belief that relatively little understanding is required for the task of temporally locating events, so we are hoping to avoid a full semantic and syntactic analysis and to concentrate on the temporal information itself.

2. Both DRT and ter Meulen focus on tense and aspect as their main source for establishing temporal relations. While tense and aspect do occur in newswire texts, they play a larger role in stories – the genre DRT and ter Meulen's approach seem to be geared towards (or at least this is the genre most of their examples are taken from). Hence, DRT and ter Meulen might be 'overkill' for newswire texts.

3. None of the three approaches attempt to explore or catalogue the set of linguistic expressions with which temporal information is conveyed. But identifying how temporal information is conveyed is an important prerequisite, both for arriving at a representation suitable for the formal approaches, and for any system aimed at extracting temporal information. It is necessary to identify what to extract or represent before one can progress to do so.

Would it be possible to semantically label real text instead of interpreting the text and translating it into a different, abstract representation? We will explore this possibility in the remainder of this thesis. A corpus-driven approach will be adopted, and we will describe such corpus-driven approaches in the following chapter.

Chapter 5

Corpus-Driven Approaches

Annotated Corpora can be used in a variety of ways. For example, empirical linguists use them to analyse phenomena of concern; corpus linguists employ them in training and evaluating algorithms in automated systems.

This chapter discusses the research that only recently has begun to tackle temporal phenomena in real text. This chapter describes the work that has been undertaken in this field as well as giving a brief overview of corpus-annotation in general. The approaches can be broadly classified into thee areas: annotating temporal referring expressions, annotating events with time stamps, and annotating temporal relations. A section in this chapter is devoted to each of them.

The work introduced in this chapter is closely related to the approach we have developed, and which is described from chapter 6 onwards. An evaluation of the work described here with respect to our work can be found in section 5.5.

5.1 Corpora and Corpus-Annotation as a Methodology

Corpus Annotation is a research field within Computational Linguistics which has become eminent over the last two decades. Traditionally, a *corpus* refers to a body of naturally occurring language data (written text, spoken discourses etc.), used as a basis for linguistic research. This has shifted and nowadays the term *corpus* is used to designate a body of text which exists in electronic form and which can be processed by a computer, used as part of linguistic research and language processing.

There are at least four criteria which determine the value of a corpus. The sheer size, the diversity of text types contained and the care with which it has been produced are three of the criteria. A fourth, which is the focus of this section, is the degree to which 'added value' is brought to a corpus by annotation (Garside, Leech, and McEnery, 1997), which is more explicit knowledge about the text with which the corpus is 'labelled'.

Corpus annotation can refer to the practice of adding information to an electronic corpus (spoken or written data) or to the end product – the information attached to the electronic representation. Grammatical or Part-Of-Speech (POS) tagging is a very common form of corpus annotation, where a label or *tag* is associated with a word to indicate its grammatical class. An example is *taken*_VVN where the grammatical tag VVN shows that *taken* is a past participle.

Annotation is usually interpretative and (meta-)linguistic. Interpretative, because annotation depends to some degree on human understanding of the text. For example, is the word *future* in *his future bride* a noun or an adjective? How much detail (also called *granularity* or *delicacy*) should be added? Should *future* be labelled as being an adjective or as being an adjective that can only occur in a pre-nominal position? Annotation is *meta-linguistic* in the sense that it provides us with information *about* the language instead of telling is what the text itself comprises (the text itself, the 'pure' un-annotated corpus is called the *raw corpus*). This distinction is easily made for written text but is more difficult for transcribed spoken dialogue, for example, where the transcriber necessarily interprets the discourse in the process of transcribing.

Corpus annotation is important for several reasons:

1. *Extracting Information*

   The usefulness of corpora lies in the fact that we can extract knowledge or information from them. But the *raw corpus* contains no direct information about, for example, grammar. We often have to start with adding information before we can extract information.

2. *Re-usability*

   Corpus annotation is a time consuming and expensive task and an annotated corpus is a valuable re-usable resource.

3. *Multi-functionality*

   Annotation often has different purposes and applications. Grammatical tagging, for example, can be used for lexicography (the word *left* can be a noun, verb, adjective or adverb and the POS tag indicates its class) and speech synthesis (the word *lead* is pronounced different, depending on whether it is a noun or a verb). Grammatical tagging is also often seen as the first step towards more difficult levels of annotation, like syntactic and semantic annotation.

An annotated corpus has to adhere to the following standards to make it re-usable and valuable (Garside et al., 1997).

1. The raw corpus should be *recoverable*.
2. The annotations should therefore be *extricable*, so they can be stored independently.
3. *Documentation* should be available on:
   (a) the annotation scheme;
   (b) how the annotation process was carried out and by whom; and
   (c) the quality of the annotation (to what extend has the corpus been checked, how accurate is it, etc.).

An annotation scheme does not present 'God's truth' but is offered on a basis of practical usefulness. It is therefore advisable to base annotation schemes on a *consensual* or *theory neutral* analysis of the data. There is some movement towards some kind of standardisation of corpus annotation practices (for example the EAGLES initiative within the European Union (Ide and Verónis, 1995; Leech and Wilson, 1994)), which aids re-usability.

Different levels of annotation have been applied to different corpora. The overview in table 5.1 is taken from Garside et al. (1997), to which the reader is referred for more information on each type.

Linguistic level	Annotation carried out so far	Chapter in Garside et al. (1997)
Orthographic	This is generally considered part of 'mark up'	chapter 1
Phonetic/ phonemic	Widespread in speech science – but typically collected in laboratory situations	chapter 1
Prosodic	Two or three prosodically annotated corpora are available for widespread use	chapter 1
Part of speech (i.e. grammatical tagging)	The most widespread type of corpus annotation, which has been applied to many languages	chapter 2
Syntactic, i.e. (partial) parsing	This is the second widespread corpus annotation, and is rapidly developing	chapter 3
Semantic	Some exists, and more is developing	chapter 4
Discoursal	Little exists – but some is developing	chapter 5
Pragmatic/ Stylistic	(As for discoursal annotation)	chapter 1

Table 5.1: Levels of annotation (from Garside et al., 1997)

5.2 Annotating Time Referring Expressions

The most extensive work on annotating time referring expressions so far has been done during the message understanding conferences – briefly described in section 5.2.1 – or is an extension this work – introduced in section 5.2.2.

5.2.1 MUC Named Entity Time Task

In the mid 1980's, the first of seven message understanding conferences, sponsored by DARPA, was held. The aim of these conferences was to establish a quantitative evaluation regime for information extraction (IE) and message understanding (MU) systems (the term MU has disappeared now in favour of the term IE). The last MUC, MUC-7, was held in 1998. For more information about the message understanding conferences see Gaizauskas and Wilks (1998); MUC6 and MUC7.

There were typically several tasks that had to be carried out by the participating systems. The task relevant for this thesis is the ***named entity recognition*** task, which required the recognition and classification of defined named entities such as persons, locations, organisations, monetary amounts and dates.

From MUC-5 onwards, dates were part of the named entity task. Absolute and relative time expressions conveying dates and times had to be marked up (although those two types were not distinguished in the annotation). Time expressions were classified via the TYPE attribute. Type DATE referred to complete or partial date expressions and type TIME referred to complete or partial expressions of time of day.

The time expressions were not further evaluated, and only in MUC-5 and MUC-7 were the time expressions related to a scenario template.

5.2.2 An Annotation Scheme for Temporal Expressions

Wilson, Mani, Sundheim, and Ferro (2001) describe a set of guidelines for annotating time expressions and associating a canonical representation of the times they refer to with them. A method for extracting such time expressions from multiple languages is also introduced. The two main novel features are:

1. The approach goes well beyond the annotation guidelines used in the Message Understanding Conferences (MUC7). The range of expressions flagged is much wider and, more importantly, the time values communicated by the time expressions have a richer representation. The key difference is that in MUC, the time expressions were only annotated whereas Wilson et al. (2001) evaluate the expressions and normalise the representation for the times referred to.

2. Context-dependent time expressions like *today* are handled in addition to fully specified time expressions like *September 3rd, 1997.* Context can be local (within the same sentence) or global (outside the sentence). Indexical time expressions, that require knowledge about the time of speech, like *now* are also included. A corpus study (Wilson and Mani, 2000) showed that two-thirds of time expressions in print and broadcast news are context dependent, so this feature is significant.

The following criteria are addressed by the annotation scheme:

- *Simplicity with Precision:* The scheme is simple enough to be applied by humans but also precise enough to be used in various natural language processing tasks.

- *Naturalness:* The distinctions a human could be expected to reliably annotate are reflected rather then what an automated system could do.

- *Expressiveness:* Time values are specified as fully as possible within the bounds of what can be confidently inferred by annotators.

- *Reproducibility:* The format[1] for representing time values is used and consistency among annotators ensured by using an example-based approach.

[1]Available at ftp://ftp.qsl.net/pub/g1smd/8601v03.pdf.

Annotation takes place in two steps. Firstly, all temporal expressions are flagged and secondly, a time value is identified for these expressions. Three different kinds of time values are represented: points, durations and frequencies. Problems like fuzzy boundaries (as in *the 1960s*) and non-specificity (as in *April is usually wet*) are addressed as well.

Although developed in English, the semantic representation used is intended for use across languages and thus aids the development of temporal taggers for different languages based on a common annotation scheme. An English reference corpus consisting of 32,000 words of a telephone dialogue corpus (English translations of the 'Enthusiast' corpus used at CMU and by Wiebe et al. (1998)), 35,000 words of New York Times newspaper text and 120,000 words of broadcast news has been created.

5.3 Annotating Events with Time Stamps

5.3.1 MUC-5 and MUC-7 Time Slots

As already mentioned in section 5.2.1, MUC-5 and MUC-7 established a relation between times and events. Participants were required to assign a calendrical time to certain specified event types (joint venture announcements and rocket launchings, respectively).

Scenario template filling requires the identification of specific relations holding between template elements. For example, the MUC-7 scenario template filling task concerned rocket launch events. The scenario template contains information about vehicles, pay load, launch site, mission function etc. It also contained a field called LAUNCH_DATE, which contained a link to the appropriate time entity (if the launch event could be associated with a time).

The temporal relation between the time and the event was not further evaluated. Temporal relations between events and other events are not addressed.

5.3.2 Assigning Time-Stamps to Event Clauses

In Filatova and Hovy (2001), a method for breaking news stories into their constituent events and assigning time-stamps to them is described. The authors analyse implicit time reference (mainly the tense system) and explicit time reference (temporal adverbials) to aid in reconstructing the chronological order of the events in the news story or stories.

Simple clauses are assumed to be the text conveying events and they are identified using a parser which produces semantically labelled syntactic parse trees. Some problems are ignored in this approach, for example multiple verbs with different tenses in one sentence cause incorrect behaviour of the system. The time representation chosen is a slightly modified version of the one suggested in Allen and Ferguson (1994).

Time-stamping uses two time-points for anchoring. One time-point is the time of the article (at the moment only the date is used and the time of day is not taken into account) and the other time-point is the last time-point assigned within the same sentence. The procedure of time-stamping is as follows:

1. The date-stamps are extracted (presumably a calendar date is assigned to explicit time references).

2. A time is assigned to each event (represented by simple clauses), either the most recently assigned date in the same sentence or, if this is not defined, then the date of the article. Two kinds of selection rules are used when deciding which time is assigned. One kind is for sentences with explicit time references and one kind is for sentences without explicit time references. An example for the former is the *Day of the week* rule which states that if the day of the week used in the simple clause is the same as the day of the week of the article date then the date of the article is the time-point assigned to the event (there may be exceptions, for example if there are words before the day of the week signalling that the event happened in fact before the date of the article then a different time-point is assigned to the event). If the sentence does not contain explicit date information then a time is assigned depending

on the tense of the verb. Present perfect and past perfect event clauses are assigned an open-ended interval whose starting point is unknown and whose endpoint is either the most recently assigned date or the date of the article. Similar rules apply to future, present and past indefinite clauses.

After all events have been stamped with a time, the event clauses are arranged in chronological order. The authors report 60.76% recall and 55.82% precision, averaged over the texts of the small trial corpus they have created.

5.4 Annotating Temporal Relations

5.4.1 Annotation of Intrasentential Temporal Information

Katz and Arosio (2001) aim to create a large multi-lingual corpus, in which intrasentential temporal relations are tagged in addition to standard morphological and syntactic annotation. To aid this, they have developed a language-neutral and theory-neutral method for annotating sentence internal temporal relations. With this corpus, Katz and Arosio (2001) hope to be able to automatically acquire the lexical knowledge required for determining temporal interpretation in narrative discourse (see Lascarides and Asher, 1993; Reyle and Rossdeutscher, 2000). Automatic induction from large scale corpora is assumed to be the most promising approach to achieve this (Rooth et al., 1998).

A temporal interval is associated with each verb in the sentence and the temporal relations between those verbs are of concern. The temporal interpretation should be closely linked to the syntactic context (which is of importance since it is not known beforehand to what degree the cues used by the speaker are lexical and to what degree they are grammatical). This linking is needed to keep track of both the semantic relations among times as well as the syntactic relations among the words in the sentences that refer to these times.

The authors have adapted a syntactic annotation tool developed at the University of Saarbrücken (Brants and Plaehn, 2000) and added a layer of semantic annotation. The verbs in the sentence are linked via secondary edges labelled with a temporal relation. Precedence and inclusion and their duals are the possible relations. Indexical information is included by introducing the symbol ∘ for the speech time, which is automatically prefaced to all sentences prior to annotation. Figure 5.1 illustrates their approach.

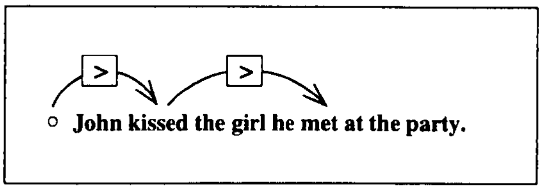

Figure 5.1: Example for temporal information within a sentence

The annotation procedure begins with a syntactically annotated tree bank. Then the speech time marker is added and the temporal relations among verbs are annotated (as the annotator naturally understands them).

A method to compare temporal information based on a model-theoretic interpretation is described in Katz and Arosio (2001) as well. A searchable multi-language annotated treebank has been created where each sentence is stored in a relational database with both syntactic and temporal annotations. This makes is possible to query the corpus ("Find the sentences containing a relative clause which is interpreted as temporally overlapping the main clause" (Katz and Arosio, 2001)).

5.4.2 Semantic Tagging of Newswire Texts

Schilder and Habel (2001) have developed a semantic tagging system for temporal expressions in newswire articles. In their terminology, temporal expressions are chunks of text that express direct or inferred temporal information, such as dates, prepositional phrases and verbs. The main part of their system is a Finite State Transducer (FST) based on handwritten rules. Their target language is German.

The ultimate goal is to establish the temporal relations between all events in the text.

The current state is to anchor temporal expressions on the absolute time line, although the semantics of these expressions are not fully developed yet.

Temporal expressions are classified into ***time-denoting expressions*** that refer to a calendar or clock time and ***event-denoting expressions*** which refer to events. Prepositional phrases like *on Friday* denote temporal relations between events and times. The set of temporal relations proposed is ***before, after, incl, at, starts, finishes*** and ***excl*** (equivalent to Allen (1983)'s relations).

The FST tags all time-denoting expressions, all verbs and an experimental version tags nominal expressions. A semantic representation is then proposed, based on which inferences are drawn, especially about temporal relations. In its current state, the FST establishes temporal relations between times and events. The tagger was evaluated with respect to a small corpus (10 news articles) and an overall precision rate of 84.49% was achieved.

5.5 Discussion

Wilson et al. (2001) present a very thorough annotation scheme for time expressions. However, they do not include temporal relations of any kind (between events and events or events and times) in their approach.

Katz and Arosio (2001) focus on intrasentential annotation. They annotate temporal relations (inclusion and precedence) between events (conveyed by verbs) and have started to create a multi-lingual corpus. They do not identify how the temporal relations are expressed in text. but they hope to gain insight into the mechanisms used from the corpus.

Both Schilder and Habel (2001) and Filatova and Hovy (2001) have a similar aim in that they both want to locate every event in a newspaper article in time, but they differ significantly in their approach. Filatova and Hovy (2001) want to assign a time stamp, an actual calendar date, to every event. Events are defined as clauses, and the possibility of, for example, nominalisations conveying events is not taken into account. Some of the time stamps assigned are fairly crude (for example an interval whose start time is unknown and whose end time is the date of the article) and temporal relations between events are ignored. We believe, however that the relations holding between events are of importance, especially when we cannot establish a better calendar date then 'at some point before the article'.

Schilder and Habel (2001) present a very promising approach, aimed at German newspaper articles. The work is at an early stage, establishing the temporal relations between events and times only. It would have been interesting to see the rules on which the FST bases the tagging of the temporal expressions and to see the inference rules used.

In the remainder of this thesis, we will describe an annotation scheme developed to capture temporal information in text. The scheme differs from the related work described above, the most important points being the following.

- The temporal relations holding between events and events as well as between events and times in multi sentence texts are annotated. The relations can be intra- as well as intersentential. Only Katz and Arosio (2001) address event-event relations, but concentrate on intrasentential relations only.

- A wide variety of mechanisms used to convey temporal information are identified and annotated. Filatova and Hovy (2001), for example, do not take into account that a significant portion of events are expressed using non-finite verb clauses and nominalisations.

- An interactive inference engine is used to derive further temporal relations which are not necessarily anchored in the text.

Chapter 6

A Conceptual Framework for Temporal Annotation

In this chapter, the conceptual framework that is the basis of the annotation scheme will be described in detail.

The previous chapters were dedicated to related work and background information regarding the goal of ordering events in time. We have described different ways of conceptualising and classifying events, time and temporal relations as well as how these entities can be expressed in natural language. We have talked about different formal discourse approaches and corpus-driven approaches that are relevant to our goal.

We have already made clear, that our approach is a corpus-driven approach (see chapter 4). The background presents a range of choices to us (as described in chapters 2 and 4, and the decisions we made about these will be made clear as the conceptual framework is being introduced.

6.1 The Framework

Before an annotation scheme for temporal information can be proposed we must make clear the sorts of temporal entities and relations we suppose exist. Much has been written concerning the appropriate temporal ontology and set of temporal relations for analysing temporal phenomena in natural language (see chapter 2). However, our goal here is not to arrive at some indisputably 'true' description of temporal reality. Rather, we wish to provide a framework that can be used in classifying expressions in real texts in a fashion that enables us to gain useful insights into how temporal information is conveyed in written language. Ultimately the aim is to develop an algorithm which can identify events in newswire texts and determine their temporal order or position in calendrical time well enough to answer questions about the ordering or times of events at the level we would expect of an average human reader, questions like *When did the TW crash happen? What happened at the same time?*.

Given this perspective, we may summarise our conceptual or descriptive framework quite simply. It presumes the world contains the following primitive types: events, states, times, and temporal relations – each of these is discussed in detail below. Of course this framework is not complete, but we believe it provides a useful starting point.

Although our approach has been shaped by the genre of newswire articles, we believe that the base approach (described in sections 6.2 to 6.5) is applicable to almost all genres and domains. On top of this base approach, several extensions are possible. These can be genre or domain or even application dependent. Section 6.6 discusses some extensions we have integrated into our approach and some possible extensions that we chose not to integrate (and the reasons for doing this).

When we look at newswire texts, then we see that events and states are described and the centre of interest, not time itself. This has lead us to develop a dependent-time model (see section 2.1), where we concentrated on events first and then introduced time.

6.2 Events

Intuitively an event is something that happens, something that one can imagine putting in a time graph. What defines an event is very much dependent on the application, the domain and maybe even the genre. Our main interest lies in temporal information and how events are located in time – and not in the events themselves. We are not interested in the details of events, like participants, location and so on. Thus, we view events virtually as black boxes with no details (unless one of the details is an event itself – more about this in chapter 7).

Events have to be anchorable in time, and they are usually conveyed by finite verbs or nominalisations. Events can be ongoing or conceptually instantaneous, we do not distinguish between these.

Examples of event expressions are:

(122) *A small single-engine plane crashed into the Atlantic Ocean.*

(123) *By midafternoon, several vessels and a helicopter were combing the area.*

(124) *Searchers found the plane's landing gear, seat cushions and other debris.*

(125) *The 1996 crash of the TWA 747 remains unexplained.*

Events can be further subclassified, which will be discussed in section 6.6.

6.3 Times

Like events, times can be viewed as having extent (intervals) or as being punctual (points). Rather than trying to reduce one perspective to the other, as has happened in much philosophical discussion on time (see section 2.1.2), we shall simply treat both as *time objects*. It must, however, be possible, at least in principle, to associate a calendrical time with the time object. Examples of time referring expressions are:

yesterday
four months ago
last Thursday
in *January 1996*

However, time referring expressions can be quite complex, as in example (126), where the whole expression denotes a point in time.

(126) *17 seconds after hearing the sound ...*

The general structure here is that a time interval is explicitly related to an event and the calendrical time the whole structure refers to is the time 17 seconds after the event. This kind of time referring expression is called a *complex* time referring expression.

Following general convention, and the approach taken in MUC, we distinguish between two classes of time objects, *dates* and *times*, times which are larger or smaller than a day, respectively.

MUC also distinguishes between ***absolute*** time expressions (indicating a specific segment of time as in *20 minutes after 10, midnight, 10th of October*) and ***relative*** time expressions (indicating a date relative to the date of the document as in *yesterday* or *last month*). In our opinion, most time expressions, whether they are what MUC calls specific segments or not, are relative to the date of the document. If we consider, for example, the expression *October 10th* then it depends on the context whether the following or past October is referred to, as the following examples illustrate.

The plane crashed on October 10th.
The CEO will give a talk on October 10th.

An alternative would be to define an absolute time expression as one where the time object represented by the expression can be placed unambiguously on a calendrical timeline or time graph without any additional information from any other part of the text it occurs in. A relative time expression would, accordingly, be one where one does need additional information.

The latter distinction is the one we would favour, but since it is not clear at this point whether it would be helpful, we have ignored the relative/absolute distinction for now.

6.4 Temporal Relations

We described in section 3.4 how temporal relations can be conveyed in natural language, which we will briefly repeat here. Events can be related to times or to other events. Temporal prepositional phrases are the most common way to relate events to times, as example (127) shows.

(127) *A small single engine plane crashed into the Atlantic Ocean <u>on</u> Wednesday.*

Temporal relations holding between events are most commonly expressed by using temporal conjunctions as in examples (128) and (129).

(128) *The plane crashed <u>after</u> the pilot and his crew ejected.*

(129) *...<u>before</u> the craft fell, its three rotor blades shot off.*

Neither event-event relations nor event-time relations need necessarily be explicitly expressed, as example (130) shows:

(130) *Sunday, an F-14D crashed into the Pacific Ocean off southern California, killing its two crew members.*

Here simple juxtaposition is used to relate the plane crash event expression to the time referring expression *Tuesday*, where usually the preposition *on* would have been employed.

Although the temporal relation during is not explicitly signalled it is clear from the context. The plane crash expression is also related to the expression conveying the killing of the passengers, the latter being temporally included in the former. No temporal subordinate conjunction is used and again the temporal relation is not explicitly signalled in the text; but the non-finite construction is enough to make clear that the killing of the passengers happened *during* the crash.

In a conceptual framework we need to abstract from the large variety of the relations that exist in the real world and also account for the 'fuzziness' and vagueness with which they are, at times, expressed in natural language. We need to find relations that capture what is needed for our purposes while they must, at the same time, be easy enough to annotate so that the annotation scheme we are aiming to devise can actually be applied.

We found that the temporal relations we described in section 2.2 were not suitable for our task, for the following two reasons.

1. There are too rigid for what is expressed in real text. Temporal relations are not often expressed in text with the preciseness that Allen's and Gagnon's relations presuppose. It is rarely expressed, for example, that the start of one event exactly coincides with the end of another event – which is the only condition under which the temporal relation meets holds.

2. Some of the relations, especially the ones developed by Gagnon and Lapalme (1996), are too complicated and burdensome to annotate in practice.

The set of temporal relations we propose is geared towards annotation. With them we are trying to bridge the gap between relations suitable for capturing the temporal information in real texts, and those which are easy to annotate. The set is based on both Allen's and Lapalme's relations and the full set we propose at present is:

included and includes An event is temporally included within another event or time (example (131)) or it includes another event or time (examples (132) and (133)). Or rather the time interval over which the event happens is a subinterval or superinterval of the time interval represented by the time referring expression.

(131) *The plane crashed on Wednesday.*

(132) *By midafternoon, several vessels were combing the area.*

(133) *The airplane crashed, killing all passengers on board.*

before and after An event happens before or after another event or time.

(134) *The plane crashed after the pilot and his crew ejected.*

(135) *...before the craft fell, its three rotor blades shot off.*

(136) *This was the first crash since 1992.*

simultaneous This is the 'fuzziest' relation and should be called 'roughly at the same time'. It is surprisingly often that all we know about two events is that they overlap somehow or maybe happen at the same time, but we do not know anything more specific, as in example (137). If we would not allow the fuzzy simultaneous, and since we cannot apply a more precise relation, examples like (137) could not be captured at all.

