Empirical Validation of Reichenbach's Tense Framework

Leon Derczynski
Department of Computer Science
University of Sheffield, UK
leon@dcs.shef.ac.uk

Robert Gaizauskas
Department of Computer Science
University of Sheffield, UK
robertg@dcs.shef.ac.uk

Abstract

There exist formal accounts of tense and aspect, such as that detailed by Reichenbach (1947). Temporal semantics for corpus annotation are also available, such as TimeML. This paper describes a technique for linking the two, in order to perform a corpus-based empirical validation of Reichenbach's tense framework. It is found, via use of Freksa's semi-interval temporal algebra, that tense appropriately constrains the types of temporal relations that can hold between pairs of events described by verbs. Further, Reichenbach's framework of tense and aspect is supported by corpus evidence, leading to the first validation of the framework. Results suggest that the linking technique proposed here can be used to make advances in the difficult area of automatic temporal relation typing and other current problems regarding reasoning about time in language.

1 Introduction

In his 1947 account, Reichenbach offers a three-point framework for describing the tenses of verbs. The framework uses the concepts of *speech*, *event* and *reference* points and the relations between them in order to give descriptions of tenses. This framework has since been widely adopted and scrutinised by those working in the fields of linguistics and type-theoretic semantics.

Within computational linguistics, increased interest in temporal semantics, automatic annotation of temporal information, and temporal information extraction has led to temporally annotated resources being created and the discovery of many interesting problems. One of the most difficult problems in temporal information extraction is that of automatically determining the nature of the temporal order of times and events in a given discourse.

Temporal ordering is an important part of language – it allows us to describe history, to communicate plans and to discuss change. When automatic temporal annotation is broken into a tripartite task of detecting events, detecting times, and automatically determining the ordering of events and times, the third part – determining temporal ordering – is the most difficult. This is illustrated by, for example, the low performance scores at the most recent TempEval exercise (Verhagen et al., 2010), which focuses on automatic annotation of temporal relations. Event-event ordering is the hardest temporal relation typing task, and the focus of this paper.

Reichenbach's framework not only offers a means of formally describing the tenses of verbs, but also rules for temporally arranging the events related by these verbs, using the its three abstract points. This can, for a subset of cases, form a basis for describing the temporal ordering of these events.

The framework is currently used in approaches to many computational linguistics problems. These include language generation, summarisation, and the interpretation of temporal expressions. When automatically creating text, it is necessary to make decisions on when to shift tense to properly describe events. Elson and McKeown (2010) relate events based on a "perspective" which is calculated from the reference and event times of two verbs that each describe events. They construct a natural language generation system that uses reference times in order to correctly write stories. Further, reference point management is critical to medical summary generation. In order to helpfully unravel the meanings of tense shifts in minute-by-minute patient reports, Portet et al. (2009) required understanding of the reference point. The framework also helps interpret linguistic expressions of time (timexes). Reference time is required to interpret anaphoric expressions such as *"last April"*. Creation of recent timex corpora prompted the comment that there is a "need to develop sophisticated methods for temporal focus tracking if we are to extend current time-stamping technologies" (Mazur and Dale, 2010) – focus as a rôle filled by Reichenbach's reference point. In fact, demand for accurate reference time management is so persistent that state of the art systems for converting times expressed in natural language to machine-readable format now contain extra layers solely for handling reference time (Llorens et al., 2012).

Given the difficulty of automatically determining the orderings, or temporal relations, between events, and the suggested ability of Reichenbach's framework to provide information for this, it is natural to apply this framework to the temporal ordering task. Although tense has played a moderately useful part in machine learning approaches to the task (Hepple et al., 2007), its exact role in automatic temporal annotation is not fully understood. Further, though it was not the case when the framework was originally proposed, there now exist resources annotated with some temporal semantics, using TimeML (Pustejovsky et al., 2005). Comparing the explicit temporal annotations within these resources with the modes of interaction proposed by Reichenbach's framework permits an evaluation of the validity of this established account of tense and aspect.

This paper addresses the following questions:

1. How can Reichenbach's framework be related to a modern temporal annotation schema?
2. Between which event-relating verbs should the framework be applied?
3. Given Reichenbachian descriptions of pairs of verbs in English, how can one automatically determine the temporal relation between the events described by the verbs?
4. Do the behaviours that Reichenbach proposes agree with human-annotated, ground-truth data?

The main contributions made by this paper are twofold. Firstly, it provides an account of how tensed verb events, described according to Reichenbach, can be linked with each other to extract information about their temporal ordering. Secondly, it provides the first corpus-based validation of Reichenbach's framework against human-annotated ground truth data.