(137) *All 75 people on board the Aeroflot Airbus died when it ploughed into a Siberian mountain in March 1994.*

Note that this set consists of two pairs of inverse relations (A includes/is after B iff B is included in/is before A) and the **simultaneous** relation. Thus, there really need be just three primitive relations and the remaining two can be viewed as being defined in terms of the others. This is an initial minimal set, defined after analysing a number of newswire articles. It can easily be expanded should it prove necessary or beneficial.

6.5 States

A state is a relation between entities or the holding of an attribute of an entity which, while capable of change, is ongoing over a time span, usually longer than the time span covered by the article. Examples for state expressions are:

(138) *The plane, which can carry four people, ...*

(139) *The water is about 125 feet deep in that area.*

Typically, a change of state constitutes an event. At this point we are less interested in states, and we do not take them into account in our annotation scheme.

6.6 Subclassifying Events

The base approach can be extended to include either genre/domain dependent or independent specifics. Specific types of events would be an example for a genre dependent extension whereas as the inclusion of a theory of causality would be genre independent.

During our analysis of newswire articles it became clear that events can be classified into groups. This is not a new idea. In section 2.1 we described two ways of classifying events. One is based on Vendler (1967) and Moens and Steedman (1988) and classifies events into aspectual classes based on their temporal extension and their association with consequent states.

Vendler-based classifications like this are based on aspect, a feature we did not include in our approach for the following reason. Exploiting aspect is based on analysing consecutive sentences and this cannot be easily exploited in newswire texts where the paragraphs are very short. Note that including aspect into an annotation scheme is also impractical. It would be extremely difficult to burden an annotator with having to decide for each verb whether it conveys a point, culmination, culminated process or process (see chapter 2). An a priori classification of all verbs is equally impractical.

We believe that other sources of temporal information, such as explicit time referring expressions and event co-reference (see chapter 7) are more important than aspect in the Vendlerian sense. Verb tense is a simple mechanisms, which only crudely locates events in time. Aspect is impractical and burdensome for the annotator, as we have mentioned earlier, it is less suited for newswire texts (see section 3.5). Thus, the event classification based on Vendler and Moens/Steedman is not adopted in our approach.

The other classification was developed by Halliday (see section 2.1). His classification, however, is not suitable for our purposes. For example, a *relational process* can also be *The fair is on Tuesday*, which we would want to place on a time graph whereas we would regard *Peter has a piano* as stative and not put it on a time graph.

What we need is a classification which helps us in the project of locating events in time, and so we propose the following classification scheme.

Occurrence Events Most events are what we call *occurrence events* – these are the events we want to place on a time graph. Examples are:

(140) *A small single-engine plane crashed into the Atlantic Ocean about eight miles off New Jersey on Wednesday*

(141) *By midafternoon, several vessels and a helicopter were combing the area about eight miles east of Sea Bright, N.J.*

(142) *Revenue in Avco and Textron Financial rose 19 percent last year.*

Reporting Events As mentioned above, there are also *reporting events*, whose main function is to associate the source of information with an (occurrence) event. Examples are:

(143) *The Coast Guard reported finding aircraft debris and fuel slick.*

(144) *Searchers found the plane's landing gear, seat cushions and other debris, Petty Officer Fenn said.*

(145) *The Coast Guard said the craft had taken off from Allaire Airport.*

The most common case is that the reported event happened in the past and thus before the reporting event. In some cases this helps to temporally locate an event more accurately than just before the date of the article, as the following example shows.

(146) *Aeroflot general manager for Hong Kong Vassili Tkatchenko said on Tuesday he was unaware the writ had been filed.*

Reporting events are what Bell (1991) calls attribution, the source of the information. The most common case is, that reporting events are not associated with a time. This means that we can locate them only crudely in time (they happen before the date of the article). Because of this and since their main function is to identify the information source, we do not put them on a time graph, unless they provide additional temporal information, as in example (146).

Perception Events The following example illustrates another type of event:

(147) *The plane was seen hitting the water shortly after 11 a.m. by a fisherman.*

We call these events ***perception events*** and although they are relatively rare, the benefits of annotating them justify their being included in the scheme. Perceived events (like *the plane hitting the water* in the example) happen roughly at the same time as the perception event (like *was seen by a fisherman* above) and although their exact temporal relationship might not be known, we are able to locate the perceived event more accurately in time than we would have been able to using only the occurrence event.

Aspectual Events The final class of events we distinguish is the class of ***aspectual events***, as example (148) illustrates.

(148) *At the former Grumman hangar in Calverton, investigators on Tuesday <u>began</u> piecing together the fractured parts of the airplane.*

Aspectual events have a very similar structure to reporting events. Both have the main event (*began* in example (148)) and an event as an argument (*piecing together*, also example (148)). Aspectual events usually involve ***aspectual verbs*** like *start, stop, finish etc.* Their temporal consequence is that the aspectual event indicates the start or ending of the related event.

Other Event Classes Another event class that could be included is the class of *attitude events*. Examples (149) and (150) show such events. Attitude events are similar to *reporting* and *perception* events in that they take another event as an argument.

(149) *John hopes to go to New York on Friday*

(150) *Mary believes that the plane hit the the Atlantic Ocean*

Like reporting events and unlike perception events, attitude events do not guarantee the reality of the participant event (X may report or believe that Y without Y being the case; in contrast, if X sees that Y then Y must be true). However, while both reporting and perception events usually take place after the event reported or at the same time as the even perceived, attitude events stand in no such clear temporal relation to their participant event. Thus, it could be useful to distinguish this event class from the others when taxonomising events with respect to their temporal properties.

Hypothetical events, as shown in examples (151) and (152) are difficult to interpret where time is concerned. Different hypothetical time lines could be created, depending on the consequences of these events.

(151) *The crash would never have happened had the pilot made a normal take-off or taken the cloudy weather into account.*

(152) *Aviation safety investigators say a sharp movement of the rudder could have caused the jet's deadly roll.*

6.7 Other Temporal Relations

Two other event-event relationships which are not temporal per se but have temporal implications are *causality* (where cause precedes effect) and *subeventness* (where subevents are temporally included in their 'container' events). However, these relationships are very difficult to define and to distinguish.

(153) *The plane crashed, killing all passengers on board.*

In example (153), is the *killing* event caused by the *crash* or is it a subevent? What constitutes subeventness and thus distinguishes it from causality cannot easily be stated precisely. For example, an event can happen during another event without being a subevent. Eating an apple while driving does not mean that the eating the apple event is a subevent of driving.

One could argue that a subevent is necessary for the event and that the event would not take place without the subevent. But not all events are as rigid as that. A 'going to the supermarket' event does not necessarily contain a 'buying milk' event – but often does. And when it does it would be a subevent. It would be too demanding a task for an annotator to decide for each event whether it stands in a causality or subevent relation to another event, so these two types of relationship are not included in the approach. Note that the temporal implications of these, temporal inclusion and temporal sequence, are still part of our approach as they are included in the set of temporal relations.

6.8 Discussion

In this chapter, we laid the conceptual foundation on which an annotation scheme to capture temporal information in text can be built. The temporal ontology introduced contains events, times, states, and temporal relations. Their respective classifications are also described in this chapter. The framework can be divided into a base part, which is applicable to all genres and domains, and into extensions, which cater for genre specific and potentially for domain specific mechanisms.

With this framework in place, the annotation scheme that was based on these is described in the next chapter.

Chapter 7

Annotating Temporal Information

Given the conceptual framework of chapter 6.1, we can now turn to proposing an annotation scheme, which enables events, times, and temporal relations to be marked up in newswire texts; event identity is treated as a special temporal relation from now on. The main purpose of the annotation scheme is to let texts be marked up in a way that makes it possible to determine the temporal relations between events and times and to build a time graph as accurately as possible.

One of the problems encountered when devising this annotation scheme was the trade-off between practicality, for which the scheme should be as simple as possible, and wealth of information encoded, which implies a certain degree of complexity. If the annotation scheme is not simple enough then it is very difficult to find human annotators willing to do the job and inter-annotator agreement is much harder to achieve. On the other hand, a certain amount of information needs to be encoded in the scheme to make interesting and meaningful results possible.

The process of evolving an annotation scheme and the methodology we adopted are cyclical. A proposal for the annotation was applied to several texts. Then the problems, weaknesses and strengths were analysed, which resulted in a revision of the annotation scheme. This led to a new proposal, which was applied to several texts etc.

Another point is that annotations for events, times and temporal relations did not evolve independently from each other. Events and times stand in temporal relations to each other and are thus intrinsically linked; the annotation scheme has to reflect this. Changing the annotation of one did often affect the annotation of the other.

Both the cyclical process and the interaction of the annotations are difficult, if not impossible, to reflect properly in the linear fashion the medium of paper dictates. What we have done instead, is to separate this chapter into two parts. The first part talks about annotating events, about the main proposals and their weaknesses and strengths. We then turn to time referring expressions and temporal relations, and we will briefly describe how their annotation interacted with annotating events. The second part describes the annotation scheme, that was the result of the process. The annotation of events, times and temporal relations are specified in detail, respectively, in the second part[1].

Like many modern annotation schemes, our scheme is defined in SGML (Structured General Mark-up Language, (Goldfarb, 1990)). Arguably, XML (eXtensible Mark-up Language, (Goldfarb, 1998)) is a more appropriate choice, but this work was already well advanced when the XML standard was agreed. The scheme we have defined could easily be adopted in XML and might benefit from this (see footnotes on pages 101 and 102).

7.1 Evolving an Annotation Scheme

In the beginning, when we started off with simple examples like *The plane crashed on Wednesday*, we tacitly assumed that there is a one-to-one relationship between events and clauses. This is not the case, as examples like *The plane was seen hitting the water by a fisherman* demonstrate, where two events (*hitting the water* and *was seen*) are conveyed in one clause. Also cases where more than one clause cover one event seem possible.

We will adopt the following terminology for the rest of the chapter. If a clause conveys

[1]Earlier versions of our work have been reported in Setzer and Gaizauskas (2000a), Setzer and Gaizauskas (2000b) and Setzer and Gaizauskas (2001).

more than one event, then the clause is said to describe a *compound* event. If only one event is conveyed by a clause, then this is called a *simple* event.

We will also mainly use compound events as examples, because not only is the annotation of simple events subsumed within the annotation of compound ones, but also because most of the problems and interesting cases occur with compound events. Time referring expressions are included in the examples but not analysed. A detailed description of how time referring expressions and temporal relation expressions evolved will follow in section 7.1.2.

7.1.1 Proposals for Annotating Events

We will now describe the main proposals we tested during the 'maturing' of the annotation scheme as well as briefly discuss their strengths and weaknesses.

Proposal 1

The first proposal we considered, was to annotate the whole text span covering the (simple or compound) event expressions, including time referring expressions, should the event be related to one. Whether the event expressed was a simple or a compound one, would be indicated by an appropriate attribute. Figure 7.1 shows the approach and example (154) shows how the annotation might look.

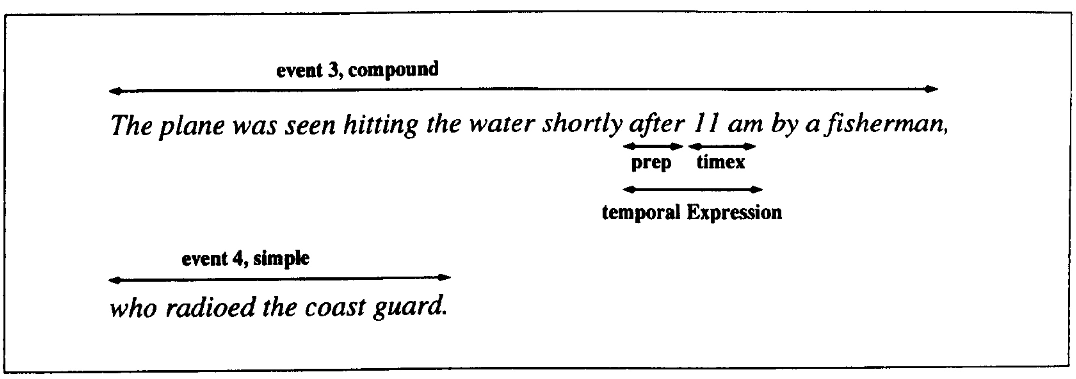

Figure 7.1: Proposal 1

(154) <event eid="3" type="COMPOUND>
The plane was seen hitting the water shortly
<tempExp eid="3">
<prep>after </prep>
<timex>11 a.m. </timex>
</tempExp>
by a fisherman
</event>
,
<event eid="4" type="SIMPLE>
who radioed the Coast Guard.
</event>

There are two main advantages of this proposal. One is, that it is a clause level scheme and there is a simple syntactic criterion for identifying event expressions: events are identified with finite clauses. The other advantage is that the annotation encompasses the whole text span and if more detailed information about the event should be needed later on, then this is still extractable. The weakness of this proposal is that compound events are treated the same way as simple events. The problem of dealing with compound events is not solved but merely deferred. Also, key information is not explicitly marked.

Another problem is that an event can be conveyed by discontinuous text spans as in *The plane, which can carry four people, crashed on Wednesday morning.* In this case, a portion of text would be included in the annotation, which does not belong to the event at all. The part *which can carry four people* is regarded as a state and does not belong to the crash event. This cannot be marked in the annotation scheme.

Proposal 2

In this proposal, compound event expressions are further analysed to the level of subevents. Each subevent expression, or in the case of simple events the event expression itself, are identified by annotating the finite verb, nominalisation, or non-finite verb representing the (sub)event. The textspan containing the simple or compound event is marked up as well. Figure 7.2 shows the approach and example (155) illustrates a possible annotation.

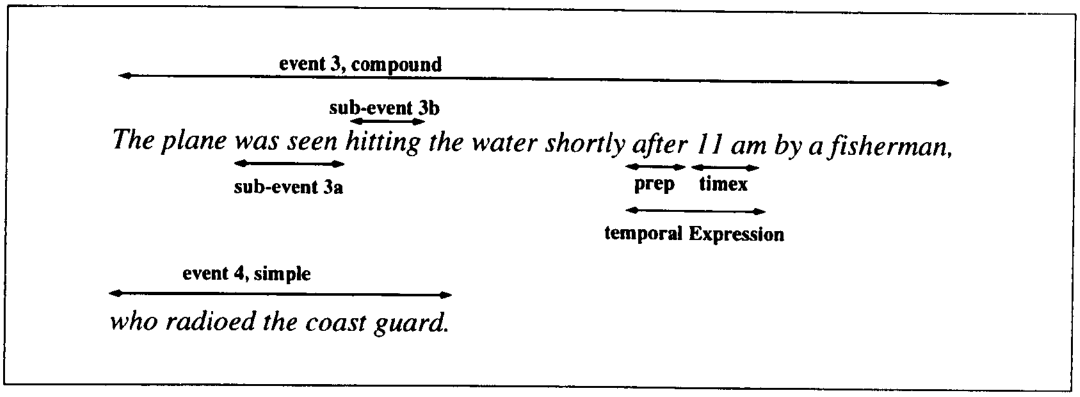

Figure 7.2: Proposal 2

(155)
```
<event eid="3" type="COMPOUND>
    The plane
<sub_event seid="3a" class="PERCEPTION">was seen </sub_event>
<sub_event seid="3b" class="OCCURRENCE">hitting </sub_event>
    the water shortly
<tempExp eid="3">
    <prep>after </prep>
    <timex> 11 a.m. </timex>
</tempExp>
    by a fisherman
</event>
    ,
<event eid="4" type="SIMPLE>
    who radioed the Coast Guard.
</event>
```

The proposal is an improvement on the previous one in that it identifies subevents. However, the problem of discontinuous text spans still persists. This proposal also proved to be too impractical to be useful.

Proposal 3

This proposal addresses the problem of events being conveyed by discontinuous text spans. The whole text span conveying the event is still annotated, as in the earlier proposals. In addition to that, we break up the text span into constituents, where each constituent covers a continuous portion of text belonging to an event expression. Each constituent also carries the ID or IDs of the event expression(s) it belongs to. Arguments like subject and object are not distinguished. Figure 7.3 and example (156) show how this could be done.

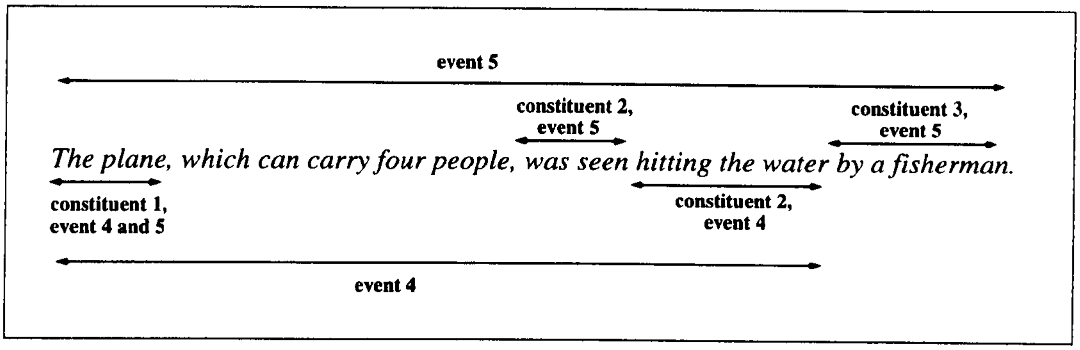

Figure 7.3: Proposal 3

(156) <event eid=5 arg_id=4 class=PERCEPTION tense=past >
<event eid=4 class=OCCURRENCE tense=past >
<constituent cid=1 eid=4,5 > The plane </constituent>
, which can carry four people,
<constituent cid=2 eid=5 > was seen </constituent>
<constituent cid=2 eid=4 > hitting the water </constituent>
</event>
<constituent cid=3 eid=5 > by a fisherman </constituent>
</event>

This proposal still identifies subevents and also solves the problem of discontinuous text spans by breaking the text span up into constituents. However, having to decide which text span belongs to which event or events proved far too burdensome for the annotators. Also, the information gained did not contribute to the actual task of ordering events in time.

Proposal 4

The idea behind this proposal is related to event co-reference. In order to decide, whether two event expressions convey identical events, it is helpful to consider the participants. If they are not identical, then consequently the two event expressions cannot convey identical events. This is an easy way to rule out event co-reference.

The proposal addresses this issue by separately annotating event indicators (like finite verbs and nominalisations) and event arguments (like subject and direct object) and have each element carry the ID of the event it belongs to. Figure 7.4 and example (157) illustrate the approach.

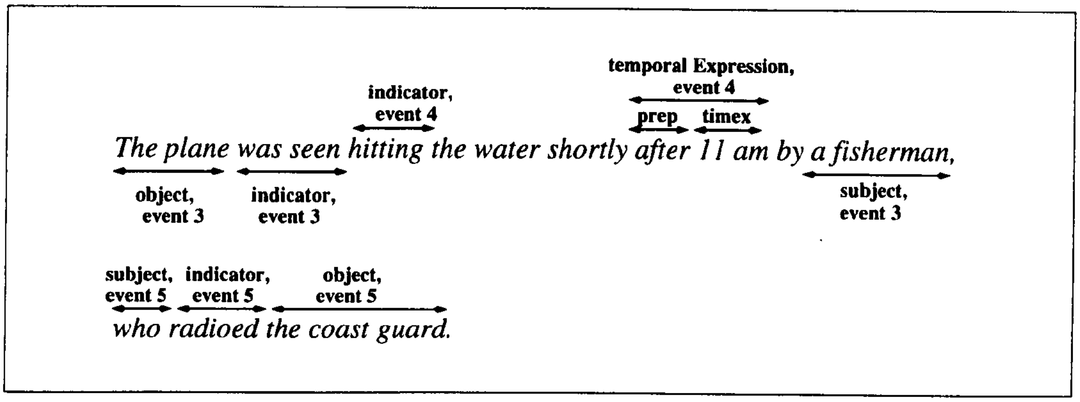

Figure 7.4: Proposal 4

(157)
```
<subject eid="3"> <object eid="4">The plane </object> </subject>
<indicator eid="4">was seen </indicator>
<indicator eid="3">hitting </indicator>
<object eid=">3the water </object>
    shortly
<tempExp eid="3">
    <prep>after </prep>
    <timex>11 a.m. </timex>
</tempExp>
    by
<subject eid="4">a fisherman </subject>,
<subject eid="5">who </subject>
<indicator eid="5"> radioed </indicator>
<object eid="5">the Coast Guard. </object>
```

Compound event expressions do not pose a problem anymore and there is easy access to details of the events like subjects and objects. But not only is this method is far too complex to apply, the level of detail this annotation scheme provides did not prove useful to the task of temporally ordering events.

In section 7.2, we will describe the annotation scheme we have adopted for events. This is based mainly on marking the head of the finite verb group, nominalisation or non-finite verb group.

We will now turn to time referring and temporal relation expressions.

7.1.2 Proposal for Annotating Time and Temporal Expressions

In this section, we will briefly describe how the annotation of temporal expressions[2] and temporal relation expressions interacted with the event annotation. As we have seen in the previous sections, we started out with annotating the whole text span conveying the event expression. This text span included temporal expressions, if there was one (see figure 7.5). The temporal expression was associated with the event simply by including the temporal expression annotation in the event annotation.

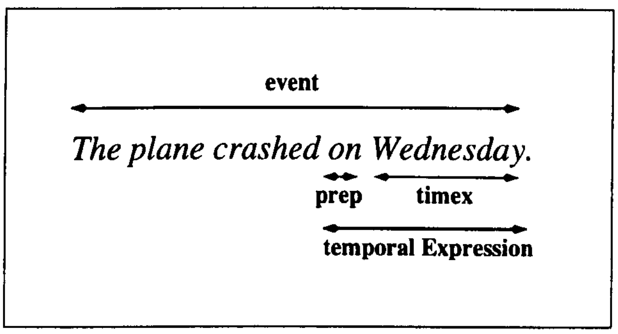

Figure 7.5: Temporal expression annotation included in the event annotation

This was unsatisfactory for several reasons. If the time is related to more than one event, as expressed in *The plane caught fire and crashed on Wednesday*, then we could not annotate this fact, because the temporal expression *on Wednesday* would have to included in one or the other event annotation. Similarly, the time the event is related to is not

[2] *Temporal expression* is a term we used in the earlier stages, when we were treating the time expression and the signal (preposition) as one entity – a temporal expression, see figure 7.5

necessarily expressed in the same sentence, as in *The plane crashed. On the same day* ....

Also, on a conceptual level, it would more consistent if the annotation scheme would reflect the fact that both times and events are entities in their own right.

The policy of annotating the whole text span was abandoned, for this and other reasons described in the previous section, and we decided to only annotate a representative of the event. At this point, temporal expressions were annotated by marking up the text span that conveyed the time referring expression and the preposition, each of which were individually annotated as well. This annotation now included a unique ID for the temporal expression to identify it. Figure 7.6 show this stage, with the event now being represented by the finite verb and the temporal expression annotation not included in the event annotation anymore. The association of the event and the time is realised via an SGML attribute (more about this in the next section).

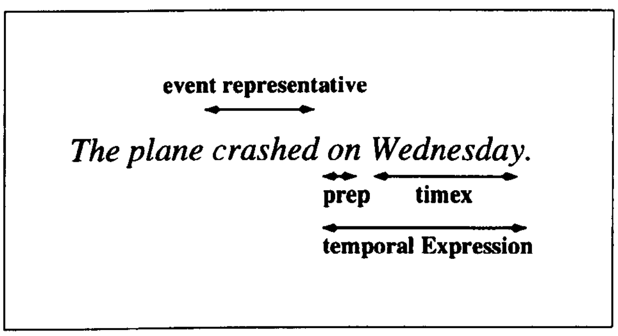

Figure 7.6: Temporal expression annotation outside the event annotation

When we consider annotating event-event relations, then we face a similar scenario. In the sentence *All passengers died when the plane crashed*, two simultaneous events are described. The temporal relation is signalled by *when*. The conjunction *when* is annotated separately from both event expressions, as shown in figure 7.7

Here, we were relating two event expressions via a signal for the temporal relation (the conjunction). The same approach was adopted for temporal expressions. These were broken up into the time referring expression and the preposition. The preposition fulfils the same

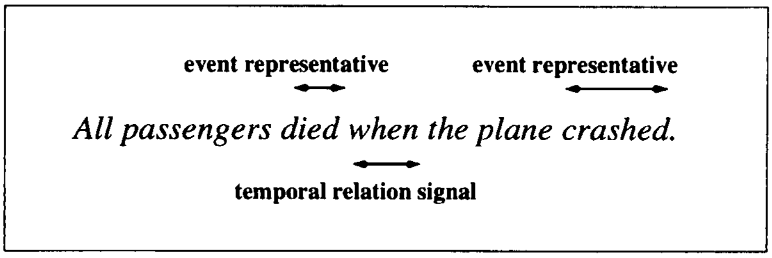

Figure 7.7: Two events related via a conjunction

function as the conjunction, that of indicating the temporal relation between entities, and is now treated in the same way. We have a unified, conceptually neater approach of using one principle for relating event expressions to time expressions and event expressions to event expressions, illustrated in figure 7.8.

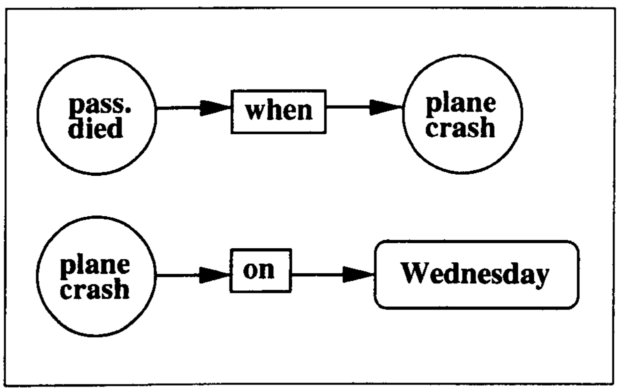

Figure 7.8: Relating and event to a time and an event to another event

Events can be related to other events or time-objects. This may be expressed explicitly via a preposition or a temporal subordinate conjunction which signals the temporal relation between the two entities. In this case, it seems natural to link the two entities in the annotation via this signal and have the actual temporal relation (like *included*) as an attribute of the signal. Example (158) illustrates this approach.

(158) The plane <event eid=2> crashed </event>
<signal sid=1 eid=2 tid=1> on </signal>
<timex tid=1> Wednesday </timex>.

But events can also be related *implicitly* to other events or times and this poses a serious problem to above approach, since there is no signal to attach the temporal relation to and it is not clear what an alternative would be.

The solution we propose is to always associate the temporal relation and the ID of event it is related to with an event itself, using attributes. Should this relation be explicitly signalled then the ID of the signal becomes an attribute of the event as well, so the connection to the signal is not lost. Figures 7.9 and 7.10 illustrate this approach.

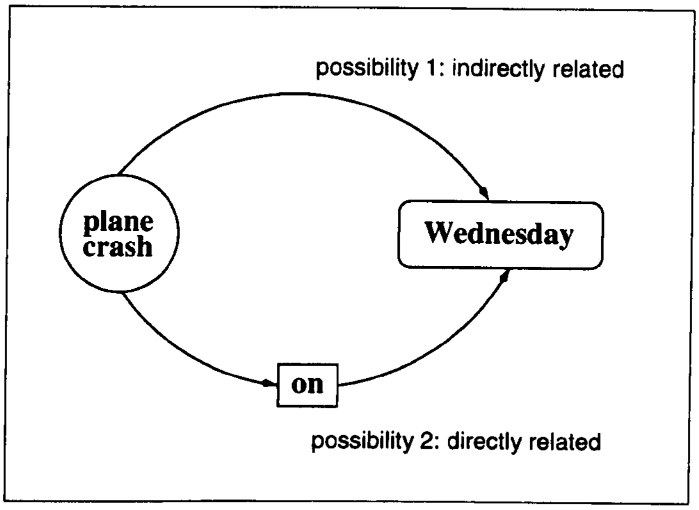

Figure 7.9: Event implicitly and explicitly related to a time

In this section, we have given a brief account of how the annotations for events, times and temporal relations have influenced each other.

7.2 The Annotation Scheme

After having considered the above proposals and tried them out on sample texts, we came to the conclusion that it is not worthwhile trying to enable access to the argument-level details of the events (i.e. subjects and objects etc.), because this leads to the scheme being too impractical. Furthermore, we are not so much interested in the events themselves but

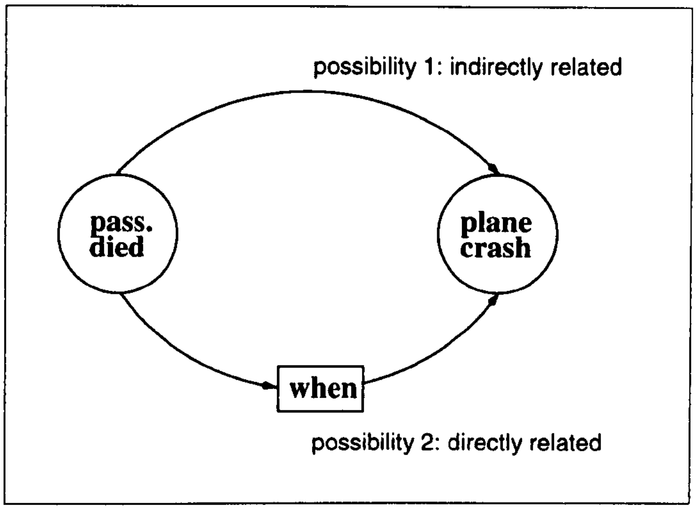

Figure 7.10: Event implicitly and explicitly related to another event

rather in the temporal relations holding between events and between events and times, we can see events as black boxes without details like logical subject or logical object, unless one of the arguments is an event itself. The following sections will present the scheme finally adopted for the annotation of events, times and signals and the temporal relations between them. In chapter 9, we will present a pilot study we conducted, during which this annotation scheme was applied to several newswire texts. The annotation guidelines given to the annotators, which detailed how exactly this scheme was to be applied can be found in appendix A.

7.2.1 Annotating Events

We annotate a representative of the event expression and the first candidate for a representative is the **head of the finite verb group**. If the event is conveyed by a nominalisation then we chose the **head of the nominalisation** as the representative. If the event is represented by a non-finite clause, we annotate the **non-finite verb** as the representative.

The following examples, which do not yet include attributes, show the basic approach.

(159) The plane <event> hit </event> the water near New Jersey.

(160) The plane crashed, <event> killing </event> all passengers on board.

(161) The 1996 <event> crash </event> of the TWA 747 remains unexplained.

Event expressions have a number of optional and obligatory attributes associated with them which are described below, including examples which only show the attributes relevant to the example. An example with all attributes is given further below.

eid: The event ID uniquely identifies the event expression in the text.
potential values: positive integers
optional: no

The plane <event eid=3> crashed </event>.

class: An event belongs to one of these classes.
potential values: OCCURRENCE, PERCEPTION, REPORTING, ASPECTUAL [3]
optional: no

See next attribute for example.

argEvent: Reporting, perception and aspectual event expressions usually have another event or other events as an attribute and the argument ID identifies these.
potential values: positive integers
optional: yes

The MoD <event eid=5 class=reporting argEvent=7> announced </event> that the jet fighter <event eid=7 class=occurrence> ploughed </event> into the mountain.

tense: If the event is signalled by a finite verb, then the verb tense is the (obligatory) value for this attribute. If the event is signalled by a nominalisation or non-finite verb then

[3] These event classes, which have been discussed in section 6.6, are not an exhaustive set, but a good starting point for our purposes. Other possible, not included, event classes are also discussed in section 6.6.

the annotator can use this field to indicate whether the event happens in the present, past or future. In this case, it is an optional attribute
potential values: PAST, PRESENT, FUTURE
optional: yes

> The plane <event eid=3 tense=past> crashed </event>.
> Mr Smith will <event eid=11 tense=future> make </event> an announcement tomorrow afternoon.

aspect: It is sometimes helpful to know whether the aspect of the verb is progressive or perfective, which can be indicated using the attribute aspect.
potential values: PROGRESSIVE, PERFECTIVE
optional: yes

> ...several vessels and a helicopter were <event eid=13 tense=past aspect=PROGRESSIVE> combing </event> the area.

relatedToEvent: The ID of the event expression that the current event expression is temporally related to is stored here[4].
potential values: positive integers
optional: yes

> See next attribute for example.

eventRelType: The type of temporal relation holding between the related events is stored in this attribute.
potential values: BEFORE, AFTER, INCLUDES, IS_INCLUDED, SIMULTANEOUS (see chapter 6.4 for a description of these relations)
optional: yes

[4]Not being able to relate an event expression to more than one event expression is a weakness of our scheme and this needs addressing. SGML does not offer a straightforward solution, one proposal is to use a list like structure for the argument as well as eventRelType. A disadvantage is, that one has to ensure that the order of the event IDs reflects the order of the relation types. This problem did not occur in the texts we analysed and did not cause a problem in the pilot study. This is a point where porting the scheme to XML might be beneficial.

All 75 people on board the Aeroflot Airbus
<event eid=4 relatedToEvent=5 eventRelType=simultaneous >
died </event>
when it <event eid=5> ploughed </event>
into a Siberian mountain.

relatedToTime: The ID of the time-object the current event is related to is stored here[5].

potential values: positive integers

optional: yes

See next attribute for example.

timeRelType: The type of temporal relation holding between the event and related time is stored in this attribute.

potential values: BEFORE, AFTER, INCLUDES, IS_INCLUDED, SIMULTANEOUS

optional: yes

A small single-engine plane <event eid=9 relatedToTime=5 timeRelType=is_included signalID=9> crashed </event>
into the Atlantic Ocean <signal sid=9> on <signal>
<timex tid=5> Wednesday </timex>.

signalID: The ID of the text span that signals the temporal relation holding between two entities can be kept in this attribute.

potential values: positive integers

optional: yes

See relatedToEvent and relatedToTime for example.

[5]Similarly to relatedToEvent, it is a weakness of our approach that an event expression cannot be related to more than one time referring expression. This seems less likely to occur and hence is less of a problem. However, solving the problem of relating an event expression to multiple event expressions will automatically solve this part too - using XML might be an option.

Note, that the annotation scheme, and the annotation of events in particular, was planned with future extensions in mind. We did not include all event attributes into the analysis of the annotation, because we concentrated on aspects like surface cues and event identity first.

Verb tense, for example, indicates only a very crude temporal location. Past tense locates the event before the date of the article, future tense after this date and present tense indicates that the event happens roughly at the same time. Nominalisations and non-finite verbs lack tense and cannot be located with this feature. Because we wanted to focus on other aspects of temporal information, verb tense and (Vendlerian) aspect are not included in the analysis of the temporal information.

Perception, aspectual events, and reporting event associated with time did occur so infrequently in the corpus (see section 9.1), that these are not included in the analysis yet.

7.2.2 Annotating Time Expressions

We distinguish between *simple* (example (162)) and *complex* (example (164)) time referring expressions. Complex referring time expressions refer to a point in time by relating (*after*) an interval (*17 seconds*) to an event (*hearing the sound*). The point in time referred to is the point at the end of the interval. Simple time referring expressions are often the noun part in temporal prepositional phrases or adverbial phrases, see section 3.3 for more examples.

(162) *last Thursday*

(163) *October 3rd*

(164) *17 seconds after hearing the sound ...*

(165) *three days before Easter*

For simple time referring expressions we annotate the whole text span conveying the time-object, as in examples (166) and (167).

(166) <timex tid=5 type=DATE> Tuesday </timex>

(167) <timex tid=5 type=TIME> 11 a.m. </timex>

Following general convention, and the approach taken in MUC, we distinguish between two classes of time objects, *dates* and *times*, units which are larger or smaller than a day, respectively.

Like event expressions, time referring expressions have a unique ID so they can be uniquely identified in the text and be associated with events. All time referring expressions, simple and complex, have the following attributes:

tid: The ID uniquely identifies the time referring expression in the text.
potential values: positive integers
optional: no

type: The type of the time object conveyed.
potential values: DATE, TIME, COMPLEX
optional: no

calDate: The calendrical date represented by the time object.
potential values: [[DD]MM]YYYY or ('SPR'|'SUM'|'AUT'|'WIN')YYYY
optional: no

Note that we only annotate time *referring* expressions (see section 3.3). Examples for the annotation of time referring expressions are (166) and (167).

The more complex time referring expressions, like the one in example (164), include a time interval (*17 seconds*), a preposition (*after*) and an event (*hearing the sound*) or time. Annotating the whole time span would not only be tedious for the annotator but also obscure the elements involved, especially since a marked up event would be nested inside the annotation of the time referring expression.

The time referred to in example (164) is the point in time 17 seconds after the event mentioned. The chosen solution, which is similar to event annotation, reflects this. The interval is chosen as the representative for the time referring expression and related to the event expression via the temporal relation, usually signalled by the preposition. Figure 7.11 illustrates this approach. These complex time objects have the following additional

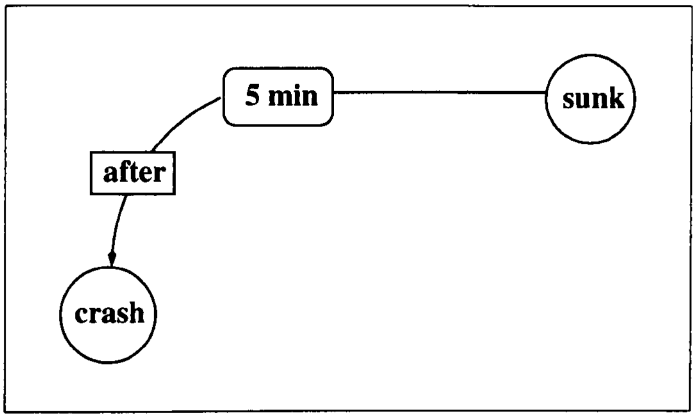

Figure 7.11: A complex time expression

attributes:

eid: The ID of the event the time interval is related to.

potential values: positive integers

optional: no

signalID: The ID of the signal indicating the temporal relation.

potential values: positive integers

optional: no

relType: The temporal relation holding between the time interval and the event.

potential values: BEFORE, AFTER, INCLUDES, IS_INCLUDED, SIMULTANEOUS

optional: no

Example (168) illustrates the approach.

(168) <timex tid=5 type=complex eid=3 signalID=7 relType=after>
17 seconds </timex>
<signal sid=7> after <signal> <event eid=3> hearing</event>
the sound...

7.2.3 Annotating Temporal Relations

Events can be related to times and to other events, expressed either implicitly or explicitly, as we have described in chapter 3.

If an event is related to a time, then we store the ID of the time referring expression and the temporal relation holding between them with the event expression (as SGML attributes). If this relation is explicitly signalled, then the signal expression ID is stored as an attribute too. If the relation is implicitly expressed, then this attribute is simply omitted. Examples (169) and (170) express an implicitly and explicitly related event expression, respectively.

(169) A small single-engine plane
<event eid=9 class=OCCURRENCE tense=past
relatedToTime=5 timeRelType=included >
crashed </event>
into the Atlantic Ocean about eight miles off
New Jersey
<timex tid=5> Wednesday </timex>.

(170) A small single-engine plane
<event eid=9 class=OCCURRENCE tense=past
relatedToTime=5 timeRelType=included
signalID=9>
crashed </event>
into the Atlantic Ocean about eight miles off
New Jersey
<signal sid=9> on <signal>
<timex tid=5> Wednesday </timex>.

To annotate event-event relations, we choose one event expression and store the information about the temporal relation and the ID of the other event expression as attributes.

Is this relation explicitly signalled, then the ID of the signal expression is stored as an attribute too; if it is implicitly signalled then this attribute is omitted. Example (171) shows an event expression being explicitly related to another event expression.

(171) All 75 people on board the Aeroflot Airbus
<event eid=4 class=OCCURRENCE tense=past
relatedToEvent=5 eventRelType=simultaneous
signalID=7>
died </event>
<signal sid=7> when <signal>
it
<event eid=5 class=OCCURRENCE tense=past >
ploughed </event>
into a Siberian mountain.

7.3 Comparing Temporal Annotations

The main reason annotation schemes are developed, is to create annotated corpora and often the goal is to be able to do this automatically. Before it is possible to do this, the annotation scheme has to be validated, which is often done by hand-annotating a trial corpus and analysing the results. To overcome the potential inconsistencies hand-annotated corpora are prone to and to ensure the quality of the description of the scheme used by the annotators during annotation, it is necessary to be able to compare annotations and to assess their 'goodness'.

In this chapter we will describe a method of comparing the temporal annotations that can be produced using the annotation scheme described above, and a similar method used to evaluate co-reference annotation.

7.3.1 The Comparison Method

When comparing annotations based on the proposed scheme, an additional complexity is that the annotations are of a semantic nature and should be compared in semantic not formal terms. Two annotations should be said to be equivalent if they convey the same 'temporal information', even if different ways of annotating this are chosen. How semantically identical but differently annotated cases can look like is explained in the following example.

Event A and event B are simultaneous and both are before event C. This is illustrated in figure 7.12. It is not necessary to annotate all three relations, we can arrive at the same

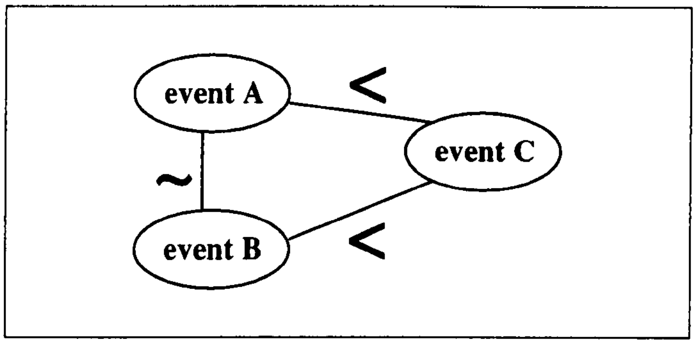

Figure 7.12: Comparing annotations 1

temporal information using simple inference rules. From *A is simultaneous to B* and *A is before C*, we can infer that *B is before C*. Similarly, from *A is simultaneous to B* and *B is before C* we infer that *A is before C*. Accordingly, these are the two ways in which the same information can be annotated. This is shown in figure 7.13. The annotations are

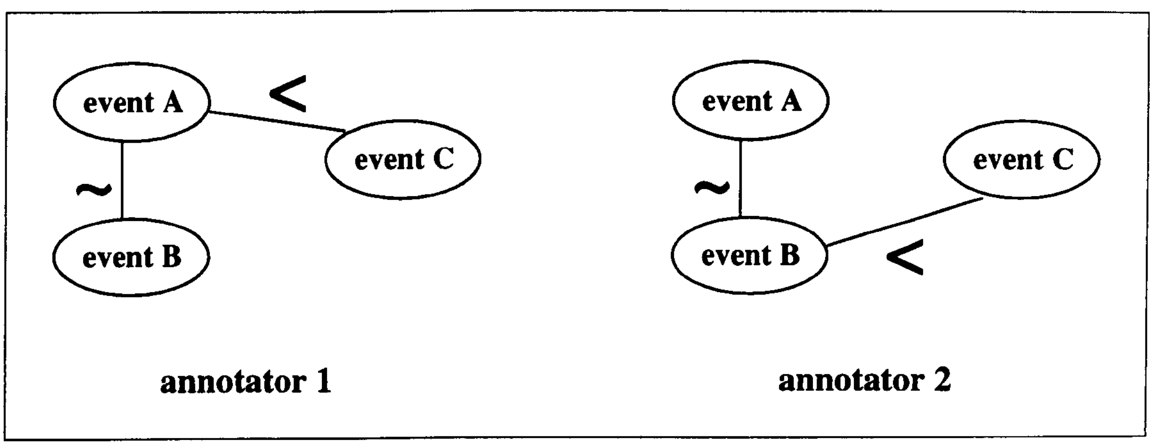

Figure 7.13: Comparing annotations 2

equivalent, because we arrive at the same temporal information. Yet, they are different. Any comparison of annotation should take this into account.

The proposed solution is inspired by the intuitive notion of the complete temporal information content of a text, which contains the temporal relations holding between all pairs of events and events and all pairs of events and times. The solution associates an annotation with a model theoretic interpretation, in the following way.

The set of temporal relations that we include in our framework is {*includes, is_included, before, after, simultaneous, identity*}[6]. The relations *includes* and *is_included* as well as the relations *before* and *after* are inverse relations (respectively), and we can reduce the number of different types of temporal relations to three primitives using only *simultaneous, before* and *includes*. All relations of the form A *after* B (A *is_included* B) are normalised to B *before* A (B *includes* A) and are subsumed within the relation *before* (*includes*). *Identity* is translated into *simultaneous*. This normalisation make the set of inference rules smaller and the process clearer.

The event and time expressions annotated in a text, or rather their respective IDs, form two sets, E and T, respectively. Since all of our temporal annotations are binary relations relating event expressions or time expressions to other event expressions or time expressions, the denotation of each relation as specified in the text can be viewed as a subset of $(E \cup T) \times (E \cup T)$. For each temporal relation certain formal properties pertain. For example *simultaneous* is an equivalence relation, while *before* and *includes* are transitive, but asymmetric and irreflexive. Therefore, given a partially specified model of the temporal relations in a text (specified as a set of pairs comprising a part of the denotation of the relation), the deductive closure of each relation can be computed to arrive at a total model. If the deductive closures of two partially specified models are identical, then these two models are equivalent, though not themselves identical. Further if any (partial) model $\mathcal{M}$ is such that no proper subset of $\mathcal{M}$ has an equivalent deductive closure to $\mathcal{M}$, then $\mathcal{M}$ is a *minimal* model of the temporal relations in the text. This minimal model cannot be unique,

[6]Recall, that we included event identity in our set of temporal relations.

as the simple example shows (figures 7.12 and 7.13). A minimal model for the temporal relations holding between the three events A,B, and C has to contain two temporal relations. Figure 7.13 shows two models containing two temporal relations each, and each model is minimal.

Let us denote sets of pairs from $(E \cup T) \times (E \cup T)$ which constitute the denotations of *simultaneous, before* and *includes* by S, B and I respectively. The set of inference rules we use is:

$\forall x, y, z \in (E \cup T)$:

1. $(x, y) \in S \Rightarrow (y, x) \in S$
2. $(x, y) \in B \wedge (y, z) \in B \Rightarrow (x, z) \in B$
3. $(x, y) \in I \wedge (y, z) \in I \Rightarrow (x, z) \in I$
4. $(x, y) \in B \wedge (y, z) \in I \Rightarrow (x, z) \in B$
5. $(x, y) \in I \wedge (x, z) \in B \Rightarrow (y, z) \in B$
6. $(x, y) \in S \wedge (y, z) \in S \Rightarrow (x, z) \in S$
7. $(x, y) \in B \wedge (y, z) \in S \Rightarrow (x, z) \in B$
8. $(x, y) \in I \wedge (y, z) \in S \Rightarrow (x, z) \in I$
9. $(x, y) \in S \wedge (y, z) \in I \Rightarrow (x, z) \in I$
10. $(x, y) \in B \wedge (x, z) \in S \Rightarrow (z, y) \in B$

We then can denote the deductive closure of S, B, and I by $S^{\models}$, $B^{\models}$, and $I^{\models}$ respectively. Given these definitions we are now in a position to specify what precision and recall mean in this framework. Letting S_k and S_r denote the annotated *simultaneous* relations in the answer key and system response respectively and $S_k^{\models}$ and $S_r^{\models}$ their deductive closures, respectively (and similarly for B and I). The recall and precision for the *simultaneous* relation is given by:

$$R = \frac{|S_k^{\models} \cap S_r^{\models}|}{|S_k^{\models}|}$$

$$P = \frac{|S_k^{\models} \cap S_r^{\models}|}{|S_r^{\models}|}$$

Recall and precision measures can be defined in a parallel fashion for the other relations. An overall recall and precision measure for all temporal relations can then be defined as follows[7]:

$$R = \frac{|S_k^{\models} \cap S_r^{\models}| + |B_k^{\models} \cap B_r^{\models}| + |I_k^{\models} \cap I_r^{\models}|}{|S_k^{\models}| + |B_k^{\models} + |I_k^{\models}|}$$

$$P = \frac{|S_k^{\models} \cap S_r^{\models}| + |B_k^{\models} \cap B_r^{\models}| + |I_k^{\models} \cap I_r^{\models}|}{|S_r^{\models}| + |B_r^{\models} + |I_r^{\models}|}$$

There are limitations to the inference rules caused by the nature of the temporal relations we have chosen (see chapter 6.1). The relation *simultaneous* is not as clearly defined as *before* or *includes*; it is a 'fuzzy' relation chosen to accommodate the lack of precision in what we observed in newswire articles (see chapter 6.1). This is good for annotating natural language newswire texts but has the consequence that some of the inference rules we use can potentially lead to incorrect conclusions, as we now show.

Rules 1 to 3, which represent properties of the temporal relations like transitivity of *before* and *includes*, and symmetry of *simultaneous*, hold under any circumstance. If event A is simultaneous to event B then B is simultaneous to A too. If event A is before event B and event B is before event C, then event A is before event C. Rules 4 and 5, which are solely based on the relations *before* and *includes* hold irrespective of the event or time involved as well. If the pilot ejected from the plane before it crashed and the crash includes an explosion, then the pilot ejected before the explosion (rule 4), similar for rule 5.

Rules 6 to 10, however, are different. They involve the fuzzy *simultaneous* relation and we could find examples where these rules do not supply the correct temporal relation. For example, it would be possible to imagine an example following the temporal structure

[7]This method can be compared to Crowe (1997)'s way of evaluating clause-event grids.

shown in figure 7.14 which would not fit with rule 7. Events z and y are considered to be 'fuzzily' simultaneous, and event x is before event y. These rules hold for the corpus we

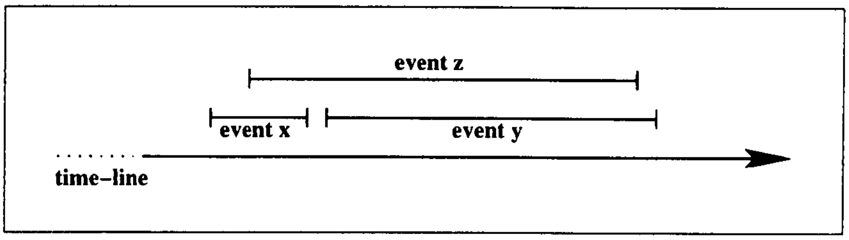

Figure 7.14: Counter example

have developed (see chapter Experiments) and we expect them to hold for a large majority of all cases, so we kept them in the set of inference rules.