The rest of this paper is constructed as follows. Firstly Reichenbach's framework is introduced with accompanying examples (Section 2). Relevant parts of the TimeML annotation scheme are covered in Section 4. Discussion of how event-signifying verbs may be associated and then ordered is in Section 3. Section 5 introduces a way of connecting TimeML with Reichenbach's three time points. A corpus-based evaluation of Reichenbach's framework is in Section 6, and conclusion in Section 7.

2 Reichenbach's Framework

The core of the framework comprises three time points – speech time, event time and reference time. These are ordered relative each other using equality (e.g. simultaneity), precedence or succession operators. The tense and aspect of each verb is described using these points and the relations between them.[1] Interactions between verbs can be described in terms of relations between the time points of each verb.

2.1 Time Points

Reichenbach introduces three abstract time points to describe tenses. Firstly, there is speech time, S.[2] This represents the point at which the tensed verb described is uttered or written. Secondly, event time E is the time that the event introduced by the verb occurs. The position of this point relative to other verbs' Es reveals the temporal order of events related by a discourse. Thirdly, there is reference time R; this is an abstract point, from which events are viewed. Klein (1994) describes R as "the time to which a claim is constrained." In Example 1, speech time S is the point when the author created the discourse.

(1) *By then, she had left the building.*

[1] Although Reichenbach's suggests the framework is for describing tense, it also provides an account of perfective aspect. For example, Reichenbach's anterior tenses correspond to perfective aspect in English.

[2] For this paper, it is assumed that speech time is equivalent to DCT, unless otherwise explicitly positioned by discourse. Following the description of discourse deixis by Fillmore (1971), this is the same as always setting speech time S equal to his encoding time *ET* and not decoding time *DT*.

Relation	*Reichenbach's Tense Name*	*English Tense Name*	*Example*
E<R<S	Anterior past	Past perfect	*I had slept*
E=R<S	Simple past	Simple past	*I slept*
R<E<S R<S=E R<S<E	Posterior past		*I expected that I would sleep*
E<S=R	Anterior present	Present perfect	*I have slept*
S=R=E	Simple present	Simple present	*I sleep*
S=R<E	Posterior present	Simple future	*I will sleep (Je vais dormir)*
S<E<R S=E<R E<S<R	Anterior future	Future perfect	*I will have slept*
S<R=E	Simple future	Simple future	*I will sleep (Je dormirai)*
S<R<E	Posterior future		*I shall be going to sleep*

Table 1: Reichenbach's tenses; from Mani et al. (2005)

In this sentence, one perceives the events from a point S after they occurred. Reference time R is *"then"* – abstract, before speech time, and after event time E, the leaving of the building.

2.2 Tense Structure

Using these points, Reichenbach details the structure of nine tenses (see Table 1). The tenses detailed by Reichenbach are past, present or future, and may take a simple, anterior or posterior form. In English, the tenses apply to single non-infinitive verbs and to verbal groups consisting of a head verb and auxiliaries. Reichenbach's tense system describes the arrangement of the time points for each tensed verb.

In Reichenbach's view, different tenses specify different relations between S, E and R. Table 1 shows the six tenses conventionally distinguished in English. As there are more than six possible ordering arrangements of S, E and R, some English tenses might suggest more than one arrangement. Reichenbach's named tenses names also suffer from this ambiguity when converted to $S/E/R$ structures, albeit to a lesser degree. Past, present and future tenses imply $R < S$, $R = S$ and $S < R$ respectively. Anterior, simple and posterior tenses imply $E < R$, $E = R$ and $R < E$ respectively.

3 Associating Event Verbs

This validation relies on assessing temporal orderings suggested by Reichenbach's framework. These temporal orderings are between event-describing verbs. Therefore, we must determine which verbs may be directly temporally associated with one another. The simplest case is to examine relations between the smallest set of events which contains at least one relation: an event pair. So, in order to proceed, the following must be defined:

1. How does connecting a pair of verbs affect the relative positions of one verb's $S/E/R$ to anothers;
2. Which pairs of events can be linked;
3. How the results of linking events can be propagated from Reichenbach's framework to TimeML.

3.1 Reichenbachian Event-Event Relations

When sentences are combined to form a compound sentence, verbs interact, and implicit grammatical rules may require tenses be adjusted. These rules operate in such a way that the reference point is the same in all cases in the sequence. Reichenbach names this principle **permanence of the reference point**:

> We can interpret these rules as the principle that, although the events referred to in the clauses may occupy different time points, the reference point should be the same for all clauses.