Other possible rules are excluded because they would lead to too many wrong inferences (even though they might hold for a few cases), an example is $(x, y) \in I \wedge (y, z) \in B \Rightarrow (x, z) \in B$, illustrated in figure 7.14.

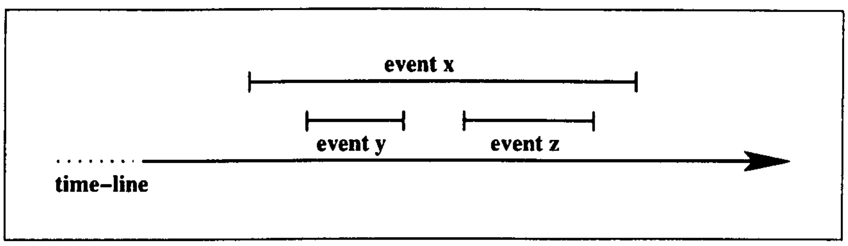

Figure 7.15: Example for excluded inference rule

The problem of scoring semantically identical but differently annotated temporal relations is similar to the co-reference scoring methods and problems in MUC 6, described in the following section.

7.3.2 The MUC co-reference scoring scheme

The difficulty with scoring co-reference is how to score semantically identical but differently annotated co-reference chains. Consider the following case where A, B, C and D are coreferring. This co-reference chain can be equivalently represented in multiple ways, for example by the following two sets of links:

<A-B, B-C, A-D>

<A-B, B-C, C-D>

The problem was to compute recall and precision between a system response and a human-annotated key without assuming syntactically identical annotations.

The solution developed for MUC6 exploited the fact that co-reference is a transitive, symmetrical and reflexive relation and so the sets of links form equivalence classes. For example, both sets above form the equivalence class {A, B, C, D} since they both represent the same co-reference chain.

We will briefly explain how to calculate Recall and Precision, using the following simple example (taken from MUC6). The elements involved are $\{A, B, C, D\}$ and $\{B, C, D\}$ are coreferring. The key is <B-C, C-D> and the response is <B-C A-D>.

Recall Intuitively, the Recall should be 0.5 since the minimal number of links needed to link the three elements in the key class is two and the response provides only one correct link. If c is the minimal number of correct links needed and m is the number of links missing in the response then recall is:

$$Recall = \frac{c - m}{c}$$

And for the example:

$$Recall = \frac{2 - 1}{2} = 0.5$$

Precision To calculate precision the approach is reversed. It is necessary to count the number of links which must be added to the key to yield the equivalence class of the response.

If, for example, the key is <B-C, C-D> and the response is <B-C A-D> then intuitively the precision is 0.5 since of the two given links only one is correct. If c' is the minimal number of links needed to generate the response equivalence class and m' is the number of links missing in the key to generate that class, then the formula for precision is:

$$Precision = \frac{c' - m'}{c'}$$

And for the example:

$$Precision = \frac{2-1}{2} = 0.5$$

For more detailed information and more complex examples, please refer to MUC6.

7.3.3 Contrasting the Scoring Methods

There are two reasons, why the MUC scoring scheme was not adopted. Firstly, the MUC method does not capture the intuition humans have about the temporal information content of a text, which is the full set of temporal relations holding between all event-event pairs and all event-time pairs.

Whereas the MUC scoring method is based on a minimal model for the coreference chain and its cardinality, the scoring method presented in section 7.3.1 is based on a temporal closure being computed over the temporal relations annotated. The temporal closure establishes as much of the temporal information content as possible, given the axioms described above and the temporal relations annotated. This is illustrated with the following example.

Imagine 9 simultaneous events $\{e_1, e_2, \ldots, e_9\}$. The key might form a 'simultaneity chain' like the following:

$$e_1 \sim e_2 \sim e_3 \sim e_4 \sim e_5 \sim e_6 \sim e_7 \sim e_8 \sim e_9$$

The response might consist of 3 smaller chains:

$$e_1 \sim e_2 \sim e_3 \qquad e_4 \sim e_5 \sim e_6 \qquad e_7 \sim e_8 \sim e_9$$

The scoring method will compute the deductive closure over both key and response. For the key, the deductive closure will contain 72 temporal relations; from the 81 potential event pairs, the 9 relations holding between an event and itself are not included. The deductive closure for the response contains 18 events. Because each chain of 3 events has no connection to either of the other two chains, the temporal closure cannot establish any temporal relations between events in different chains. The temporal relations, both annotated and inferred, can be intuitively visualised using a matrix. Each entry corresponds to a relations holding between two events. This is illustrated in figures 7.16 and 7.17.

	e1	**e2**	**e3**	**e4**	**e5**	**e6**	**e7**	**e8**	**e9**
e1		~	~	~	~	~	~	~	~
e2	~		~	~	~	~	~	~	~
e3	~	~		~	~	~	~	~	~
e4	~	~	~		~	~	~	~	~
e5	~	~	~	~		~	~	~	~
e6	~	~	~	~	~		~	~	~
e7	~	~	~	~	~	~		~	~
e8	~	~	~	~	~	~	~		~
e9	~	~	~	~	~	~	~	~	

Figure 7.16: Temporal Closure for the key

Recall in this case is $18/72 = 0.25$. Precision is 1, since all relations in the response are in the key as well.

The recall value is different, if the response misses different temporal relations. Imagine the response being:

$$e_2 \sim e_3 \sim e_4 \sim e_5 \sim e_6 \sim e_7 \sim e_8$$

	e1	e2	e3	e4	e5	e6	e7	e8	e9
e1		~	~						
e2	~		~						
e3	~	~							
e4					~	~			
e5				~		~			
e6				~	~				
e7								~	~
e8							~		~
e9							~	~	

Figure 7.17: Temporal Closure for the response

The temporal closure for this response contains 42 relations, computed as described above and as illustrated in figure 7.18.

	e1	e2	e3	e4	e5	e6	e7	e8	e9
e1									
e2			~	~	~	~	~	~	
e3		~		~	~	~	~	~	
e4		~	~		~	~	~	~	
e5		~	~	~		~	~	~	
e6		~	~	~	~		~	~	
e7		~	~	~	~	~		~	
e8		~	~	~	~	~	~		
e9									

Figure 7.18: Temporal Closure for the second response

Recall for this response is $42/72 = 0.58$, precision is, again, 1.

This method captures the intuition, that a long 'chain' of simultaneous events carries more information than several short ones, because we can infer more temporal relations. If we look at figures 7.17 and 7.18, then we see that more entries in the matrix are filled in for the second response than for the first, which means that more relations have been inferred. With the information given in the second response, it is possible to answer more question such as *Are events 3 and 6 simultaneous?*.

Using the MUC scoring method, the cardinalities of the sets are used to compute recall and precision. Recall in this case is $6/8 = 0.75$, since of the 8 links necessary to link the 9 events together, only 6 are in the response. Precision, again, is 1. This holds for both responses given. The MUC scoring scheme does not take into account that different sets or 'chains' of simultaneous events can lead to a different number of inferences being drawn despite both sets or 'chains' having the same cardinality.

A similar argument holds for computing precision (also see Bagga and Baldwin (1998)). Wrongly connecting two large chains leads to more wrong inferences being drawn than connecting two short chains, and leads to a lower precision value. The MUC scoring method would compute the same value for precision in both cases.

This example also shows, that the scoring method we adopted is 'harsher' than the MUC scoring method. Both results for recall are lower than the recall results for MUC. The reason for this is, that computing a closure over the temporal relations in the key and response quadrates the number of temporal relations compared (as shown in the example above).

The second reason for not adopting the MUC scoring method is twofold. The MUC method is not based on a unique minimal model, but on the cardinality of **a** minimal model. The method is also based on the fact that coreference is an equivalence relation, with which such neat models can be formed. We do not have equivalence relations only, before and includes are asymmetric and irreflexive. The different relations form different sets (as explained in section 7.3.1 above), and it is unclear whether it is possible to rely on cardinality for computing recall and precision. The question, whether a scoring scheme could be developed for our approach, which is based on minimal models, is non-trivial and a research topic on its own.

7.4 Discussion

We have devised and described in detail an annotation scheme for annotating those features and relations in texts which enable us to determine the relative order and, if possible, the absolute time, of the events reported in them. This scheme has been used to construct an annotated pilot corpus, the process and results of which are described in the following two chapters. We have also introduced a model-theoretic method for comparing temporal annotations and to compute recall and precisions for annotations.

Chapter 8

Creating an Annotated Corpus

In chapter 7 we explained in detail how the text segments signalling entities in our ontology (events, times and temporal relations) are to be annotated. This chapter describes how to put the approach we adopted into practice. To make the process of annotation as easy as possible, we developed a simple annotation tool[1], but the approach described here is independent from the system used.

Following the experiences of a study on interannotator agreement for a co-reference task (Hirschman et al., 1998), where the interannotator agreement was improved by separating the task of marking co-reference into a two stage process, we decided to follow a similar approach. The annotation takes place in the two major stages described in section 8.1 and section 8.2 respectively. Each stage has 3 phases, breaking down the annotation task into manageable chunks.

8.1 Stage I of the Annotation

The first stage of the annotation has three phases.

[1]This tool, developed by the author with input from Rob Gaizauskas and Mark Hepple, is written in Perl/Tk and contains about 5000 lines of code.

Phase 1 All event and time expressions are annotated, including filling in attributes (see section 7.2). Figure 8.1 shows an example text and indicates what would be annotated during the first stage. There are four event (circled) and two time expressions (boxed) in this text.

The army said Friday that a pilot was probably showing off for his parents when he crashed an F14–A fighter in Nashville in January, killing himself, a fellow officer and three people on the ground.

Figure 8.1: Annotation process, Stage I, phase 1

Figure 8.2 shows the annotated event and time expressions, using the annotation tool.

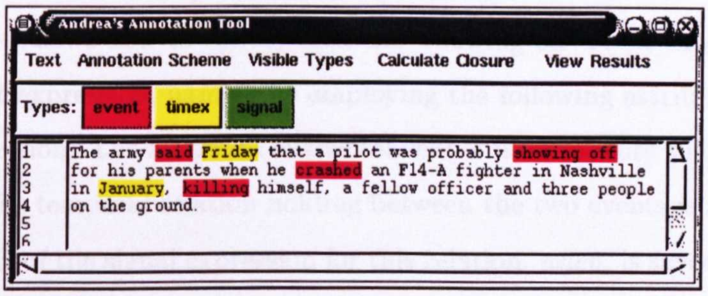

Figure 8.2: Annotated example, event and time expressions only

Phase 2 This stage requires marking up the explicitly expressed temporal relations (see section 6.4) that exist between the event and time expressions annotated in stage 1, including annotating the signals (usually prepositions or subordinate conjunctions) that indicate an explicitly expressed temporal relation. For the example text in figure 8.1, this means relating the two event expressions *showing off* and *crashed* via the temporal subordinate conjunction *when* which indicates their temporal relation. The other temporal relation involves the event expression *crashed* and the time referring expression *January*, signalled by the preposition *in*. Figure 8.3 shows the results of this stage. In addition to what was shown in figure 8.1, signal expressions (dotted circles) and relations (solid arrows) are shown.

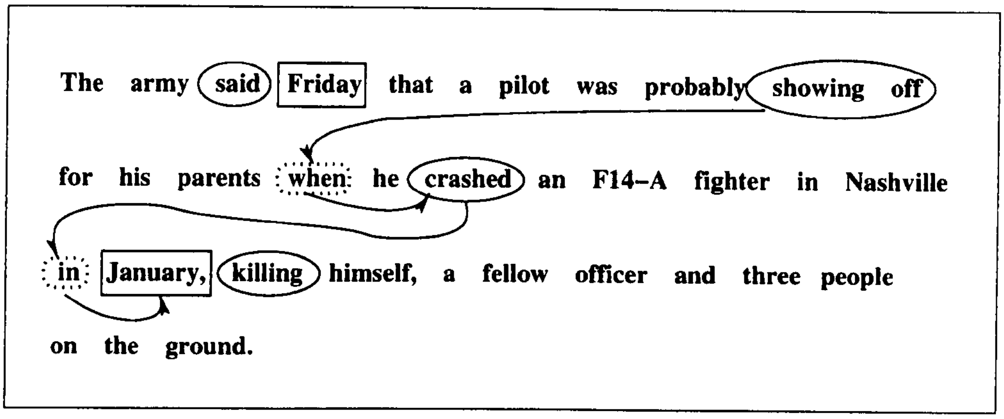

Figure 8.3: Annotation process, Stage I, phase 2

The temporal relations are captured by the annotation scheme as event expression attributes. Figure 8.4 shows how to convey that the 'showing off' event expression is related to the 'crash' event expression, namely by employing the following attributes of the 'showing off' event expression. The attribute relatedToEvent is used to store the ID of the 'crash' event expression, the temporal relation holding between the two events is recorded in eventRelType and the ID of the signal expression for this relation, *when*, is saved in the attribute signalID.

Phase 3 The last stage is the annotation of implicitly expressed temporal relations. This involves, for example, marking up event expressions that are clearly positioned in time but where the temporal relation is not signalled explicitly in the text. The event expression *said* and the time referring expression *Friday* in the example text are clearly related but the usual preposition *on* is omitted. This is a common case when event expressions are implicitly related to referring time expressions. The two events *crashed* and *killing* are related too, but again this relation is not signalled in the text. Figure 8.5 shows the newly added relations as solid arrows, while the previously added relations are now shown as dotted arrows. Using

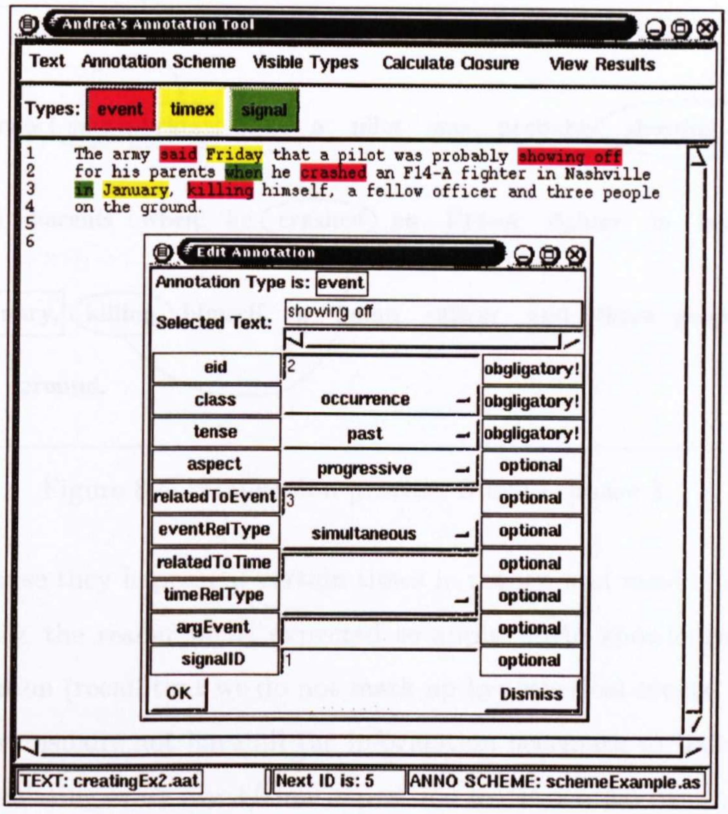

Figure 8.4: Annotated example, including signal expressions and an explicitly expressed temporal relation

the annotation scheme, implicitly expressed temporal relations are captured in the same way as explicitly expressed temporal relations, as shown in figure 8.6. The only difference is that the attribute signalID is left empty.

The first two phases are straightforward since what is to be annotated is signalled in the text. The third phase, however, is more problematic and there are two main reasons for this.

Firstly, semantically identical temporal relations can be marked up in different ways, as explained in section 7.3. This makes comparing annotation more difficult and is also potentially confusing for the annotator.

Secondly, the task for the third phase is not easy to define. Which implicitly expressed relations should be annotated? *All* events and times conveyed in a text are related to each

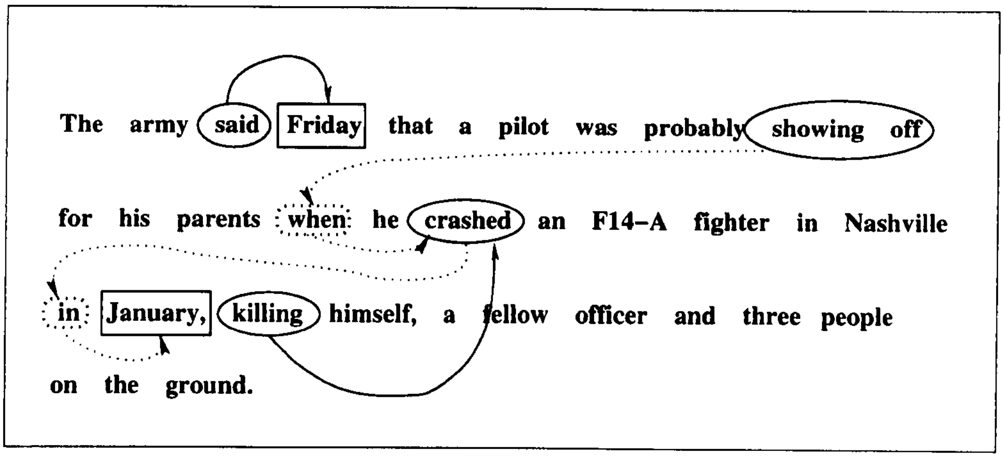

Figure 8.5: Annotation process, Stage I, phase 3

other, simply because they happen at certain times in reality, and most of the relations are expressed implicitly, the reader being expected to apply world knowledge or inference to determine the relation (recall that we do not mark up hypothetical events, see section 6.6). But not only do we usually not have all the information necessary to build a perfect time graph, the task of relating every event/time expression to every other event/time expression in the text would be far to time consuming and difficult for an annotator. Furthermore, it is not even necessary, because sufficient inferences may be drawn from a partial time graph to arrive at the same information (see chapter 7.3).

This means that we cannot and need not relate all event and time expressions in a text. But which ones should be annotated? All temporal relations that are explicitly signalled in the text are straightforward and an obvious choice. So are coreferential events. From then on the choices are less obvious. Explicitly expressed causal and subevent relations come to mind next, but as explained in section 6.6, those are too difficult to annotate as separate relations[2].

This leaves the following recommendations.

[2]Note that even though one does not mark up that two events stand in a causality or subevent relation to each other, it is still possible to annotate the actual temporal relation (includes/is_included for subevents and after/before for causality) as implicit temporal relations.

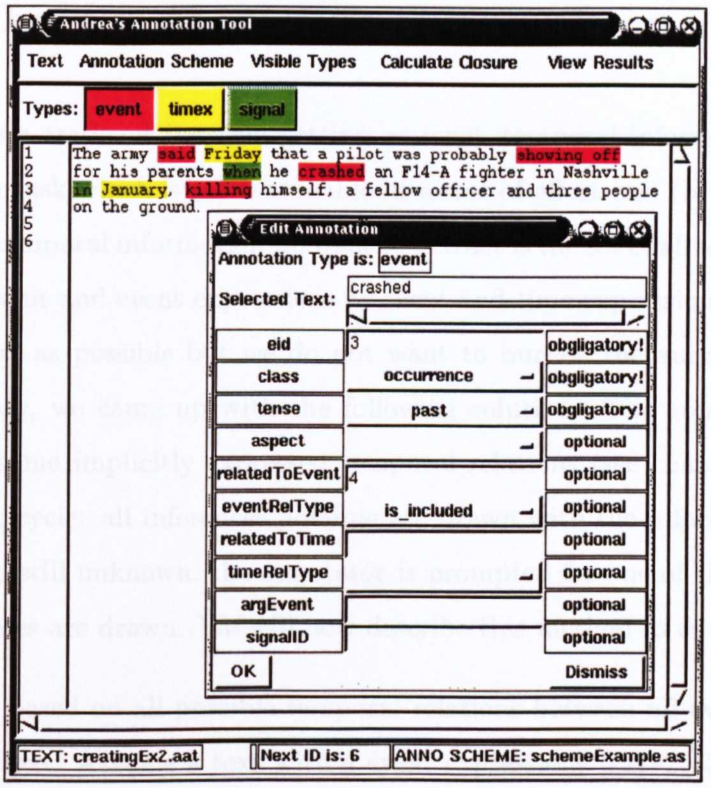

Figure 8.6: Annotated example, including signal expressions and an implicitly expressed temporal relation

- All time referring expressions should be related to some event expression, so expressions like *The navy said Friday* should lead to a temporal relation being annotated. The temporal relation is often that the event is included in the time. The motivation here is that time referring expressions in general only occur in text to position events (there are rare expressions like *the 4th of July 1966 was a lovely day*).
- Constructions like *The plane crashed, killing all passengers* contain two obviously related event expressions and this too should lead to a temporal relation being annotated. The relation is often *simultaneous*.
- Coreferential events should be annotated.

With this, the task of having to annotate all possible temporal relations between events and events or events and times has been reduced to a reasonable task.

8.2 Stage II of the Annotation

Generally, there is a trade-off between getting as much temporal information as possible and an annotation task which is (a) reasonable in terms of effort and (b) replicable across annotations. The temporal information we want to extract is the set of all temporal relations holding between event and event expressions or event and time expressions. Since we want to capture as many as possible but we do not want to burden the annotator with more work than necessary, we came up with the following solution. The annotator marks up all explicitly and some implicitly expressed temporal relations (see chapter 7). Then we enter the following cycle: all inferences possible are drawn with the information provided. If any relations are still unknown, the annotator is prompted for one of these[3] and, again, all possible inferences are drawn. We will now describe this method in detail.

The method is based on all possible temporal relations between all event and all time expressions in the text. Assume a text with 4 event expressions $(e1, e2, e3, e4)$ and 2 time expressions $(t1, t2)$. Potentially, there are $(4+2)\times(4+2) = 36$ temporal relations possible. Some of these are filled in during stage I, as shown in figure 8.7. We indicate a filled entry using a tick whereas unfilled entries are left blank. To make the example easier, we did not include the actual temporal relations. The diagonal in the matrix shown is greyed out, because temporal relations between an entity and itself are irrelevant.

PhaseI All temporal relations annotated in the text are translated into facts of the form `eid:1 SIM eid:3`. The temporal relations are also normalised so that only the relations *before, includes* and *simultaneous* are used. This can lead to some of the entries in the matrix of temporal relations to change position. If the matrix had an entry for `eid:6 AFTER eid:7`, it would now have an entry for `eid:7 BEFORE eid:6` instead. We have omitted this, to keep the description of this stage as simple as possible. Figure 8.7 shows how a temporal relation matrix for the above example may look like. At this point, the relations holding between the annotated times are automatically established by the tool. Recall, that times

[3]Note that *unknown* is a possible value for a temporal relation.

	e1	e2	e3	e4	t1	t2
e1						
e2	✓				✓	
e3						
e4		✓	✓			
t1						✓
t2					✓	

Figure 8.7: Matrix of temporal relations, phase 1

have to be associated with a calender date (see section 7.2.2), so the temporal relations are easily established.

Phase 2: All inferences that can be drawn from the current temporal relations are drawn, which leads to additional relations (see chapter 7.3) and more entries in the matrix being filled in (see figure 8.8).

	e1	e2	e3	e4	t1	t2
e1		✓		✓	✓	
e2	✓			✓	✓	
e3				✓		
e4	✓	✓	✓			
t1	✓	✓				✓
t2					✓	

Figure 8.8: Matrix of temporal relations, phase 2

Phase 3: Until there are no temporally unrelated events in the temporal model of the text do:

1. Draw all inferences that can be drawn from the temporal relations given in the current temporal model of the text according to the temporal inference rules and add them to the temporal model.
2. Identify an unrelated event-event or time-event pair in the temporal model and prompt

the user for the temporal relation (which may be "unknown").

Figure 8.9 shows how an annotator is prompted for a temporal relation by the tool. The

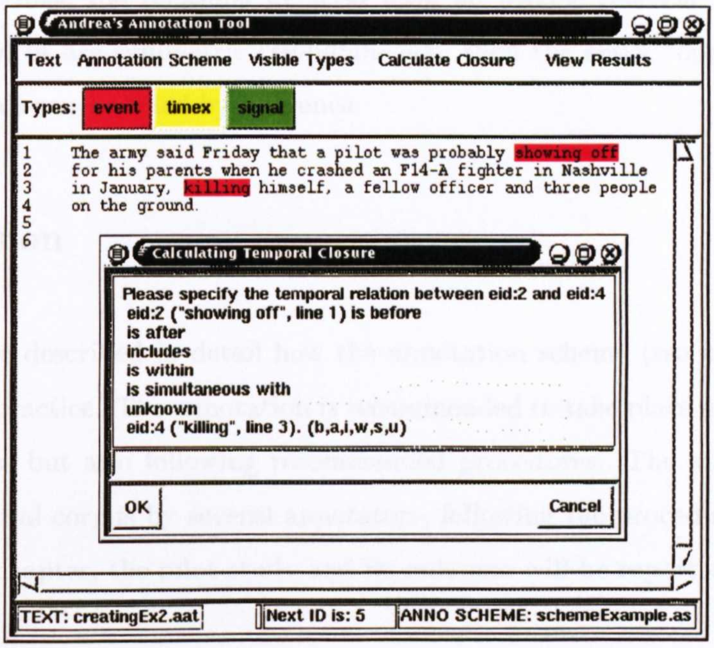

Figure 8.9: Prompting for a temporal relation

matrix is gradually filled until all entries are filled in, unknown relations are indicated by a cross, see figure 8.10.

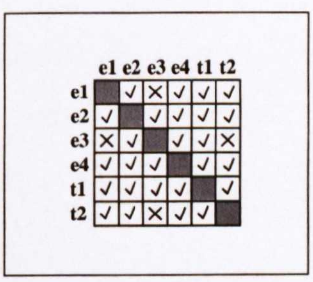

Figure 8.10: Matrix of temporal relations, phase 3

After these two stages of annotation, the result is a temporal model of the text that is as complete as possible.

Note that this is a simplified portrayal. In reality, less than half the matrix needs to be filled in, because the relations we used have an inverse relation (*before/after* and *included/including*) or are symmetric (*simultaneous*), then the entry "opposite" (mirrored across the diagonal) can be filled by inference.

8.3 Discussion

In this chapter, we described in detail how the annotation scheme (see chapter 7) can be applied to text in practice. The annotation is recommended to take place in stages, not only to ease the process but also following recommended procedures. The annotation scheme was applied to a trial corpus by several annotators, following the procedure recommended here. In the next chapter, the pilot study and its outcome will be reported.

Chapter 9

The Trial Corpus

This chapter describes a pilot study in which a group of annotators has been supplied with annotation guidelines for the annotation scheme reported in chapter 7 (the guidelines can be found in appendix A) and asked to apply the annotation scheme to a trial corpus. In particular we were interested in answers to these questions:

- how unambiguous and comprehensive are the annotation guidelines?
- how much genuine disagreement is there about temporal relations in text?
- how burdensome is the annotation procedure? – i.e. is it feasible to think of annotating a corpus of significant size at this level of detail?

This chapter discusses the design of the pilot study, its outcome and the answers to these questions, insofar as we have been able to determine them.

9.1 The Corpus

The trial corpus consists of 6 newswire articles taken from the New York Times, 1996. They were part of the MUC7 (MUC7) training data and chosen because they were available here at the University of Sheffield in machine readable form. Basic statistics about the corpus are presented in table 9.1.

	sentences	words	number of annotators
text1	26	448	3
text2	18	333	2
text3	13	269	3
text4	13	213	2
text5	10	211	3
text6	13	399	3
total	93	1873	3

Table 9.1: The corpus

Each text was annotated by either two or three annotators, in addition to the author who produced what in the following is taken to be the 'gold standard' or 'key' annotation. To produce the gold standard, an initial annotation was created for each text, compared with the annotations provided by the annotators, and then revised if necessary. The results is the 'key' annotation. The annotators were research students and assistants in the NLP research group of the University of Sheffield. They had virtually no training for this particular task, but did have the linguistic knowledge required. They were volunteers who participated in the experiment in their free time.

9.2 Assessing the Annotation Scheme

As mentioned in the introduction to this chapter, the purpose of the pilot study was to ascertain how clearly defined the guidelines were, how much genuine disagreement there is about temporal relations, and how feasible this complex annotation task is. The assessment of the scheme is divided into two parts. In sections 9.2.1 and 9.2.2, the annotations are compared to the 'gold standard'. The results are a measure of how well the scheme was applied and how much agreement there is about events and times as well as temporal relations. In section 9.2.3, interannotator agreement figures are presented, which indicate how well the task is defined.

To compare the annotators' output we have compiled several sets of figures, corresponding to a measure agreement after stage I of the annotation process as described in chapter 8 (i.e. after the initial annotation of events, times, and temporal relations, but prior to any computation of deductive closure) and then after and during stage II (i.e. after and during interactively extending the temporal model in conjunction with the deductive closure computations). We have also compiled a set of figures illustrating interannotator agreement.

We will use the example shown in figure 9.1 throughout this chapter to illustrate our approach. Figure 9.1 shows the key annotation on the left and the response annotation on the right. Note that the results for the example are not representative and serve only to illustrate the process of computing recall and precision.

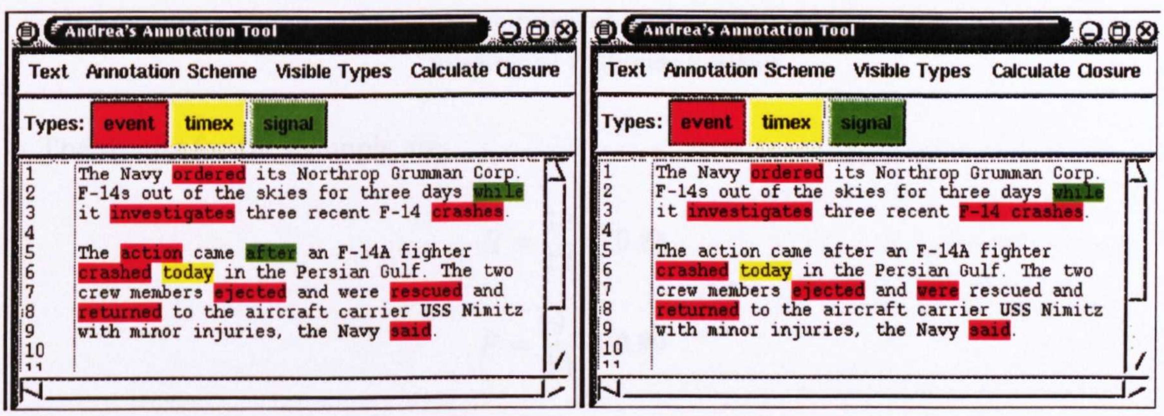

Figure 9.1: Key for the example on the left and response on the right

Please note that when we calculate averages over the corpus we do so by weighting each text equally, we do not weigh texts based on how many entities/attributes/relations they contain.

9.2.1 Stage I – Annotation Only

To measure Stage I agreement we first calculated recall and precision figures for temporal entities (events, times, temporal signal expressions) and then for all attributes associated with matched instances of these entities[1].

Experiments To compute recall and precision for the entities alone, we attempted to match each entity in the response with an entity in the key. Two entities are judged to match if the same text span was annotated with the same type (event, time or signal). We adopted a liberal definition of match where if the response text span included the key text span and was marked as the same type of entity, or vice versa, then this was counted as a match too (see figure 9.1, line 3). The formulae for recall and precision are:

$$R = \frac{\text{number of matches}}{\text{number of key entities}}$$

$$P = \frac{\text{number of matches}}{\text{number of response entities}}$$

The results for the example are:

$$R = \frac{10}{12} = 0.83$$

$$P = \frac{9}{10} = 0.90$$

It is important in this stage to remember that the same entity can have different IDs for different annotators. A translation table from the IDs of a response to the key ID is built during this stage and used in subsequent stages. The translation table for the example is:

response key: eid:1 → key ID eid:1

response key: eid:2 → key ID eid:3

[1]This is similar to scoring matched slots in template objects in the MUC IE tasks, see section 7.3.2 and MUC6; MUC7.

response key: sid:5 $\rightarrow$ key ID sid:2

response key: eid:6 $\rightarrow$ key ID eid:7

response key: tid:7 $\rightarrow$ key ID tid:8

response key: eid:8 $\rightarrow$ key ID eid:9

response key: eid:10 $\rightarrow$ key ID eid:11

response key: eid:11 $\rightarrow$ key ID eid:12

To obtain results for the attributes, we separately counted all attributes for each entity in the key and in the response. For the entities in the response that matched entities in the key we counted the matching attributes. Since the IDs of the entities do not contribute any temporal information, event expression IDs (event ID (eid), time expression ID (tid) and signal expression ID (sid)) were not included when counting the matches. The formulae are:

$$R = \frac{\text{number of matching attributes}}{\text{number of key attributes}}$$

$$P = \frac{\text{number of matching attributes}}{\text{number of response attributes}}$$

Figure 9.2 shows the attributes for the *ejected* event on line 7, which is annotated by both the key and the response. All attributes except aspect match for this entity and thus it contributes 8 positive matches towards recall and precision. Overall there are 65 matching attributes for the 9 matching entities. The key contains 86 and the response 77 attributes altogether. This yields the following results:

$$R = \frac{65}{86} = 0.75$$

$$P = \frac{65}{77} = 0.84$$

When we count all attributes, we also count those attributes that are not filled in. For example the attributes argEvent, aspect, relatedToTime, relatedToEvent etc. are not necessarily filled in for each event. If both key and response have not filled an attribute

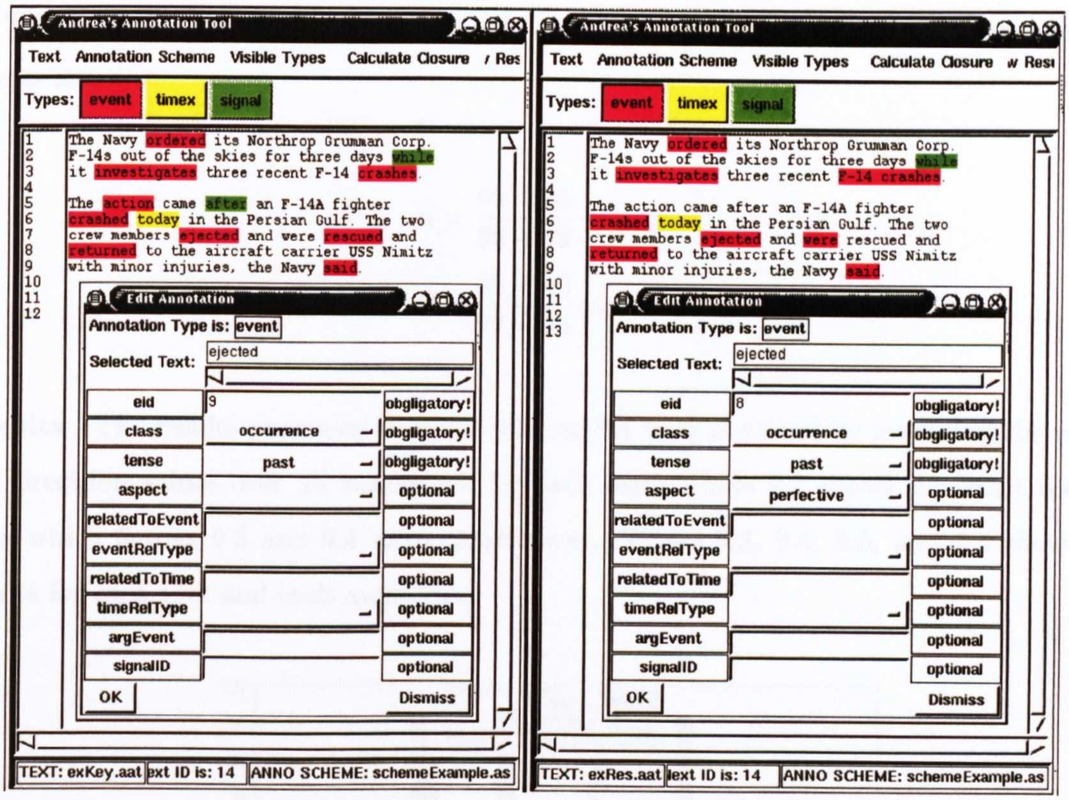

Figure 9.2: Key on the left and response on the right

in then it is questionable whether this is a positive match or this should not contribute to the results. If we look at figure 9.2 then we can see that the attributes for relatedToEvent, eventRelType, relatedToTime, timeRelType, argEvent and signalID are not filled in for either key or response. To see the effect when non-filled attributes were taken out of the equation, we computed a second set of results for recall and precision, where we did not count the attributes that are not filled for all responses and the key (we call those non-filled attributes in the formulae below). In other words, an attribute has to filled in by at least one of the responses or the key to be counted. This method guarantees that same number of attributes is compared when computing recall and precision, for each annotation. The formulae are:

$$R = \frac{\text{number of matches} - \text{non-filled attributes}}{\text{number of key entities} - \text{non-filled attributes}}$$

$$P = \frac{\text{number of matches} - \text{non-filled attributes}}{\text{number of response entities- non-filled attributes}}$$

The example contains 46 attributes that are filled for neither key nor response and the results are:

$$R = \frac{65 - 46}{86 - 46} = 0.48$$

$$P = \frac{65 - 46}{77 - 46} = 0.62$$

Results The results presented in figure 9.3 and 9.4 were obtained by averaging the recall and precision values over all annotators for each text. Table 9.2 shows the exact results from which figures 9.3 and 9.4 were constructed. Tables 9.3, 9.4, 9.5, and 9.6 show the results for each text and each annotator.

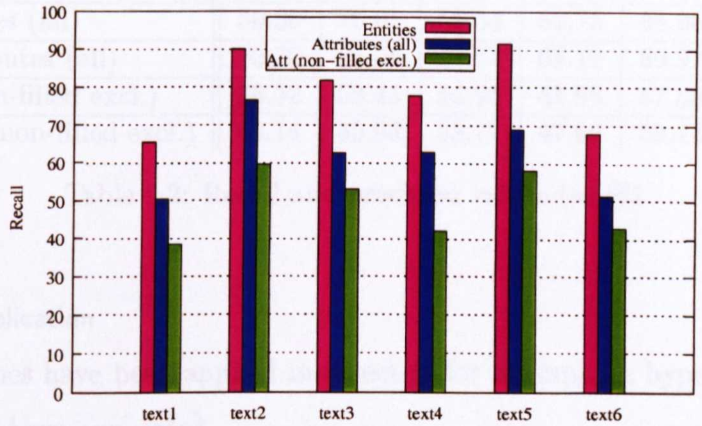

Figure 9.3: Recall for stage I of the annotation

Analysis When we look at the errors made for the entities only then we can broadly distinguish two groups. Entities that were missed out (and thus affect the recall) and entities that were annotated in addition to the entities in the key (affecting the precision). We can further subclassify additional entity mistakes as follows:

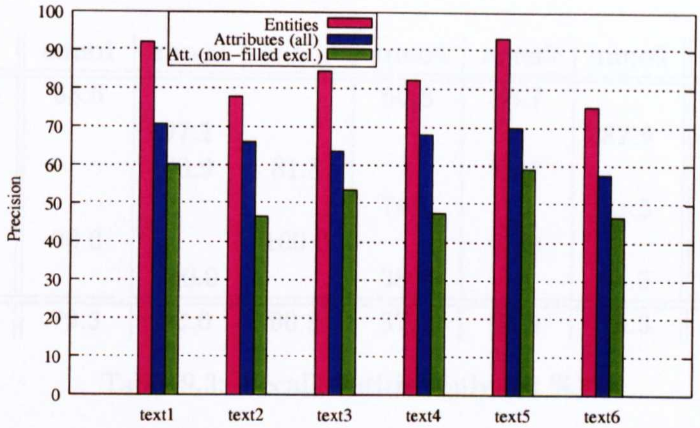

Figure 9.4: Precision for stage I of the annotation

	text1	text2	text3	text4	text5	text6	avg
Recall Entities	65.36	90.00	81.82	77.78	91.67	67.74	79.06
Precision Entities	92.05	77.87	84.68	82.31	93.36	75.41	84.28
Recall Attributes (all)	50.36	76.61	62.54	62.75	68.90	51.22	62.07
Precision Attributes (all)	70.70	66.06	63.74	68.12	69.97	57.58	66.03
Recall Att. (non-filled excl.)	38.32	59.41	52.93	41.98	57.85	42.83	48.89
Precision Att. (non-filled excl.)	60.14	46.63	53.77	47.66	59.14	46.70	52.34

Table 9.2: Recall and precision values (as %)

- guideline application

  The guidelines have been applied incorrectly; for example, a hypothetical event expression has been annotated.

- not anchored in time

  The time or event expressed is not anchorable in time. The time expression is either non-referring (see section 3.3), or cannot be placed on a time line due to lack of contextual information. The expression should not be marked up according to the guidelines, but has been annotated despite this. In the example *Macau authorities, who lost radar contact at 10 am...*, the time *10 a.m.* is not anchored in time, because the article does not provide the contextual knowledge necessary to establish which day this time of day belongs to. Nevertheless, it has been annotated by some annotators.

  This is a special case of guideline application, but occurs so frequently in the corpus,

	anno1	anno2	anno3	anno4	anno5	anno6	avg.
text1	68.6			60.8	66.7		65.4
text2		97.1				82.9	90.0
text3		90.9	81.8		72.7		81.8
text4				74.1		81.5	77.8
text5	90.0		100.0		85.0		91.7
text6		100.0		38.7		64.5	67.7
avg.	79.3	96.0	90.9	57.9	74.8	76.3	

Table 9.3: Recall, entities only (as %)

	anno1	anno2	anno3	anno4	anno5	anno6	avg.
text1	97.2			93.9	85.0		92.1
text2		85.0				70.7	77.9
text3		83.3	81.8		88.9		84.7
text4				80.0		84.6	82.3
text5	94.7		90.9		94.4		93.4
text6		81.6		70.6		74.1	75.4
avg.	96.0	83.3	86.4	81.5	89.4	76.5	

Table 9.4: Precision, entities only (as %)

that it is reported independently.

- wrong text span

  It is clear what time or event expression was intended, but the text span annotated for this time or event differs too much from the text span in the key and is thus not recognised as the same entity by the scoring software (Note: such an error leads to a missing event/time and an additional event/time. We have left figures for both in the tables to indicate what proportion of the overall errors these form). You can see in the worked example in figure 9.1 that the *rescued* event expression is marked up in the response, but was annotated as *were* and not as *rescued.* Neither response nor key text span contains the other and so the scoring software cannot recognise that essentially the same event expression was meant.

	anno1	anno2	anno3	anno4	anno5	anno6	avg.
text1	52.0			50.4	48.7		50.4
text2		88.5				64.8	76.6
text3		78.5	58.6		50.5		62.5
text4				52.3		73.2	62.8
text5	74.4		69.5		62.8		68.9
text6		79.5		27.8		46.3	51.2
avg.	63.2	82.2	64.1	43.5	54.0	61.4	

Table 9.5: Recall all attributes (as %)

	anno1	anno2	anno3	anno4	anno5	anno6	avg.
text1	74.0			80.1	58.0		70.7
text2		76.8				55.4	66.1
text3		71.6	58.6		61.0		63.7
text4				57.5		78.8	68.1
text5	76.7		62.6		70.5		70.0
text6		64.8		54.1		53.8	57.6
avg.	75.3	71.0	60.6	63.9	63.2	62.6	

Table 9.6: Precision all attributes (as %)

And, missing entity mistakes can subclassified as:

- different opinion

  It is arguable, given the guidelines, whether or not an event or time expression should be annotated. Annotators had different opinions about whether *persuaded* conveys an event or not in the following sentence: *They were partly persuaded by a surprising discovery found ...*

- wrong

  An entity was not annotated, without an apparent reason.

- wrong text span

  As above.

	Times	Events
not anchored in time	71.43% (10 out of 14)	47.06% (32 out of 68)
wrong text span	28.57% (4 out of 14)	14.70% (10 out of 68)
guideline application	0% (0 out of 14)	38.24% (26 out of 68)

Table 9.7: Mistakes for additional entities

	Times	Events
different opinion	35.29% (6 out of 17)	12.27% (13 out of 106)
wrong	41.19% (7 out of 17)	78.30% (83 out of 106)
wrong text span	23.53% (4 out of 17)	9.43% (10 out of 106)

Table 9.8: Mistakes for missing entities

Tables 9.7 and 9.8 show the distribution of mistakes made, summed across annotators and distinguishing between mistakes made for times and those for events.

Examining the subgroups, it becomes evident that some of the mistakes could easily be rectified with more training of the annotators and an improved version of the guidelines. Wrong text span, guidelines, including not anchored, and wrong fall under this category, which covers 90.73% of all errors. More difficult is the category opinion, where there is genuine ambiguity about the right results. This category covers the remaining 9.27% of all errors.

As mentioned before, we take into account that IDs may differ from annotator to annotator and we do not compare entity IDs when comparing attributes. What we do not take into account, however, is the fact that the same relation can be expressed in different ways. If the two events 5 and 7 are related, then this can be expressed in two different ways. Assume that event 5 is before event 7. Then we can encode this as 5 before 7 and have the following (excerpt of) attributes for event 5 and 7:

eid = 5	eid = 7
relatedToTime = 7	relatedToEvent =
timeRelType = before	timeRelType =

Or we could encode this as 7 after 5 and the attributes for the events would be:

eid = 5	eid = 7
relatedToTime =	relatedToEvent = 5
timeRelType =	timeRelType = after

Imagine the response would be the former and the key the latter. Then we would not match a single attribute for the example. The temporal closure phase described later on takes these cases into account, but the results for the attributes have to be interpreted with this in mind. What we did, however, was to compute a different set of results where we only took those attributes into account that are not related to temporal relations (class, tense, aspect and argEvent); the results are shown in figures 9.5 and 9.6. We included the results for entities and all attributes for comparison. Tables 9.9 and 9.10 contain the figures from which these figures were constructed.

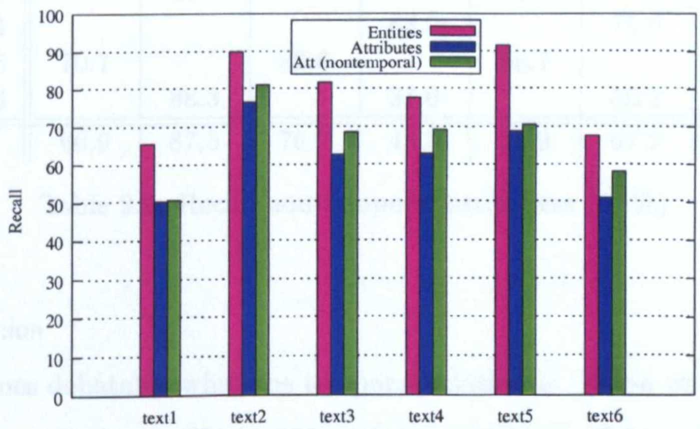

Figure 9.5: Recall for stage I of the annotation, non-temporal attributes

Looking at the attribute errors, we can distinguish the following groups:

- missing
  An attribute was not filled in.
- wrong
  An attribute is present but incorrect.

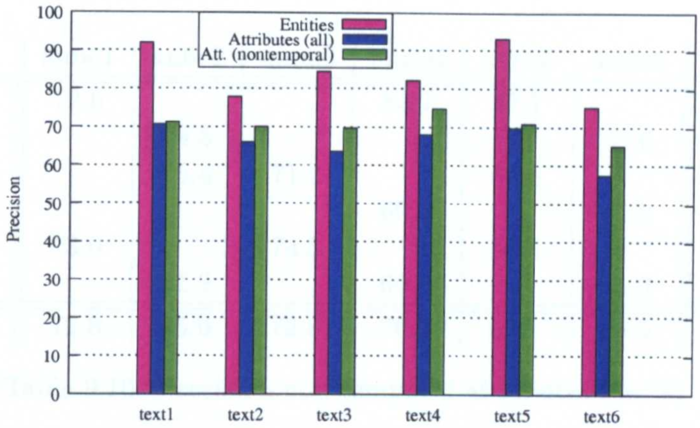

Figure 9.6: Precision for stage I of the annotation, non-temporal attributes

	anno1	anno2	anno3	anno4	anno5	anno6	avg.
text1	51.1			52.8	48.1		50.6
text2		91.5				70.9	81.2
text3		82.7	71.2		51.9		68.6
text4				63.0		75.6	69.3
text5	70.7		82.6		58.7		70.7
text6		88.3		31.0		55.2	58.2
avg.	60.9	87.5	76.9	48.9	52.9	67.2	

Table 9.9: Recall non-temporal attributes (as %)

- different opinion

  It is sometimes debatable what the temporal relation is. When we take the following sentence, for example: ... ***Flight 800 exploded in midair off Long Island and plunged into the Atlantic Ocean, killing all 230 people on board*** then it is uncertain whether the *killing* event was simultaneous, during or after the *explosion*.

- additional

  Implicit temporal relations can be filled in, which results in attributes not present in the key. This is not actually a mistake, but the automatic recognition of mismatches cannot distinguish between genuine mistakes and additional information. Looking at the example in the previous point, then the temporal relation between *exploded* and *plunged* does not need to be filled in according to the guidelines. It is, however, not

	anno1	anno2	anno3	anno4	anno5	anno6	avg.
text1	72.6			83.7	57.7		71.3
text2		79.5				60.6	70.0
text3		75.4	71.2		62.8		69.8
text4				69.1		81.0	75.0
text5	73.0		74.5		65.8		71.1
text6		71.9		60.0		64.0	65.3
avg.	72.8	75.6	72.8	70.9	62.1	68.5	

Table 9.10: Precision non-temporal attributes (as %)

wrong to fill in that the *plunge* event happened after the *explosion*.

- guideline application

  The guidelines were not applied correctly. For example, for a reporting event the argument event should be entered into the argEvent slot, but was sometimes was entered into the relatedToEvent slot.

The distribution of the attribute errors is shown in table 9.11.

different opinion	missing	wrong	additional	guidelines
5.34%	26.70%	35.11%	13.43%	19.79%
33 out of 618	165 out of 618	217 out of 618	83 out of 618	120 out of 618

Table 9.11: Mistakes for attributes

9.2.2 Stage II – Temporal Closure

Experiments To measure Stage II agreement, we looked at agreement on temporal relations after the three different phases mentioned earlier: after the temporal relations in the text had been extracted and normalised, after computing the deductive closure over these facts and after additional information from the annotator has been requested and the deductive closure had been computed. Recall, that the idea behind the temporal closure was to compensate for semantically identical but syntactically different temporal relations. We expect recall and precision to rise after the normalisation, because we expect annotators to mark up the same temporal relations in different, but semantically identical, ways. Before normalisation, these relations are not matched but after normalisation they are recognised as identical and contribute to both recall and precision.