Example 2 show a sentence in which this principle applies.

(2) *John told me the news, but I had already sent the letter.*

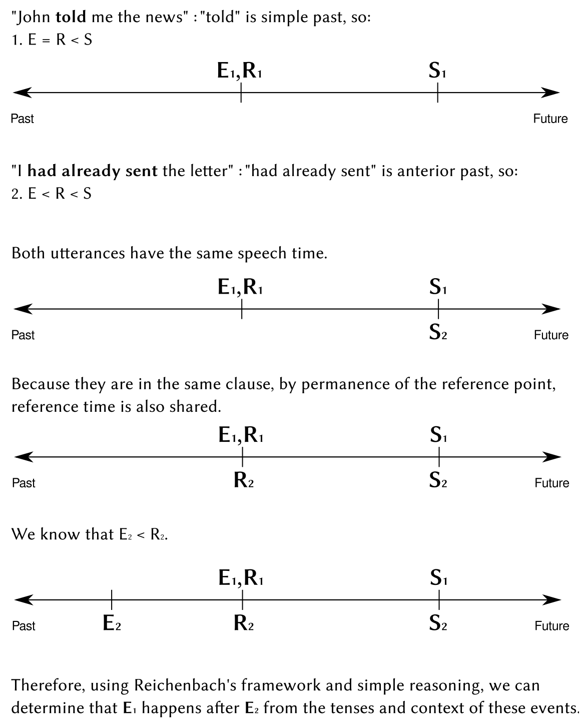


Figure 1: An example of permanence of the reference point.

Example 2 shows a sentence with two verb events – *told* and *had sent*. Using Reichenbach's framework, these share their speech time S (the time of the sentence's creation) and reference time R, but have different event times (see Figure 1). In the first verb, reference and event time have the same position. In the second, viewed from when John told the news, the letter sending had already happened – that is, event time is before reference time. As reference time R is the same throughout the sentence, we know that the letter was sent before John mentioned the news. Arranging S, E and R for each verb in a discourse and linking these points with each other ensures correct temporal ordering of events that the verbs describe.

3.2 Temporal Context

In the linear order that events and times are introduced in discourse, speech and reference points persist until changed by a new event or time. Observations during the course of this work suggest that the reference time from one sentence will roll over to the next sentence, until it is repositioned explicitly by a tensed verb or time. To make discussion of sets of verbs with common reference times easy, following Derczynski and Gaizauskas (2011a), we call each of these pragmatic groups a **temporal context**.

Temporal contexts may be observed frequently in natural language discourse. For example, the main body of a typical news article shares the same reference point, reporting other events and speech as excursions from this context. Each conditional world of events invoked by an "if" statement will share the same context. Events or times linked with a temporal signal will share a reference point, and thus be explicitly placed into the same temporal context. Reichenbach constrains the verbs which may be linked under his framework by using a grammatical device – the sequence of tenses. This is the only description in his paper of which in contexts the framework applies.

Several previous studies have indicated temporal context-like bounds in discourse. Dowty (1986) describes something similar to temporal context with the idea of the **temporal discourse interpretation principle** (TDIP). This states:

> Given a sequence of sentences S_1, S_2, ..., S_n to be interpreted as a narrative discourse, the reference time of each sentence S_i (for i such that $1 < i - n$) is interpreted to be:
>
> (a) a time consistent with the definite time adverbials in S_i, if there are any;
>
> (b) otherwise, a time which immediately follows the reference time of the previous sentence S_{i-1}.

The TDIP accounts for a set of sentences which share a reference and speech point. However, as with other definitions of temporal context, this principle involves components that are difficult to automatically determine (e.g. "consistent with definite time adverbials"). Webber (1987) introduces a listener model, incorporating R as a means of determining temporal focus. Her focus resumption and embedded discourse heuristics capture the nesting behaviour of temporal contexts. Further, Eijck and Kamp (2010) describe context-bounding, tense-based rules for applicability of Reichenbach's framework. These comprise a qualitative model of temporal context.

As described in Chapter 4 of Hornstein (1990), permanence of the reference point does not apply between main verb events and those in embedded phrases, relative clauses or quoted speech. These latter events occur within a separate temporal context, and it is likely that they will have their own reference time (and possibly even speech time, for example, in the case of quoted speech).