We also expect recall and precision to rise after the next step. Only a certain amount of the temporal relations holding between entities in a text are caught by the annotation scheme. The 'rest' of these relations is caught by the interactive phase. We assume that, overall, annotators agree about temporal relations and so we assume that the interactive phase will contribute more 'matching' temporal relations and thus increase both recall and precision.

We will now describe, how recall and precision are computed for this stage. Both key and response go through the three phases of this stage in parallel.

In phase 1, all temporal relations in the key and the response are normalised and translated into facts of the form `eid:3 BEFORE eid:5`. In the worked example, the event 7 is related to time 8 and the (relevant) attributes look like this:

eid = 7
relatedToEvent = 8
eventRelType = before

This relation leads to the fact `eid:7 BEFORE eid:8`. As explained in chapter 7.3, the relations after and is_included are normalised to BEFORE and INCLUDES respectively and the relation identity is mapped to SIMULTANEOUS. Thus had the relation above been encoded in the attributes of event 8 and had looked like this:

eid = 8
relatedToEvent = 7
eventRelType = after

then this still would have led to the same fact being included in the temporal closure.

To compute recall we attempt to match each fact for the response to a fact in the key, employing the translation table for IDs, as mentioned above.

The temporal relations for the example text for key and response at the before phase 1 are:

key	response
e1 is_included e3	e1 is_included e2
e5 identity e1	
e5 after e7	
e7 is_included t8	e6 is_included t7

After the relations have been normalised and the response IDs have been mapped onto key IDs, the facts look like this:

key	response
e3 INCLUDES e1	e3 INCLUDES e1
e5 SIMULTANEOUS e1	
e7 BEFORE e5	
t8 INCLUDES e7	t8 INC e7

The recall is 50% and the precision 100%.

After phase 2 (computing the deductive closure over those relations, for both the key and the response), the relations are:

key	response
e3 INCLUDES e1	e3 INCLUDES e1
e3 INCLUDES e5	e3 INCLUDES e5
e5 SIMULTANEOUS e1	
e7 BEFORE e5	
e7 BEFORE e1	
t8 INCLUDES e7	t8 INCLUDES e7

Recall and precision are the same as before and phase 3 has been omitted for the example for brevity.

Recall that we do not want to place reporting events on the time graph (see section 6.6). They are automatically taken out before step I commences, unless they are explicitly related to a time.

Results Figures 9.7 and 9.8 show the recall and precision values for the second stage of the annotation. Table 9.12 shows the exact results from which figures 9.7 and 9.8 were constructed and tables 9.13, 9.14, 9.15, 9.16, 9.17, and 9.18 show the results for each text and each annotator and each phase.

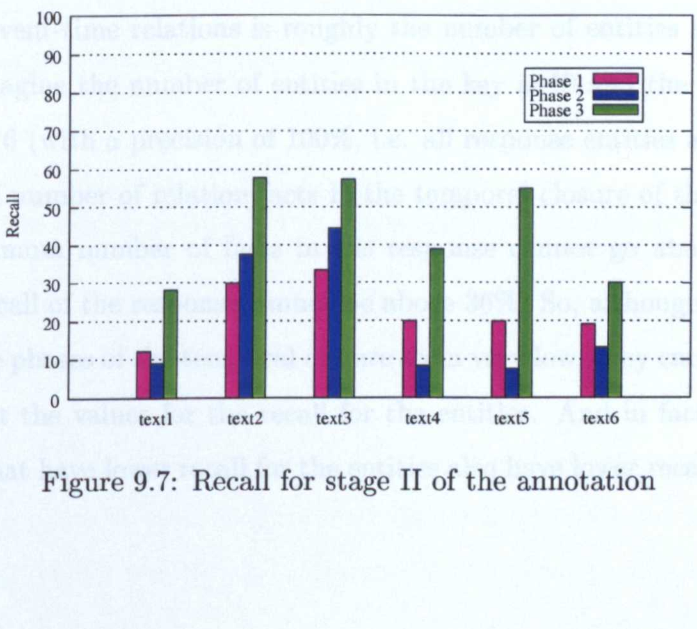

Figure 9.7: Recall for stage II of the annotation

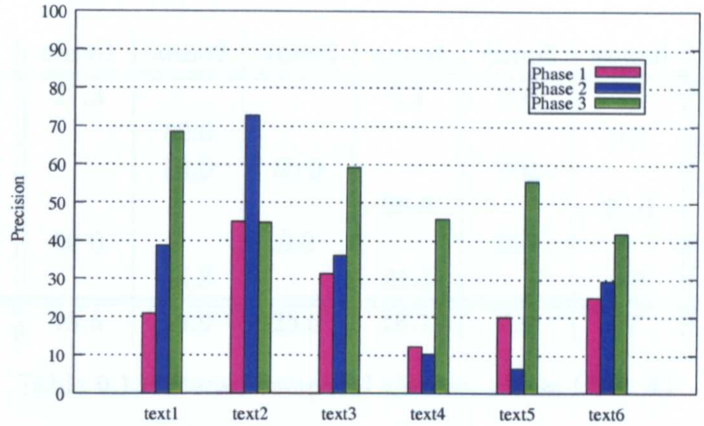

Figure 9.8: Precision for stage II of the annotation

	text1	text2	text3	text4	text5	text6	avg
Recall Phase1	12.12	30.00	33.33	20.00	20.00	19.30	22.46
Precision Phase1	20.94	45.00	31.34	12.18	20.00	25.00	25.74
Recall Phase2	9.03	37.50	44.45	8.33	7.58	13.18	20.01
Precision Phase2	38.62	72.88	36.00	10.26	6.40	29.35	32.25
Recall Phase3	28.09	57.67	57.24	38.86	54.90	30.16	44.49
Precision Phase3	68.51	44.71	59.08	45.73	55.61	41.85	52.58

Table 9.12: Recall and precision values (as %)

Analysis Note that there is a strong correlation between the recall of temporal relations following stage II and the recall of temporal entities following stage I only. The number of event-event and event-time relations is roughly the number of entities squared. Thus, for example, if we imagine the number of entities in the key is 10 and the number of entities in the response is 6 (with a precision of 100%, i.e. all response entities are also in the key) then the potential number of relation facts in the temporal closure of the key is $10^2 = 100$ whereas the maximum number of facts in the response cannot go above $6^2 = 36$. This means that the recall of the response cannot be above 36%. So, although the values for the recall for the three phases of the temporal closure seem very low, they cannot be interpreted without looking at the values for the recall for the entities. And in fact, one can observe that those texts that have lower recall for the entities also have lower recall for the temporal closure.

	anno1	anno2	anno3	anno4	anno5	anno6	avg.
text1	27.3			9.1	0.0		12.1
text2		60.0				0.0	30.0
text3		50.0	50.0		0.0		33.3
text4				20.0		20.0	20.0
text5	60.0		0.0		0.0		20.0
text6		36.8		21.1		0.0	19.3
avg.	43.6	48.9	25.0	16.7	0.0	6.7	

Table 9.13: Recall temporal closure, phase I (as %)

	anno1	anno2	anno3	anno4	anno5	anno6	avg.
text1	11.8			15.3	0.0		9.0
text2		67.1				7.9	37.5
text3		81.8	51.5		0.0		44.4
text4				8.3		8.3	8.3
text5	18.2		4.5		0.0		7.6
text6		27.1		12.4		0.0	13.2
avg.	15.0	58.7	28.0	12.0	0.0	5.4	

Table 9.14: Recall temporal closure, phase II (as %)

There are different types of facts. First, there are the facts that are produced based on the relations encoded in the text. Second, there are the facts solicited from the user; we call both classes user supplied. The third type are the facts that are triggered by the inference rules. The 'user supplied' errors can be divided into these classes:

- wrong

  Caused by tiredness, for example, a wrong relation can be entered.

- different opinion

  As before, the temporal relation holding between two entities is far from unambiguous.

Missing is not part of these errors, because missing temporal relations only occur in phase one. These missing relations are the ones that the annotator did not include in the annotation. But these errors are already accounted for in the previous section, where we

	anno1	anno2	anno3	anno4	anno5	anno6	avg.
text1	31.7			24.5	28.0		28.1
text2		74.2				41.2	57.7
text3		72.7	60.6		38.4		57.2
text4				27.1		50.6	38.9
text5	69.8		55.9		39.0		54.9
text6		61.0		8.2		21.3	30.2
avg.	50.8	69.3	58.2	19.9	35.1	37.7	

Table 9.15: Recall temporal closure, phase III (as %)

	anno1	anno2	anno3	anno4	anno5	anno6	avg.
text1	46.1			16.7	0.0		20.9
text2		90.0				0.0	45.0
text3		55.6	38.5		0.0		31.3
text4				7.7		16.7	12.2
text5	60.0		0.0		0.0		20.0
text6		35.0		40.0		0.0	25.0
avg.	53.1	60.2	19.2	21.5	0.0	5.6	

Table 9.16: Precision temporal closure, phase I (as %)

computed recall and precision figures for events and times. In the following phases, all missing relationships are prompted for and the annotator cannot skip them.

The 'closure driven' errors can be classed into:

- wrong inference

  As explained in chapter 7.3, the inference rules do not take into account the 'fuzzy' nature of the simultaneous relation and could potentially result in wrong facts being inserted.

- consequence

  If the sets of temporal facts for key and response at the outset of stage I are not the same then the inferences based on that set can be different.

	anno1	anno2	anno3	anno4	anno5	anno6	avg.
text1	64.6			51.3	0.0		38.6
text2		97.9				47.8	72.9
text3		71.0	37.0		0.0		36.0
text4				3.9		16.7	10.3
text5	17.4		1.8		0.0		6.4
text6		38.0		50.0		0.0	29.3
avg.	41.0	69.0	19.4	35.0	0.0	21.5	

Table 9.17: Precision temporal closure, phase II (as %)

	anno1	anno2	anno3	anno4	anno5	anno6	avg.
text1	79.6			72.2	53.7		68.5
text2		59.2				30.2	44.7
text3		62.3	60.6		54.3		59.1
text4				22.6		68.8	45.7
text5	55.2		66.7		44.9		55.6
text6		55.2		33.3		37.0	41.9
avg.	67.4	58.9	63.6	42.7	51.0	45.4	

Table 9.18: Precision temporal closure, phase III (as %)

The sheer volume of temporal facts for each text meant that it was impossible to handcraft a table for the distribution of errors into those classes.

9.2.3 Interannotator Agreement

Interannotator agreement (IAA) is commonly used to assess how well defined an annotation scheme is (see Hirschman et al., 1998). The Kappa statistic is widely used as a measure of IAA for corpus annotation tasks which are classification tasks (e.g. tagging) (Carletta, 1996; Siegel and Castellan, 1988). However, our annotation task cannot easily be construed as a classification task and thus the Kappa statistic is not an appropriate measure for IAA in this case.

What we have done instead, is to follow the example set by Hirschman et al. (1998)

and calculate interannotator agreement for each text in the following way. For each text, we used one annotator as the key and the other(s) as the response and calculated recall and precision for the entities only and for the attributes. The average of all pairwise combinations of annotators over the texts can be seen in table 9.19. Note that the corpus consisted of only 6 texts, each annotated by 2 or 3 annotators, and the small size of the corpus should be taken in to account when interpreting the figures.

text	recall entities	precision entities	recall attributes	precision attributes	number of annotators
text1	77.50	80.82	59.84	60.03	3
text2	76.56	76.56	62.19	62.19	2
text3	77.26	78.77	55.14	55.27	3
text4	86.31	86.31	64.21	64.21	2
text5	90.21	91.09	62.34	62.42	3
text6	67.24	67.24	44.83	46.50	3
average	79.18	80.13	58.09	68.80	

Table 9.19: Interannotator agreement recall and precision in percent

The figures confirm that the annotation guidelines need improving before the scheme is applied to a larger corpus. After the improvements described in the previous section are in place, interannotator agreement has to be recalculated and analysed.

According to the organisers of the Message Understanding Conferences (Hirschman et al., 1998), the figures for IAA have to be at least in the low 90s, before the task can be said to be well defined. Hence, as the figures in table 9.19 show, higher IAA has to be achieved before the annotation scheme can be applied to a larger corpus.

It is important to understand why the annotators disagree, to be able to improve IAA results as well the results reported in sections 9.2.1 and 9.2.2. This will be discussed in detail in the following section.

9.2.4 Analysis and Improvements

Analysis

The situation for the experiments was not ideal. There was not enough time to repeat the experiments and refine the guidelines or train the annotators. There were not enough resources to build a bigger corpus, as we had to rely on volunteers doing the experiment in their spare time.

Nevertheless, the corpus is indicative and we can use it to answer the questions raised at the beginning of this chapter.

- *How unambiguous and comprehensive are the annotation guidelines?*

  The overall figures for recall and precision for stage I annotations are:

  - for the entities only: 76.8% recall and 81.4% precision
  - for the attributes: 59.1% recall and 62.4% precision
  - for non-temporal attributes: 63.8% recall and 67.2% precision

  These figures indicate that there is a consensus about what the entities in a text are and what their attributes are. These figures, as well as the interannotator agreement figures, show that the guidelines (amongst other parameters) need improving.

- *How much genuine disagreement is there about temporal relations in text?*

  More research is needed to answer this question. We know, for example, that there is a strong relation between the recall of entities in stage I and the recall of phase in in stage II, but this needs investigating further.

- *How burdensome is the annotation procedure? – i.e. is it feasible to think of annotating a corpus of significant size at this level of detail?*

  According to the annotators, the annotation phase was difficult but is feasible – with some improvements to the guidelines and the tool (see below). The temporal closure phase is a different matter. The number of questions asked per text varied from 16 to 204 (62 on average) and is was very tiring for the annotators to answer that many questions. Before this has been improved, the annotation of a larger corpus would not be recommended.

Improvements

It became evident that there are several areas for improvement, detailed below.

Entities and attributes The annotation of entities and attributes would benefit from:

- An automatic pre-annotation of time expressions. Most time expressions are not very difficult to detect in a text and are usually uncontroversial.
- A control mechanism to prevent identical IDs in different entities and entering non-existent IDs into attributes like relatedToEvent. Related to this would be an automatic allocation of IDs.
- Finite verbs could be highlighted automatically so that the annotator would only have to confirm the event and fill in the attributes. This might possibly be extended to nominalisations and non-finite verbs.
- Signals are usually prepositions or temporal subordinate conjunction and it could prove helpful to point the annotator to signals that were not annotated or related.
- Indicate signal or time expressions that have not been related to an event expression.

Temporal closure Possibilities for improving stage II of the annotation include:

- Validity Check
  There is no validity check implemented for the temporal closure. At the moment, the only check is that no temporal relation can be entered twice. It is, however, possible to enter a temporal relation which can lead to contradictory relations being filled in.

- Error Correction

  When the annotator is asked for unknown temporal relations in step III, and realises that she or he made a mistake earlier on, then there is no option to correct this mistake. The closure has to be started all over again. An important improvement would be to build in such a backtracking capability. An alternative would be to include the possibility to save intermediate stages during the closure ("check-pointing") so that the annotator could resume the activity from an earlier 'error free' stage. We would not recommend using this feature to take a break, however, because the task is a very complex one and a longer break in this phase would very likely lead to confusion and wrong temporal relations being entered.

- A more intelligent way to solicit relations from the annotator might be found so that questions that maximise the number of inferences that can be automatically drawn from the solicited relation are prioritised. The following simple example illustrates the affect non-optimal soliciting can have. Imagine four events, forming a 'precedence chain':

  $$e_1 < e_2 < e_3 < e_4$$

  Imagine also that the link between e_2 and e_3 is missing in the response:

  $$e_1 < e_2 \qquad e_3 < e_4$$

  If the first question establishes the temporal relation holding between e_2 and e_3, then all other temporal relations can be inferred, based on the transitivity of before. The temporal model can be completed with one question. However, the order of questioning could be very different, establishing the temporal relations between e_1 and e_4, then between e_1 and e_3, e_2 and e_4 and then between e_2 and e_3. Four question are asked to establish the relations holding between the events.

More training of the annotators is like to be beneficial for both stages of the annotation.

The Corpus To confirm the results a bigger corpus would need to be annotated by more annotators.

9.3 Discussion

In this chapter, we have presented figures for recall and precision for both stages of the annotation scheme. The figures show that there is an overall consensus about the entities (events, times and signals) that have to be annotated and, to a lesser extent, about the attributes of these entities. Nevertheless, the annotation guidelines and the annotation tool developed need to be improved and we have pointed out avenues of improvement.

We also talked about the interannotator agreement achieved, which confirms that higher recall and precision scores have to be achieved before the annotation scheme can be applied to a larger corpus.

Chapter 10

Temporal Phenomena

In the previous chapters, we have described a temporal annotation scheme and how it was applied to real text to create a trial corpus. In this chapter, we will describe the insight we have gained from the trial corpus into temporal phenomena in text.

In chapter 3, the mechanisms used to convey temporal information in language were described and the first part of this chapter will show the distribution of these mechanisms in the trial corpus. The second part of this chapter looks at the temporal information contained in a text, how it can be measured, how far we have been able to capture it with the annotation scheme introduced in chapter 7, and how the approach introduced in this thesis can be used to answer questions such as *What proportion of the temporal information is covered by explicitly expressed temporal relations in a text?*.

This investigation reported in this chapter is of a preliminary nature, due to the size of the trial corpus. The corpus, which has been described in section 9.1, contains 6 texts, and the results have to be viewed with this in mind. However, we believe that this study is an indicative one if not a definitive one, and it shows what could be done with a larger corpus.

All results in this chapter are based on the 'gold standard' or key annotation of the texts.

10.1 The Distribution of Mechanisms

First of all, we were interested in the number of event expressions, time expressions and expressions signalling temporal relations in each text (called *signal expressions* from now on). On average, a text in our corpus contains 15.5 sentences and 312 words, with 26.5 event expressions and just under 5 time and signal expressions each. Table 10.1 shows the figures for each text.

	number of sentences	number of words	number of event expressions	number of time expressions	number of signal expressions
text1	26	448	40	11	10
text2	18	333	30	5	3
text3	13	269	19	3	7
text4	13	213	27	0	1
text5	10	211	16	4	3
text6	13	399	26	5	4
total	93	1873	158	28	28

Table 10.1: Number of event, time and signal expressions per text

Events can be expressed in language using finite clauses, nominalisations or non-finite clauses (see chapter 3). We were interested to examine to what extent each mechanism was used to convey time, so that future research can concentrate on the most dominant mechanism. Table 10.2 shows that the majority of the events are conveyed by finite clauses – 72.3%. Nominalisations are employed in 18.2% of the cases and non-finite clauses 9.5% of the time. All three mechanisms cover a significant enough portion of events to justify future research into them.

Events are classified into occurrence, reporting, perception and aspectual events. Table 10.3 shows the distribution of event classes in the corpus.

As explained in chapter 3, the temporal relations holding between events and events or

	number of finite verbs	number of nominalisations	number of non-finite verbs
text1	31	8	2
text2	20	7	3
text3	15	2	2
text4	21	5	1
text5	13	0	3
text6	15	7	4
total	115	29	15

Table 10.2: Distribution of mechanisms used to convey events

	occurrence	aspectual	perception	reporting	reporting associated with a time expr.
text1	32	0	0	8	0
text2	26	0	0	4	0
text3	16	0	0	3	1
text4	18	0	0	9	0
text5	9	0	1	6	0
text6	23	1	0	2	0
total	124	1	1	32	1

Table 10.3: Distribution of event classes

events and times do not necessarily have to be signalled explicitly in text. To assess the importance of implicitly expressed temporal relations, we looked at the extent to which they are used in the corpus. The implicitly expressed relations taken into account are those that are annotated, following the recommendations laid out in section 8.1. The recommendations include the annotation of event identity, cases where a time referring expression is related to an event but the usual temporal preposition signalling the relations is omitted, and cases of event-event relations like *the plane crashed, killing all passengers on board.* As can be seen in table 10.4, more than twice as many annotated temporal relations were expressed implicitly than were explicitly, showing that the method of expressing temporal relations implicitly is important.

In section 3.5, we talked about event identity being widely used in newspaper articles, where events are introduced and then come back to later to give more detailed information about them. Table 10.4 confirms the relevance of event identity, with 48.3% of the implicitly expressed relations being event identity - which is 32.6% of all temporal relations expressed text. This is clearly a significant portion, which justifies including event identity an a temporal annotation scheme aimed at newspaper texts.

	number of explicitly expressed temporal relations	number of implicitly expressed temporal relations annotated in the texts	number of event identity relations included in the implicitly exp. rel.
text1	10	15	6
text2	3	14	6
text3	7	5	5
text4	1	6	4
text5	3	2	1
text6	4	16	6
total	28	58	28

Table 10.4: Explicitly and implicitly expressed relations

To give an idea of the scale of the temporal relations associated with a text, and how many are annotated, inferred or solicited, we included table 10.5. This table shows the number of temporal relations by kind, both in absolute numbers and in percentage.

We also include table 10.6, showing how often each inference rule was employed in stage II of the annotation.

justification for temporal facts	number of occurrence	
inferred	2270	(82.2%)
annotated	111	(4.0%)
solicited	381	(13.8%)
total	2762	(100%)

Table 10.5: Justification for temporal facts

Inference rule	number of times used	
$(x,y) \in S \Rightarrow (y,x) \in S$	54	(2.4%)
$(x,y) \in B \wedge (y,z) \in B \Rightarrow (x,z) \in B$	784	(34.5%)
$(x,y) \in I \wedge (y,z) \in I \Rightarrow (x,z) \in I$	45	(2.0%)
$(x,y) \in B \wedge (y,z) \in I \Rightarrow (x,z) \in B$	167	(7.4%)
$(x,y) \in I \wedge (x,z) \in B \Rightarrow (y,z) \in B$	235	(10.4%)
$(x,y) \in S \wedge (y,z) \in S \Rightarrow (x,z) \in S$	112	(4.9%)
$(x,y) \in B \wedge (y,z) \in S \Rightarrow (x,z) \in B$	313	(13.8%)
$(x,y) \in I \wedge (y,z) \in S \Rightarrow (x,z) \in I$	101	(4.4%)
$(x,y) \in S \wedge (y,z) \in I \Rightarrow (x,z) \in I$	0	(0%)
$(x,y) \in B \wedge (x,z) \in S \Rightarrow (z,y) \in B$	459	(20.2%)
total	2270	(100%)

Table 10.6: Inference rules

10.2 Measuring Temporal Information Content

Now that we have talked about the distribution of the mechanisms used to convey temporal information in text, we will turn to the temporal information itself.

We were interested in the temporal information contained in a text and in the answers to the following questions:

1. What proportion of the temporal information is covered by the explicitly expressed temporal relations in a text?

2. How much temporal information can be inferred from the explicitly expressed temporal relations?

3. How much more do implicitly expressed relations contribute?

4. How much temporal information in addition to the annotated relations has to be solicited from the user before all the temporal information content is covered?

5. How much of the temporal information is anchored in text?

6. What is the minimum number of temporal relations from which we can infer the temporal information of a text?

Before we can progress to answer these questions, we have to define what we mean by "the temporal information contained in a text". We call the sum of the temporal relations holding between all events and all events, between all events and all times, and between all times and all times, the ***temporal information content*** of a text. Using the model theoretic approach introduced in section 7.3, we can represent the temporal information content as follows:

Recall, that the IDs of the event and time expressions annotated in a text form two sets, E and T, respectively, and the denotation of each temporal relation can be viewed as a subset of $(E \cup T) \times (E \cup T)$. The set of temporal relations holding between all events and events, between all events and all times, and all times and all times can be denoted by this Cartesian product: $(E \cup T) \times (E \cup T)$. We can visualise the temporal information content using a $|(E \cup T)| \times |(E \cup T)|$ matrix, where each entry is filled with the temporal relation holding between the entities (we have already used this representation method in chapter 8).

In the remainder of this section, we will use one of the texts from the trial corpus as an example. The text contains 17 events and 3 times; figure 10.1 shows a matrix representing the temporal information content of the text.

The diagonal contains the temporal relation between an entity and itself, and can be left out. Since all temporal relations have an inverse, the upper triangle carries the same

	e1	e2	e3	e4	e5	e6	e7	e8	e9	e10	e11	e12	e13	e14	e15	e16	e17	t1	t2	t3
e1	sim	after	after	after	after	after	before	after	after	after	after	after	after	after	after	after	after	after	unk	after
e2	before	sim	after	after	after	after	before	sim	after	after	after	after	after	after	after	after	sim	after	unk	after
e3	before	before	sim	included	sim	sim	before	after	sim	included	after	after	after	included	after	after	before	Inlcuded	before	after
e4	before	before	includes	sim	includes	before	before	before	includes	sim	after	after	after	sim	after	after	before	included	before	after
e5	before	before	sim	included	sim	sim	before													
e6	before	before	sim	after	sim	sim	before													
e7	after	after	after	after	after	after	sim													
e8	before	sim	before	before				sim												
e9	before	before	sim	included					sim					etc.						
e10	before	before	includes	sim						sim										
e11	before	before	before	before							sim									
e12	before	before	before	before								sim								
e13	before	before	before	before									sim							
e14	before	before	includes	sim										sim						
e15	before	before	before	before											sim					
e16	before	before	before	before					etc.							sim				
e17	before	sim	after	after													sim			
t1	before	before	includes	included														sim		
t2	unk	unk	after	after															sim	
t3	before	before	before	before																sim

Figure 10.1: Temporal information content matrix

information as the lower triangle – with inverse temporal relations. If the entry $(8,3)$ contains *before* (meaning that entity 8 is before entity 3), then the entry $(3,8)$ has to contain *after* (because if entity 8 is before entity 3, then entity 3 must be after entity 8). Therefore, the upper triangle can be left out.

Thus, the temporal information content can be represented and visualised by half the matrix used before and, more importantly, the number of temporal relations in the matrix provides a measure for the temporal information content. The temporal information content for the example (containing 17 events and 3 times) consists of 190 relations. The matrix contains $|(E \cup T)| \times |(E \cup T)| = 20 * 20 = 400$ entries. Taking out the 20 entries along the main diagonal and the 190 entries in the upper triangle, leaves 190 temporal relations.

The questions we have raised in the beginning of this section do not depend on the actual temporal relation holding between two entities, but on whether the temporal relation had been annotated, solicited from the annotator in stage II of the annotation, or inferred – we call this *justification* of the temporal relation. In the following, we will omit the actual temporal information but include the justification.

Using this model theoretic approach and the representation method introduced, we will now turn to the questions raised earlier.

What proportion of the temporal information is covered by the explicitly expressed temporal relations in a text?
This proportion is the number of explicitly expressed temporal relations (these are annotated), the temporal relations holding between the annotated times[1], and the relations that can be inferred from these. The annotation tool records the information necessary to count these relations. In figure 10.2, the temporal relations annotated are shown in blue and the inferred relations in yellow. In the example, the 10 annotated temporal relations are 5.26% of the temporal information content.

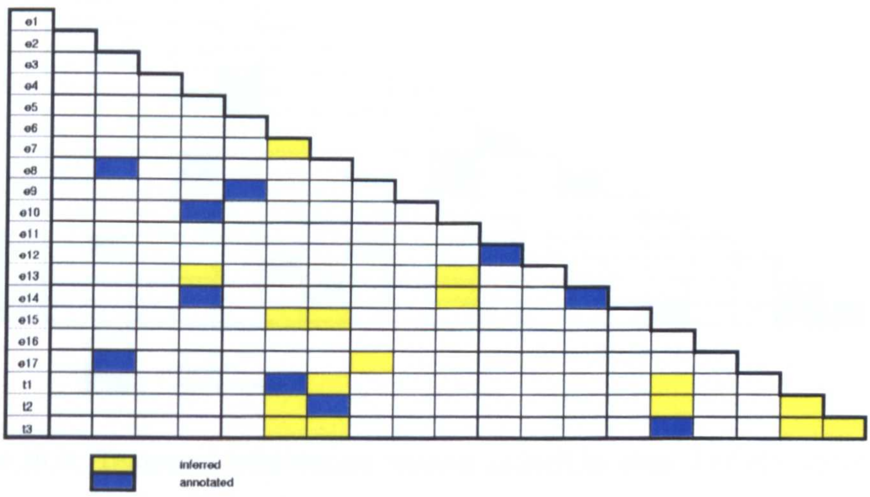

Figure 10.2: Temporal information content covered by explicitly expressed temporal relations

How much temporal information can be inferred from the explicitly expressed temporal relations?
From the 10 annotated explicitly expressed relations, 13 more were inferred and 3 relations

[1]Recall, that times have to be associated with a calender date and the relations holding between the times are automatically inferred by the tool, see chapter 8.

hold between the annotated times, which means that 13.68% of the temporal information content is covered by explicitly expressed temporal relations.

How much more do implicitly expressed annotated relations contribute?
The temporal information content covered increases, when the implicitly expressed temporal relations that are annotated in the text are added, as shown in figure 10.3. In addition to the relations already in the matrix, now shown in grey, 4 implicitly expressed annotated relations (shown in blue) and 9 relations inferred (shown in yellow) are filled into the matrix. Now, 20.53% of the temporal information content is covered.

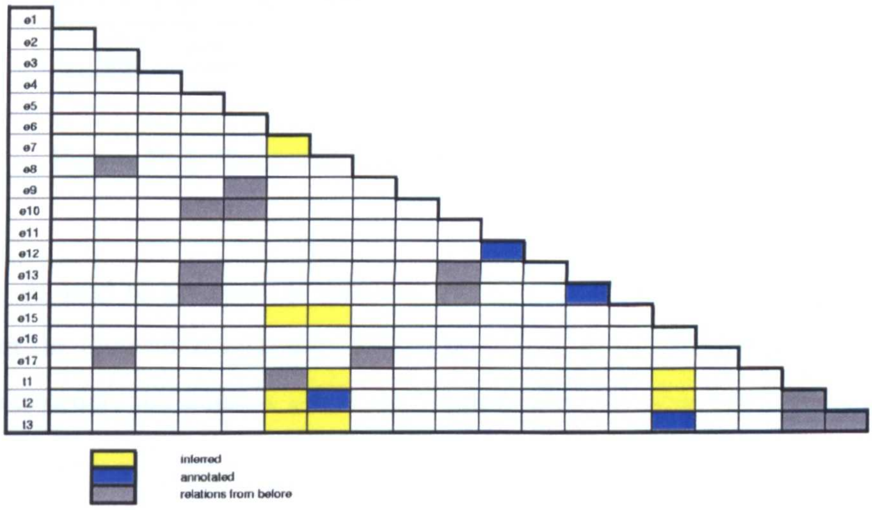

Figure 10.3: Temporal information content gained in stage I of the annotation

How much temporal information in addition to the annotated relations has to be solicited from the user before all the temporal information content is covered?
As we have seen, from 10 explicitly and 4 implicitly expressed temporal relations, 20.53% of the temporal information content is covered. 48 additional temporal relations were solicited from the annotator, before the rest of the information could be inferred – now 100% of the temporal information content is covered. Figure 10.4 shows the matrix for the example with the solicited relations in red, the annotated relations in blue and the inferred relations in

yellow. **If** the number of solicited relations is minimal then we know that that is the number

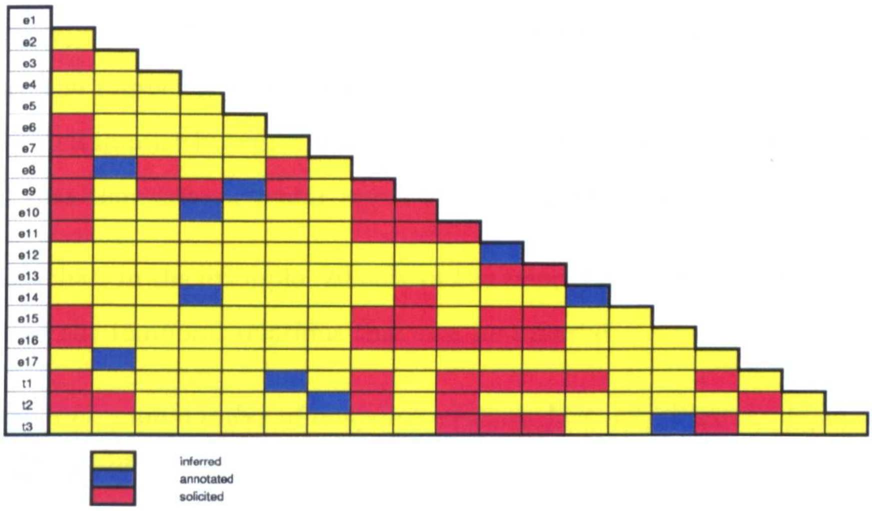

Figure 10.4: Temporal relations solicited from the annotator

of relations we need to annotate in addition, to be able to infer the temporal information content from the annotated relations. But, since we might have solicited these relations in a non-optimal order, the number might not be minimal (see section 9.2.4).

type of temporal relations added	number of relations	percentage covered by relations	percentage of temp. inf. content covered
explicitly expressed annotated	10	5.26%	5.26%
inferred from explicit	16	8.42%	13.68%
implicitly expressed annotated	4	2.11%	15.79%
inferred from implicit	9	4.74%	20.53%
solicited	48	25.26%	45.79%
inferred from solicited	103	54.21%	100.00%

Table 10.7: Temporal information gain

Table 10.7 shows, how the portion of temporal information covered grows with each 'group' of temporal relation added.

How much of the temporal information is anchored in text?
The explicitly expressed temporal relations are part of the temporal information anchored in text, as are the implicitly expressed relations that were annotated in the text. Recall, that the annotated implicitly expressed relations are only part of all the implicitly expressed relations and that they are particular 'types', like event identity.

This means that, in the example, 20.53% of the temporal information is anchored in the text – including the temporal relations that can be inferred.

What is the minimum number of temporal relations from which we can infer the temporal information of a text?
This is question of theoretical more than practical interest, since (as we will explain) a computationally feasible solution is unlikely to exist.

Let us ignore for now any annotated temporal relations and let us assume that we have the full temporal information content of a text available. What we are asking with this question is this. Is there a minimal set of temporal relations, from which – together with the axioms proposed in section 7.3 – we can infer the temporal information content? We call this the ***minimal set problem***.

The minimal number of temporal relations needed to infer the temporal information content is not necessarily the number of annotated relations plus the minimum number of additional relations needed to infer the rest. We cannot be certain whether the temporal relations annotated are optimal in the sense that they maximise the relations that can be inferred.

But if we had such a set, or sets (since it is not clear whether there would be only one minimal set), then we could compare it to the annotated temporal relations, either explicitly or implicitly expressed. We could explore whether humans chose a minimal set of temporal information, how much of the information humans supply is redundant, or to what extend language relies on inference versus redundancy.

Although the minimal set problem seems similar to the problem of finding the minimum number of relations that have to be solicited from the annotator in order to arrive at the rest of the temporal information of the text, there is a important difference, and we cannot easily use the same method to answer both questions.

The difference is, that when the number of solicited relations needs to be minimised, we have the annotated relations as a starting point on which basis we can try and find the best next relation to solicit. But with the minimal set problem, we do not have such a 'base'. The problem can be solved; after all it is a finite search space in which the answer lies. But the search space will be large and the the search computationally very expensive.

10.3 Discussion

The distribution of the mechanisms used to convey temporal information over the corpus has been described in this chapter, as well as the distribution of events, times and signals.

The notion of the temporal information content of a text has also been introduced and we showed how a corpus of this kind can be used to tackle issues like *What proportion of the temporal information is covered by the explicitly expressed temporal relations in a text?*.

Chapter 11

Conclusion and Future Work

Although temporal information plays a crucial role for natural language applications such as information extraction, question answering and topic detection or tracking, only relatively little work has been done to date on the automatic extraction of temporal information from text.

The ability to position reported events in time would be beneficial to the above applications, but before we can develop a system to automatically perform this task, we have to gain insight into the mechanisms used in natural language to do this.

One way to promote this understanding is through the development of an annotation scheme, which allows us to identify the textual expressions conveying events, times and temporal relations in a corpus of 'real' text.

This thesis describes a fine-grained annotation scheme with which events, times and temporal relations in a text can be captured. To aid the application of the scheme to text, we developed a graphical annotation tool, which, alongside allowing the easy mark up of sophisticated temporal information, also contains an interactive inference-based component supporting the gathering of temporal relation information.

In a pilot study to evaluate the scheme, a group of annotators was supplied with a description of the scheme and asked to apply it to a small corpus. The annotation took place in two stages, each having several phases. During stage I, all events and times reported are identified, before explicitly expressed temporal relations are identified. Certain implicitly expressed temporal relations are annotated as well. Stage II deals with establishing those temporal relations that were not identified during the first stage. Most of these are inferred by the tool, others are solicited from the user.

The results of stage I of the pilot study show that there is a general consensus about what events and times are reported in a text with 77% recall and 81% precision, compared to a human produced 'gold standard'. There was lesser consensus about the attributes of these events (60% recall and 64% precision). The study also showed that the scheme was difficult to apply, but is feasible – with improvements to both the scheme and the tool. Clear indications about how both the scheme and the tool can be improved, have also emerged from the study.

The results of stage II of the study, which show how much the annotators agree about the temporal relations holding between all events and times, were much lower. These results are more difficult to interpret because the recall of stage II is practically bounded to the square of the recall of stage I.

The annotation scheme was also used to create an annotated trial corpus, which consists of 6 newspaper texts. The corpus is indicative, despite its size, and first insights into the distribution of the methods used to convey temporal information have been gained. For example, the relevance of implicitly expressed temporal relations and event identity were confirmed, as well as the importance of including non-finite clauses and nominalisations besides finite clauses as event indicators.

The second stage of the annotation scheme, and its development, provided us with the means to invent a method to quantify the temporal information content of a text. Together with the results of the trial corpus, this method lead to interesting issues regarding the temporal information content. We showed, how questions like *How much of the temporal information content is covered by explicitly/implicitly expressed temporal relations?* and *How much more information can be inferred from that?* can be answered using the methods introduced.

The annotation scheme presented exceeds similar approaches in that it allows event-event relations as well as event-time relations. It also allows a wider range of event expressions to be marked up. It includes genre specific extensions and is easily extendible. The temporal relations are geared towards natural language texts to accommodate for the 'fuzziness' with which temporal relations can be expressed.

The scheme is the first annotation scheme to employ an inference-based interactive component to derive temporal relations in addition to the ones annotated.

At the end of chapter 4, we raised the questions whether it would be possible to semantically annotate a text instead of translating it into the kind of abstract representation that formal approaches require, and exploit this semantic labelling when trying to temporally locate events. We have shown a way to achieve this, and we have also shown how to build an inference engine based on this, with which we arrive at the temporal information content of a text.

11.1 Future Work

Before the annotation scheme can be applied to a larger corpus, it is necessary to achieve higher interannotator agreement results. Several areas of improvement to both the scheme and the tool have been pointed out in chapter 7. Examples are: improving the annotation tool, especially optimising the stage of soliciting additional temporal information from the annotator, and easing the annotation by pre-annotating time referring expressions and finite verbs. An important improvement of the scheme, for example, is to make it possible for the annotator to include into the annotation of an event the fact that is temporally related to multiple events or times. At the moment, the annotation of an event only allows the link to one event and one time.

The scoring method for the temporal relations is based on a deductive closure calculated over the annotated temporal relations. As we have already indicated in section 7.3.3, the investigation of a different scoring scheme, for example based on a minimal model, might also lead to better results.

With the annotation scheme introduced and once the improvements are implemented, annotated corpora can be created which would yield benefits like a better understanding of the phenomena of concern, and resources for the training and evaluation of adaptive algorithms to automatically identify features and relations of interest. A larger annotated corpus can be created once the resources are found.

Related to this is the possibility of applying the scheme to texts of other domains or genres. Other domains like financial news might display a very different variety of mechanisms used to convey temporal information. The differences encountered can easily be included as additional extensions into the scheme, once they have been identified.

This leads to another area of future work. As explained in section 6.6, hypothetical, negated and attitude events have been excluded from the approach. Further research into these classes and especially their temporal consequences is needed before they can be integrated into the scheme. The same holds for causality and subeventness, also mentioned in section 6.6. The relevance of event identity, which has been confirmed by the trial corpus results, indicates that subeventness might also be an important mechanisms used in newswire articles. Causality needs to be further investigated before similar claims can be made.

Verb tense and the event classes were not taken into account during the analysis of annotated data, for different reasons. Verb tense locates events in time only very crudely, and the analysis was aimed at other aspects of temporal information. The event classes perception and aspectual did not occur often enough in the corpus to be useful in the analysis. Neither did reporting event which were associated with a time (recall, that reporting events are only considered, if they provide more detailed temporal information about the reported event). Once a larger corpus has been created, it would be interesting to see whether both verb tense or the inclusion of the event classes will increase the number of inferences that can be drawn from the annotated temporal information.

Appendix A

Annotation Guidelines

A.1 Introduction

The overall aim is to identify events in newswire texts and to relate them to a calendrical time on a (fictional or real) time line. We devised an annotation scheme which is as simple as possible and yet tries to identify all information necessary and available to fulfil our task.

There are four basic annotation types: events, times, temporal relations and event identity. We briefly introduce these here.

Events Intuitively, an event is something that happens, something that one can imagine placing on a time line. Events can be viewed as conceptually instantaneous or as happening over a period of time and events can stand in co-reference relationships to each other.

Time Expressions Events happen at or during certain times. Times can be viewed as either points or intervals.

Temporal Relations Events and events, as well as events and times, stand in certain *temporal relations* to each other, for example before and after.

Event Identity In newspaper articles, events are often introduced and come back to later to give more information about them.

Anything else, we do not want to annotate.

For example states (non-essential relations between entities or holding of a non-essential attribute like *The plane can carry four people*) and details of events (like logical subject). The ultimate test of whether something is to be annotated as an event is whether it is anchorable in time and whether one would like to place it on a time map[1].

NOTE: Headlines are not to be annotated at the moment either !

NOTE:

> Events, times and the signals (see below) indicating the temporal relation between them have unique IDs throughout the text. To make the annotation easier, I created an ID counter that just counts from one up. This counter does not distinguish between IDs for events or times or signals, but at least if you stick with it you will assign unique IDs to everything. If an annotation is deleted, the counter does not take this into account either, so there will be gaps (which does not matter for the annotation). If an already annotated text is loaded into the tool, then the highest ID (plus one) is ID that is displayed (so here too might be gaps in the numbers, which is fine). You don't have to use this ID counter Some annotators said they would find it helpful, so I built it in.

A.2 Events

As mentioned before, events are something that 'happens' and we want to 'place in time'. Examples of events are the following:

- *A small single-engine plane crashed into the Atlantic Ocean.*

[1]Note that time map is no longer used as a term, we use the term *time graph* instead.

- *... several vessels and a helicopter were combing the area.*
- *Searchers found the plane's landing gear, seat cushions and other debris.*
- *The 1996 crash of the TWA 747 ...*

Events must be anchorable in time, even though we might not be able to accurately do so with the information in the article. To get a better idea what events are, we can compare them with states like

- *The plane can carry four people.*

or

- *The water is 125 feet deep in the area.*
- *Bill Clinton, who is president of the United States of America,...*

States are either not anchorable in time, like the first example, or are true over time spans longer than that covered by the article (second and third example).

We distinguish different classes of events, the reason being that those classes influence the 'temporal interpretation' of the associated events. This is best illustrated with the following example.

- *The plane, which can carry four people, was seen hitting the water shortly after 11 a.m. by a fisherman, who radioed the Coast Guard, said Petty Officer Jeff Fenn.*

The plane hitting the water is what we call occurrence event, one of the event classes we are interested in placing on a time map. The fisherman seeing the crash is also an *occurrence event*, but is related to the crash event in a particular way. The fisherman *saw* the crash, which means that both events happened at (roughly) the same time. To be able to exploit this special relationship (it helps to locate the crash event in time), we distinguish *perception events* (the fisherman observing) from *occurrence events* (the plane crashing). The third class of events is called *reporting events* (*said Petty Officer Jeff Fenn*), which serve a particular purpose in the genre of newspaper texts, that of giving the source of the information.

The final class of events we distinguish is *aspectual events*, such as

- *Afterwards, the statement said, there were two apparent interruptions of power to the cockpit voice recorder. The first interruption began 1 minute and 39 seconds after the sound and lasted 1 minute and 17 seconds; the second was just before the end of the tape.*

which basically follow the structure mentioned above and involve *aspectual verbs* like *start, stop, finish* etc. Their temporal consequence is that the aspectual event indicates the start or ending of the related event.

A.2.1 Linguistic Level

Events are usually conveyed by clauses containing finite verbs or certain nominalisations. For example:

- *A small single-engine plane crashed into the Atlantic Ocean.*
- *By midafternoon, several vessels and a helicopter were combing the area.*
- *Searchers found the plane's landing gear, seat cushions and other debris.*
- *The 1996 crash of the TWA 747 remains unexplained*

Events can also be represented by non-finite clauses, as in

- *The plane crashed, killing all passengers on board.*

Sometimes several events can be subsumed under one event as in

- *There have been 32 crashes since 1992.*

These events will be treated as one event unless one event is singled out and thus justifies being treated separately. At the moment there is no possibility to indicate this part-whole relationship.

Events can stand in certain co-reference relation to each other and the one we want to annotate is *event identity*. It is impossible to identify the full range of ways in which this can be expressed in text, although co-reference relation between participants, animate or inanimate, are good indicators.

A.2.2 Annotating Events

We treat events as a black boxes without details like logical subject or logical object, unless one of the arguments is an event itself. To make the actual annotation easier we annotate a representative of the event and not the whole clause covering it. The first candidate for event representative is the ***head of the finite verb group***. If the event is conveyed by a nominalisation then we chose the ***head of the nominalisation*** as the representative. If the events is represented by non-finite clauses, we annotate the ***non-finite verb*** as the representative.

The following (simplified) examples show the basics. They are simplified, because they do not show attributes, we will come back to that later.

- *The plane <event>hit</event> the water near New Jersey.*
- *The plane crashed, <event>killing</event> all passengers on board.*
- *The 1996 <event>crash</event> of the TWA 747 remains unexplained.*

Events have a number of optional and obligatory attributes associated with them which are described below, including examples which only show the attributes relevant to the example. An example with all attributes is given further below.

eid: The event ID uniquely identifies the event in the text.

Note:

Please make sure that you are using UNIQUE identifiers for each event mentioned in the text, i.e.. each time you annotate something as an event. Identical events mentioned more than once in the text get a different identifier for each time they are mentioned. And event identity is expressed by using the attributes "relatedToEvent" and "eventRelType".

For example, if we find the following in the text, then both times the crash is mentioned a DIFFERENT ID is given:

The plane crashed, <event eid=2> killing </event> all passengers on board. After the <event eid=3> crash </event> search teams...

potential values: positive integers

optional: no

The plane <event eid=3> crashed </event>.

class: An event belongs to one of these classes.

potential values: OCCURRENCE, PERCEPTION, REPORTING, ASPECTUAL

optional: no

See next attribute for example.

argEvent: Reporting, perception and aspectual events usually have another event or other events as an attribute and the argument ID identifies these.

potential values: positive integers

optional: yes

The MoD <event eid=5 class=reporting argEvent=7> announced </event> that the jet fighter <event eid=7 class=occurrence> ploughed </event> into the mountain.

tense: The tense of the indicating verb plays a crucial role in determining the time the event took place and the simple tense of it is recorded in this attribute. This does not hold for nominalisations and non-finite clauses, since these are tenseless by definition.

The annotator can fill in the tense if it is clear whether the event conveyed happened in the past, is happening or will happen in the future, but this is not necessary. For events conveyed by finite clauses, however, the tense SHOULD BE FILLED IN.

potential values: PAST, PRESENT, FUTURE

optional: yes

> The plane <event eid=3 tense=past> crashed </event>.
> Mr Smith will <event eid=11 tense=future> make </event> an announcement tomorrow afternoon.

aspect: It is sometimes helpful to know whether the aspect of the verb is progressive or perfective, which can be indicated using the attribute aspect.

potential values: PROGRESSIVE, PERFECTIVE

optional: yes

> ... several vessels and a helicopter were <event eid=13 tense=past class=PROGRESSIVE> combing </event> the area.

relatedToEvent: The ID of the event the current event is related to is stored here.

potential values: positive integers

optional: yes

> See next attribute for example.

eventRelType: The type of temporal relation those two events are related by is stored in this attribute.

potential values: BEFORE, AFTER, INCLUDES, IS_INCLUDED, SIMULTANEOUS (see section 6.4 for a description of these relations)

optional: yes

> All 75 people on board the Aeroflot Airbus
> <event eid=4 relatedToEvent=5 eventRelType=simultaneous signalID=7>
> died </event>
> <signal sid=7> when <signal> it <event eid=5> ploughed </event>
> into a Siberian mountain.

relatedToTime: The ID of the time-object the current event is related to is stored here.

potential values: positive integers

optional: yes

See next attribute for example.

timeRelType: The type of temporal relation the event and time-object are related by is stored in this attribute.

potential values: BEFORE, AFTER, INCLUDES, IS_INCLUDED, SIMULTANEOUS

optional: yes

A small single-engine plane <event eid=9 relatedToTime=5 timeRelType=is_included signalID=9> crashed </event>
into the Atlantic Ocean <signal sid=9> on <signal>
<timex tid=5> Wednesday </timex>.

signalID: The ID of the text span that signals the temporal relation holding between two entities can be kept in this attribute.

potential values: positive integers

optional: yes

See relatedToEvent and relatedToTime for example.