To handle such subordinate clauses, one must add a caveat – S and R persist as a discourse is read in textual order, for each temporal context. A context is an environment in which events occur, and may be the main body of the document, a tract of reported speech, or the conditional world of an *if* clause (Hornstein, 1990). For example:

(3) *Emmanuel had said "This will explode!," but changed his mind.*

Here, *said* and *changed* share speech and reference points. Emmanuel's statement occurs in a separate context, which the opening quote instantiates and is ended by the closing quote, and begins with an S that occurs at the same time as *said* – or, to be precise, *said*'s event time E_{said}.

However, temporal context information is not overt in TimeML annotations (Section 4) and not readily available from discourse. We therefore have the problem of needing to model temporal context, in order to decide to which event verb-event verb pairs the framework should be applied.

In order to temporally relate verb events using Reichenbach's framework, we must filter verb event pairs so that only those in which both events are in the same temporal context are considered. This requires a model of temporal context. If events in a pair are not both in the same context, Reichenbach's framework may not directly apply, and the pair should not be further analysed.

Simple techniques for achieving temporal context modelling could work based on sentence proximity. Proximity alone may not be sufficient, given this paper's earlier observations about quoted speech, re-positioning of the reference point and so on. Further techniques for temporal context modelling are detailed in experiments below in Section 6.

3.3 Progressive Aspect

While Reichenbach's basic framework provides an explicit, point-based account of the perfective, it does not do the same for the progressive. This section proposes a point-based solution for the progressive, within Reichenbach's framework.

Consider that event time E is a temporal interval, and therefore may be split into start and finish points E_s and E_f between which the event obtains. Given this, it becomes possible to place reference or speech time *within* the event interval – later than E_s but before E_f. This enable construction of scenarios where one is reporting on an ongoing process that starts before and ends after the reporting point – the same concept related by use of progressive aspect – and corresponds to Reichenbach's illustration of "extended tenses."

Examples of the Reichenbachian structure of progressive-aspect events are included in Table 3. For the simple tenses (where $R = E$) which TimeML describes aspect of NONE, it is assumed not that the event is a point, but that the event is an interval (just as in the progressive) and the reference time is *also* an interval, starting and finishing at the same times as the event (e.g. $R_s = E_s$ and $R_f = E_f$).

4 TimeML Schema and Dataset

TimeML (Pustejovsky et al., 2005)[3] is an annotation markup language for temporal semantics. It defines annotations for events and temporal expressions (both also called "intervals," because they are modelled

[3] or, in its current incarnation, ISO-TimeML

Relation	Explanation of A-relation-B
BEFORE	A finishes before B starts
AFTER	A starts after B ends
INCLUDES	A start before and finishes after B
IS_INCLUDED	A happens between B's start and finish
DURING	A occurs within duration B
DURING_INV	A is a duration in which B occurs
SIMULTANEOUS	A and B happen at the same time
IAFTER	A happens immediately after B
IBEFORE	A happens immediately before B
IDENTITY	A and B are the same event/time
BEGINS	A starts at the same time as B, but finishes first
ENDS	A starts after B, but they finish at the same time
BEGUN_BY	A starts at the same time as B, but goes on for longer
ENDED_BY	A starts before B, but they finish at the same time

Table 2: TimeML temporal interval relations

as periods of time between a start and end point). TimeML also defines annotations for the temporal relations that exist between intervals, in the form of binary interval relations.

4.1 Tense System

Under TimeML, event annotations have a part-of-speech feature. This permits easy identification of verbs, which are the relevant events for this study. Each verb has both tense and aspect features, which take values from three "tenses[4]" (PAST, PRESENT and FUTURE) and four "aspects" (NONE, PERFECTIVE, PROGRESSIVE and PERFECTIVE_PROGRESSIVE) respectively.

In many ways, TimeML's tense system is less expressive than that of Reichenbach's. It provides a maximum of 12 tense/aspect combinations, whereas the framework provides 19. The TimeML system cannot express anterior tenses according to Reichenbach's scheme. Further, TimeML does not account for the reference point, making shifts of reference time difficult to express other than by describing their end results. In its favour, TimeML does explicitly cater for progressive aspect – something that Reichenbach does not, a solution for which is proposed later in Section 3.3.

4.2 TimeML Temporal Relations

In TimeML, temporal relations may be annotated using one of thirteen interval relations. This set of relations is based on Allen's temporal interval algebra (Allen, 1983).

Temporal relations obtain between two intervals. They describe the natural of temporal ordering between those intervals. Those intervals may be either times or events, and need not be of the same type. Accordingly, a temporal relation annotation must specify two intervals and a relation that obtains from the first to the second; see Example 4. Additional information may be included, such as references to phrases that help characterise the relation (Derczynski and Gaizauskas, 2011b). Descriptions of the TimeML interval relations, based on Allen (1983)'s interval relation set, are given in Table 2.