Note about event class:

Causal and subevent relationships would come to mind as relationships one might want to annotate. But they are very difficult to define and to distinguish, which is why they are not included as an event class. If the annotator encounters such a relation then this can only be annotated using the attribute *eventRelType* and choosing the appropriate temporal relation (usually cause precedes effect and subevents are temporally included in their 'container' events). Examples for what can be feasibly interpreted as a causality or subevent relation are the following:

- *A sharp movement of the rudder caused the jet's deadly roll.* [**causality**]
- *The plane crashed, killing all passengers on board.* [**subevent**]

A.3 Time Expressions

A.3.1 Linguistic Level

Like events, times can be viewed as having extent (intervals) or as being punctual (points). Rather than trying to reduce one perspective to the other, as has happened in much philosophical discussion on time, we shall simply treat both as *time objects*. It must, however, be possible to associate a calendrical time with the time object. This is not possible with expressions like, for example, recently, and accordingly, this is not a time object. Examples of time expressions are:

- *yesterday*
- *four months (ago)*
- *last Thursday*
- *(in) January 1996*

However, time expressions can be quite complex, as in the next example, where the whole expression denotes a point in time:

- *17 seconds after hearing the sound ...*

The general structure is that a time interval is explicitly related to an event and the calendrical time this refers to is the time 17 seconds after the event.

A.3.2 Annotating Time Expressions

All time expressions have to be anchorable in time to be annotated. This means that is has to be possible to attach a calendrical date to the time expression - even though we might not be able to do so accurately (for example, we could only be able to associate the year 2000 or spring 200 with a time expression). We simply annotate the text span representing the time object, so *last Tuesday* needs to have *last* included because that part is needed to identify the Tuesday in question. Whenever possible though a calendar date has to be attached to the time expression. These can often be worked out by looking at the date of the article and then expressions like 'last Tuesday' can be associated with a calendar date.

Following general convention, and the approach taken in MUC, we distinguish between two classes of time objects, *dates* and *times*, units which are larger or smaller than a day, respectively.

Like events, times have a unique ID so they can be uniquely identified in the text. All time expressions, simple and complex, have the following attributes:

tid: The ID uniquely identifies the time object in the text.
  potential values: positive integers
  optional: no

type: The type of the time object.
  potential values: DATE, TIME, COMPLEX
  optional: no

calDate: The calendrical date represented by the time object.
  potential values: [[DD]MM]YYYY or ('SPR'|'SUM'|'AUT'|'WIN')YYYY
  optional: no

Examples of time expressions are:

```
<timex tid=5 type=date calDate=12031996>last Tuesday</timex>
<timex tid=5 type=time calDate=110025061996>11 a.m.</timex>
```

The more complex time objects mentioned above, have the following additional attributes:

eid: The ID of the event the time interval is related to.

potential values: positive integers

optional: no

signalID: The ID of the signal representing the type of temporal relation.

potential values: positive integers

optional: no

relType: The temporal relation holding between the time interval and the event.

potential values: BEFORE, AFTER, INCLUDES, IS_INCLUDED, SIMULTANEOUS

optional: no

An example of an annotated complex time expression is the following:

<timex tid=5 type=complex eid=3 signalID=7 relType=after>
17 seconds </timex>
<signal sid=7> after <signal> <event eid=3> hearing</event>
the sound ...

And the following example shows a situation involving an indefinite duration (see section below):

John <event eid=3>visited>/event> Sheffield for several weeks.

where the indefinite duration is not annotated at all.

Miscellaneous Non-Time Expressions

- Vague time adverbials: *now, recently* as in ... *three recent crashes* ....

- Indefinite duration-of-time phrases: *for several seconds, a few weeks.* The difference between indefinite durations and the durations in the example above is both the lack of an explicit temporal relation signal and indefiniteness of the duration (*several seconds* as opposed to *17 seconds*).
- Time-relative-to-event phrases: *since the beginning of arms control negotiations.* These will be covered by relating events to events.

Other Special days, such as holidays referred to by name *All Saints' Day* are to be tagged.

A.4 Temporal Relations and Event Identity

A.4.1 Linguistic Level

Events stand in certain temporal relations to other events and to times. Times may be temporally related to other times as well, although this does not happen very often in the articles we have analysed so far. **We do not mark up relations between times and times**, all that needs to be done is to fill in the calendar date for times and the temporal relations between them will be worked out automatically later.

Temporal relations are either expressed explicitly or implicitly. Explicitly expressed are those where, for example, temporal prepositional phrases, temporal adverbial phrases and temporal subordinate conjunctions are used. The first example shows how an event is related to another event (by using the subordinate conjunction *when*) and then how an event is related to a time (by using the temporal preposition *in*).

- *All 75 people on board the Aeroflot Airbus died <u>when</u> it ploughed into a Siberian mountain in March 1994.*

Two more examples to who the use of temporal subordinate conjunctions.

- *The plane crashed <u>after</u> the pilot and his crew ejected.*

- *... before the craft fell, its three rotor blades shot off.*

Temporal prepositional phrases are usually employed when relating events to times, as shown here:

- *A small single engine plane crashed into the Atlantic Ocean on Wednesday.*

Neither relating events to events nor relating events to times need necessarily be explicit, as this example shows:

- *Sunday, an F-14D crashed into the Pacific Ocean off southern California, killing its two crew members.*

Here the preposition *on* is omitted when relating the crash event to *Sunday*. And the killing even it related to the crash event by using a non-finite subordinate clause.

Events can stand in certain co-reference relationships to each other and the relation we want to annotate is *identity*, as in the following example.

- *The National Transportation Safety Board is borrowing a Boeing 737 from Seattle's Museum of Flight as part of its investigation into why a similar jetliner crashed near Pittsburgh in 1994. [...]*
  The museum's aircraft, ironically enough, was donated by USAir, which operated the airplane that crashed, ...

where the first sentence introduces the event and the second sentence refers to the same event - event identity for *crashed*.

The full set of temporal relations we suppose at present is:

is_included	*The plane crashed on Wednesday.* The event is completely included in the time or other event.
includes	*By midafternoon, several vessels were combing the area.* This is the reverse of *is_included*, where we know that the *combing* event includes *midafternoon* completely.
after	*The plane crashed after the pilot and his crew ejected.* This relation, as well as the following *before*, is chosen when an event is clearly after (or before) another event.
before	*... before the craft fell, its three rotor blades shot off.*
simultaneous	*All 75 people on board the Aeroflot Airbus died when it ploughed into a Siberian mountain in March 1994.* Events can happen at exactly the same time (like identical events) or overlap and we know how they overlap exactly. But often we only know that two events happen 'roughly at the same time' without knowing exactly whether and how they overlap. In both cases (when we know exactly how they are overlapping or we only roughly know) we choose the relation *simultaneous*.

A.4.2 Annotating Temporal Relations and Event identity

Events may be related to times or to other events. If two events are related then one of them carries the ID of the other as an attribute to link them and also the type of the relation. Which event has those attributes associated with it depends on the the type of relation (apart from simultaneous which is reflexive). If, for example, event a is *before* event b then event a could carry the ID of event b and the temporal relation before or event b could carry the ID of event a and the temporal relation *after*.

If the word or text span signalling two events being related is realised, as in the example below, then not only is the signal annotated but also the ID of the signal is stored as an attribute of the event, so the link between the signalling word and the events is not lost. If an event is related to a time object then the event carries the ID of the time object and the type of relation in its attributes. Should the signalling word be realised, again as in the example, then the ID of the signal becomes an attribute of the event. The signal has only one attribute, the unique identifier *sid*.

The following examples illustrate the approach.

All 75 people on board the Aeroflot Airbus
<event eid=4 class=OCCURRENCE tense=past relatedToEvent=5
eventRelType=simultaneous signalID=7>
died </event>
<signal sid=7> when </signal> it
<event eid=5 class=OCCURRENCE tense=past> ploughed >/even>
into a Siberian mountain.

A small single-engine plane
<event eid=9 class=OCCURRENCE tense=past relatedToTime=5
timeRelType=is_included signalID=9> crashed </even>
into the Atlantic Ocean about eight miles off New Jersey
<signal sid=9> on </signal>
<timex tid=5> Wednesday </timex>.

If the temporal relation is implicit and the signal (a preposition, for example) is omitted, then the ID of the signal is simply left out as in the following example.

The plane
<event eid=9 class=OCCURRENCE tense=past relatedToTime=5
timeRelType=is_included> crashed </event>
<timex tid=5> Wednesday </timex>.

The plane
<event eid=5 class=OCCURRENCE tense=past relatedToEvent=6
eventRelType=simultaneous> crashed </event>,
<event eid=6 class=OCCURRENCE tense=past> killing </event>
all passengers on board.

The annotation of two identical events is very similar, the only difference being that the relation type is set to *identity*.

A.5 The Process of Annotation

It is recommended to go through 5 phases when annotating.

1. Annotate all events and times, without paying attention to relating them.
2. Relate events to events or times where this is explicitly signalled. This will be done by annotating the signalling text span and adding the appropriate attributes to the event (or one of the events).
3. Relate all time objects that are not yet linked to an event.
4. Go through the events and annotate event identity.
5. After all events, times, most explicit and some implicit temporal relations have been annotated, press the "Calculate Closure" button. The program will prompt for unknown temporal relations (and try to infer as many as possible from the information given so far). Please fill these in as accurate as possible, but do not hesitate to mark relations as *unknown* when unsure.

   Mistakes cannot be corrected in this, because of the computational complexity involved in backtracking all the inferences that were drawn since the last input. Because what this does it to draw all possible inferences from the information given in the annotation and then asking for more information about temporal relation it does not know yet. Every time an answer is put it, it draws more inferences. So even it it seems it asks you every possible relation between every time and every event, it actually does not do that. But it does ask a lot (depending on the text and on the inferences possible).
   All annotations will be "switched off" (i.e.. be made invisible) once the computing of the closure starts and only the entities involved in the question will be highlighted, to make this task a little bit easier.
6. Afterwards, press the "View Results" button and save the results under an appropriate name.

A.6 To be Ignored for the Moment

- Negated sentences (i.e. where the verb is negated).
- Conditionals (e.g. *If China would have responded* ...), counterfactuals and hypothetical expressions.

A.7 Others

Annotate the date of the article at the bottom as *doa*.

A.8 An Example

NAVY BLAMES SHOWING OFF FOR JET CRASH

WASHINGTON The Navy **said**^{e1} **Friday**^{t1} that a pilot was probably **showing off**^{e2} for his parents **when**^{s1} he **crashed**^{e3} an F-14A jet fighter in Nashville **in**^{s2} **January**^{t2}, **killing**^{e4} himself, a fellow officer and three people on the ground.

(e1) Event: eid=1 class=reporting tense=past argEvent=2

(s1) Time: tid=1 type=date calDate=29111996

(e2) Event: eid=2 class=occurrence tense=past aspect=progressive relatedToEvent=3 eventRelType=simultaneous

(s1) Signal: sid=2

(e3) Event: eid=3 class=occurrence tense=past relatedToTime=2 timeRelType=included signalID=2

(s2) Signal: sid=2

(t2) Time: tid=2 type=date caLDate=011996

(e4) Event: eid=4 class=occurrence tense=past relatedToEvent=3 eventRelType=is_included
NOTE: I would have annotated this as a subevent, but since I cannot do this I included the temporal relation *is_included.*

`The pilot, Lt. Comdr. John Stacy Bates, had been` **grounded**^{e5} `for a month` **in**^{s3} **April 1995**^{t3} **after**^{s4} `he` **lost**^{e6} `control of another F-14A` **after**^{s5} **taking off**^{e7}

`from the aircraft carrier USS Lincoln. The plane` **crashed**^{e8} **after**^{s6} `the pilot and his crew` **ejected**^{e9}`. The crew members were` **rescued**^{e10}.

(e5) Event: eid=5 class=occurrence tense=past aspect=perfective relatedToTime=3 timeRelType=is_included signalID=3

(s3) Signal: sid=3

(t3) Time: tid=3 type=date calDate=041995

(s4) Signal: sid=4

(e6) Event: eid=6 class=occurrence tense=past relatedToEvent=5 eventRelTyp=before signalID=4

(s5) Signal: sid=5

(e7) Event: eid=7 class=occurrence tense=past relatedToEvent=6 eventRelType=before

(e8) Event: eid=8 class=occurrence tense=past
NOTE: This is not the same crash event as the one in the first paragraph!

(s6) Signal: sid=6

(e9) Event: eid=9 class=occurrence tense=past relatedToEvent=8 eventRelType=before

(e10) Event: eid=10 class=occurrence tense=past

Rear Adm. Bernard Smith, who **investigated**[e11] the Nashville **crash**[e12], said Bates' judgement was influenced by his parents' presence at the field and his desire to show them risky takeoff and flight manoeuvres.

(e11) Event: eid=11 class=occurrence tense=past

(e12) Event: eid=12 class=occurrence tense=past relatedToEvent=8 eventRelType=identity

The admiral **said**[e13] that **in**[s7]**taking off**[e14] from the Nashville airport on Jan. 29, Bates **ascended**[e15] at an angle steeper than 50 degrees, **violating**[e16] Navy rules.

(e13) Event: eid=13 class=reporting tense=past argEvent=14,15,16

(s7) Signal: sid=7

(e14) Event: eid=14 class=occurrence tense=past aspect=progressive
NOTE: This is not the same take off as event 7!

(e15) Event: eid=15 class=occurrence tense=past relatedToEvent=14 eventRelType=included signalID=7

(e16) Event: eid=16 class=occurrence tense=past aspect=progressive relatedToEvent=15 eventRelType=simultaneous

After[s8] the near-perpendicular **takeoff**[e17], in the clouds, Smith **said**[e18], the pilot **became**[e19] disoriented, and in all likelihood did not realise that he was **heading**[e20] earthward **until**[s9] his jet **pierced**[e21] the clouds at 2,300 feet.

(s8) Signal: sid=8

(e17) Event: eid=17 class=occurrence tense=past relatedToEvent=14 eventRelType=identity

(e18) Event: eid=18 class=reporting tense=past argEvent:17,19,20,21,22 (e19) Event: eid=19 class=occurrence tense=past relatedToEvent=17 eventRelType=after signalID=8

(e20) Event: eid=20 class=occurrence aspect=progressive tense=past relatedToEvent=17 eventRelType=after

(e21) Event: eid=21 class=occurrence tense=past aspect=progressive

(s9) Signal: sid=9

(e22) Event: eid=22 class=occurrence tense=past relatedToEvent=21 eventRelType=after signalID=9

`By then, the admiral said, it was too late to prevent the plane from` **`smashing`** `into the house of Elmer and Ada Newsom.`

(e23) Event: eid=23 class=occurrence tense=past aspect=progressive relatedToEvent=3 eventRelType=is_included
NOTE: *prevent* could be interpreted as an event, but is hypothetical and thus excluded. And because *prevent* is not annotated, the reporting event *said* is not annotated because reporting events are only marked up if their argument events are marked up.

`In addition to Bates, 33, and his radar intercept officer, Lt. Graham Alden Higgins, 28, those` **`killed`**^{e24} `were the Newsoms and Ewing Wair, who was` **`visiting`**^{e25} `them.`

(e24) Event: eid=24 class=occurrence tense=past relatedToEvent=3 eventRelType=is_included
NOTE: I interpret the killing event as a subevent of the crash, so the temporal relation is included

(e25) Event: eid=25class=occurrence tense=past aspect=progressive relatedToEvent=24 eventRelType=simultaneous

NOTE: This relation would be open for discussion. One could say that the visiting event includes the crash event but I would argue that that is not all that clear and thus I put *simultaneous*.

`The crash would never have happened had the pilot made a normal take-off or taken the cloudy weather into account, Smith said after an 11-week review of the incident.`

NOTE: No events are annotated here for the same reason as above (events being hypothetical).

04-12-96

(doa) This is the date of the article. No attributes involved.

Bibliography

J. Allen and G. Ferguson. Actions and Events in Interval Time Logic. Technical Report 251, The University of Rochester, 1994.

J.F. Allen. Maintaining Knowledge About Temporal Intervals. *Communications of the ACM*, 26:832–843, 1983.

J.F. Allen. Towards a General Theory of Action and Time. *Artificial Intelligence*, 23: 123–154, 1984.

A. Bagga and B. Baldwin. Entity-Based Cross-Document Coreferencing Using the Vector Space Model. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and the 17th International Conference on Computational Linguistics (COLING-ACL '98)*, Montreal, Quebec, 1998.

A. Bell. *The Language of News Media.* Blackwell, 1991.

A. Bell. *The Discourse Structure of News Structure*, pages 64–104. Blackwell, 1998.

T. Brants and O. Plaehn. Interactive Corpus Annotation. In *Second International Conference on Language, Resources and Evaluation, (LREC-2000)*, pages 1435–1439, Athens, Greece, 2000.

D.S. Bree and A. Feddag. The Annotation of Temporal Information in Natural Language Sentences. In P. Bosch and R. van der Sandt, editors, *Focus and Natural Language Processing: proceedings of a conference in celebration of the 10th anniversary of the journal of Semantcis*, pages 257–270. Working papers of the IBM institute for Logic and Linguistics, 1996.

J. Carletta. Assessing Agreement on Classification Tasks: the Kappa Statistic. *Computational Linguistics*, 22(2):249–254, 1996.

J. Cowie and W. Lehnert. Information Extraction. *Communications of the ACM*, 39(1): 80–91, 1996.

J.D.M. Crowe. *Constraint Based Event Recognition for Information Extraction*. PhD thesis, University of Edinburgh, 1997.

L. Ferro, I. Mani, B. Sundheim, and G. Wilson. TIDES Temporal Annotation Guidelines. Technical Report MTR 00W0000094, The MITRE Corporation, October 2000. Draft-Version 1.0.

E. Filatova and E. Hovy. Assigning Time-Stamps to Event-Clauses. In *Proceedings of ACL-EACL 2001, Workshop for Temporal and Spatial Information Processing*, pages 88–95, Toulouse, 2001.

J.G. Fiscus, G. Doddington, J.S. Garofolo, and A. Martin. NIST's 1998 Topic Detection and Tracking Evaluation (TDT2). In *Proceedings of the DARPA Broadcast News Workshop*, pages 19–24, Herndon, Virginia, 2001.

C. Freska. Temporal Reasoning Based on Semi-Intervals. *Artificial Intelligence*, 54:199–227, 1992.

M. Gagnon and G. Lapalme. From Conceptual Time to Linguistic Time. *Computational Linguistics*, 22(1):91–127, 1996.

R. Gaizauskas and Y. Wilks. Information Extraction: Beyond Document Retrieval. *Journal of Documentation*, 54(1):70–105, 1998.

A. Galton. A Critical Examination of Allen's Theory of Action and Time. *Artificial Intelligence*, 42:159–188, 1990.

A. Galton. Time and Change for AI. In D.M. Gabbay, C.J. Hogger, and J.A. Robinson, editors, *Handbook of Logic in Artificial Intelligence and Logic Programming*, pages 175–240. Clarendon Press, Oxford, 1995.

R. Garside, G. Leech, and A. McEnery, editors. *Corpus Annotation.* Longman, 1997.

C.F. Goldfarb. *The SGML Handbook.* Clarendon Press, Oxford, 1990.

C.F. Goldfarb. *The XML Handbook.* Prentice Hall PTR, Upper Saddle River, N.J., 1998.

M.A.K. Halliday. *An Introduction to Functional Grammar.* Edward Arnold, 1994. Second Edition.

C.L. Hamblin. Starting and Stoppping. *The Monist*, 53:410–425, 1969.

L. Hirschman, P. Robinson, J. Burger, and M. Vilain. Automating Coreference: The Role of Annotated Training Data. In *AAAI 1998 Spring Symposium on Applying Machine Learning to Discourse Processing*, pages 1419–1422, Stanford, 1998.

N. Ide and J. Verónis, editors. *Text Encoding Initiative.* Kluwer, Dordrecht, 1995.

H. Kamp and U. Reyle. *From Discourse to Logic Introduction to Modeltheoretic Semantics of Natural Language (Discourse Representation Theory).* Kluwer, Dordrecht, 1993.

G. Katz and F. Arosio. The Annotation of Temporal Information in Natural Language Sentences. In *Proceedings of ACL-EACL 2001, Workshop for Temporal and Spatial Information Processing*, pages 104–111, Toulouse, 2001. Association for Computational Linguistics.

H. Kucera and W.N. Francis. *Computational Analysis of Present-Day American English.* Brown University Press, 1967.

A. Lascarides and N. Asher. Semantics and Pragmatics for the Pluperfect. In *Proceedings of the Sixth European Chapter of the Association of Computational Linguistics*, pages 250–259, Utrecht, 1993.

G.N. Leech and A. Wilson. EAGLES Morphosyntactic Annotation. Technical Report EAGCSG/ITR-T3, Istituto di Linguistica Computazionale, 1994.

D. McDermott. A Temporal Logic for Reasoning about Processes and Plans. *Cognitive Science*, 6:101–155, 1982.

G. Miller and P. Johnson-Laird. *Language and Perception.* Cambridge University Press, Cambridge, MA, 1976.

M. Moens and M. Steedman. Temporal Ontology and Temporal Reference. *Computational Linguistics*, 14:15–28, 1988.

B. Moulin. Temporal Contexts for Discourse Representation: An Extension of the Conceptual Graph Approach. *Artificial Intelligence*, 7:227–255, 1997.

MUC6. *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, 1995. Morgan Kaufman.

MUC7. *Proceedings of the Seventh Message Understanding Conference (MUC-7)*, 1998. Morgan Kaufman.
Available at http://www.itl.nist.gov/iaui/894.02/related_projects/muc/index.html.

R. Quirk and S. Greenbaum, editors. *A University Grammar of English.* Longman, 1973.

H Reichenbach. *Elements of Symbolic Logic.* University of California Press, 1947.

U. Reyle and A. Rossdeutscher. Understanding Very Short Stories. cited as in (Katz 2001), 2000.

G. Ritchie. Temporal Clauses in English. *Theoretical Linguistics*, 6:87–115, 1979.

M Rooth, S. Riezler, D. Prescher, S. Schulte im Walde, G. Carroll, and F. Beil. Inducing Lexicons with the EM Algorithm. AIMS Report 4(3), Institut für Maschinelle Sprachverarbeitung, Universität Stuttgart, 1998.

B. Schiffman, I. Mani, and K. Concepcion. Producing Biographical Summaries: Combining Linguistic Knowledge with Corpus Statistics. In *Proceedings of the 39th Meeting of the Association of Computational Linguistics (ACL-EACL 2001)*, pages 450–457, Toulouse, 2001.

F. Schilder and C. Habel. From Temporal Expressions to Temporal Information: Semantic Tagging of News Messages. In *Proceedings of ACL-EACL 2001, Workshop for Temporal and Spatial Information Processing*, pages 65–72, Toulouse, 2001.

A. Setzer and R. Gaizauskas. Annotating Events and Temporal Information in Newswire Texts. In *Proceedings of the Second International Conference on Language Resources and Evaluation (LREC2000)*, pages 1287–1293, Athens, Greece, 2000a.

A. Setzer and R. Gaizauskas. Building a Temporally Annotated Corpus for Information Extraction. In *Proceedings of the Workshop Information Extraction Meets Corpus Linguistics held in conjunction with the Second International Conference on Language Ressources and Evaluation (LREC2000)*, pages 9–14, Athen, Greece, 2000b.

A. Setzer and R. Gaizauskas. A Pilot Study On Annotating Temporal Relations In Text. In *Proceedings of ACL-EACL 2001, Workshop for Temporal and Spatial Information Processing*, pages 73–80, Toulouse, France, 2001.

S. Siegel and N.J. Castellan. *Nonparametric Statistics for Behavioral Sciences, 2nd Edition.* McGraw-Hill, London, New Yord, 1988.

J.F. Sowa. *Conceptual Structures.* Addison Wesley, Reading, Mass., 1984.

M. Steedman. Temporality. In J. van Bentham and A. ter Meulen, editors, *Handbook of Logic and Language*, pages 895–938. Elsevier, 1997.

A. ter Meulen. *Representing Time in Natural Language.* MIT Press. Cambridge, Massachussets, 1995.

D. Thompson, editor. *The Oxford Compact English Dictionary.* Oxford University Press, 1996.

J. van Eijck and H. Kamp. Representing Discourse in Context. In J. van Bentham and A. ter Meulen, editors, *Handbook of Logic and Language*, pages 179–237. Elsevier, 1997.

Z. Vendler. *Linguistics in Philosophy.* Cornell University Press, Ithaca, New York, 1967.

M.B. Vilain. A System for Reasoning about Time. In *Proceedings of AAAI-82*, pages 197–201, 1982.

E. Voorhees. Overview of the TREC-9 Question Answering Track. In *Proceedings of the Ninth Text Retrieval Conference (TREC-9)*, Gaithersburg, Maryland, 2001. Available at http://trec.nist.gov/pubs/trec9/t9_proceedings.html.

E. Voorhees and D. Harman. Overview of the Ninth Text REtrieval Conference (TREC-9). In *Proceedings of the Ninth Text Retrieval Conference (TREC-9)*, Gaithersburg, Maryland, 2001. Available at http://trec.nist.gov/pubs/trec9/t9_proceedings.html.

J.M. Wiebe, T.P. O'Hara, T. Ohrstrom-Sandgren, and K.J. McKeever. An Empirircal Approach to Temporal Reference Resolution. *Artificial Intelligence Research*, 9:247–293, 1998.

G. Wilson and I. Mani. Robust Temporal Processing of News. In *Proceedings of the 38th Meeting of the Association of Computational Linguistics (ACL 2000)*, pages 69–76, Hong Kong, October 2000.

G. Wilson, I. Mani, B. Sundheim, and L. Ferro. A Multilingual Approach to Annotating and Extracting Temporal Information. In *Proceedings of ACL-EACL 2001, Workshop for Temporal and Spatial Information Processing*, pages 81–87, Toulouse, 2001.