(4)
```
John <EVENT eiid="e1" tense="PAST" aspect="NONE">told</EVENT>
me the news, but I had already
<EVENT eiid="e2" tense="PAST" aspect="PERFECTVE">told</EVENT>
the letter.
<TLINK eventInstanceID="e1" relType="BEFORE" relatedToInstance="e2" />
```

4.3 TimeBank

TimeBank v1.2 is a TimeML annotated corpus. It contains 6 418 temporal link annotations, 1 414 time annotations and 7 935 event annotations. TimeBank's creation involved a large human annotator effort and multiple versions (Pustejovsky et al., 2003); it is currently the largest temporally-annotated corpus containing explicit temporal relations.

[4]In TimeML v1.2, the tense attribute of events has values that are conflated with verb form. This conflation is deprecated in the most versions of TimeML, though no significant volume of ground-truth data is annotated under these later schemas.

TimeML Tense	TimeML Aspect	Reichenbach structure
PAST	NONE	$E = R < S$
PAST	PROGRESSIVE	$E_s < R < S, R < E_f$
PAST	PERFECTIVE	$E_f < R < S$
PRESENT	NONE	$E = R = S$
PRESENT	PROGRESSIVE	$E_s < R = S < E_f$
PRESENT	PERFECTIVE	$E_f < R = S$
FUTURE	NONE	$S < R = E$
FUTURE	PROGRESSIVE	$S < R < E_f, E_s < R$
FUTURE	PERFECTIVE	$S < E_s < E_f < R$

Table 3: TimeML tense/aspect combinations, in terms of the Reichenbach framework.

Inter-annotator agreement (IAA) describes the quality of annotation in TimeBank. Events were annotated with agreement 0.78; given events, their tenses were annotated with agreement 0.96 and aspect with agreement of 1.00 (complete agreement). For temporal relations between intervals, the type of relation reached agreement of 0.77. TimeBank is the ground truth used to validate Reichenbach's framework.

5 Mapping from TimeML to Reichenbach

Given the above accounts of the two schemas for describing events, tense and aspect, we shall now consider how they may be joined. TimeML and Reichenbach's framework do not use the same temporal semantics, so some work is required to map descriptions from one format to the other.

5.1 Interval Disjunctions

Based on our above accounts of Reichenbach's framework, TimeML, progressive aspect, temporal context, and temporal ordering, it is now possible to derive a mapping from TimeML to Reichenbach based on three-point algebra. Accordingly, the TimeML tenses and aspects may be mapped to $S/E/R$ structures using the translations shown in Table 3.

Working on the hypothesis that Reichenbach's framework may constrain a TimeML relation type to more than just four possible groupings, the table of tense-tense interactions is rebuilt, giving for each event pair a disjunction of TimeML relations instead of one of four labels. In TimeML, aspect values are composed of two "flags", PERFECTIVE and PROGRESSIVE, which may both be independently asserted on any verb event annotation. For simplicity, PERFECTIVE_PROGRESSIVE aspect was converted to PERFECTIVE; this feature value accounts for 20 of 5974 verb events, or 0.34% – a minority that does not greatly impact overall results. Another simplification is that participle "tenses" in TimeML (PASTPART and PRESPART) are interpreted the same way as their non-participle equivalents.

When determining corresponding TimeML TLINK `relType` values given two Reichenbachian tense structures, there is often more than one possible `relType`. In fact, multiple potential TimeML interval relation types apply in many cases. Given the events and tenses in Example 4, the relation could be not only BEFORE but also IBEFORE. Instead of specifying the exact relation, this *constrains* the type of temporal ordering.

The disjunctions of interval relations indicated by various tense/aspect pair combinations frequently recur, and are not unique to each tense/aspect pair combination. In fact, this approach to event-event ordering causes the framework to generate a limited set of such disjunctions, which matches the interval relation disjunctions corresponding to semi-intervals.

5.2 Emergence of Semi-intervals

Where an interval is defined by its start and end bounds, and both of these are required in order to perform interval reasoning, a semi-interval is defined using only one of its bounds. The sets of interval relation disjunctions indicated by the above tense/aspect combinations overlaps with the relation types present in a semi-interval temporal algebra. This algebra, identified by Freksa (1992), differs from the conventional interval reasoning described above by only make one bound of each interval finite. A full list of Freksa's semi-interval relations is provided in Table 4.

Freksa semi-interval relations can be described in terms of groups of Allen relations. The disjunctions

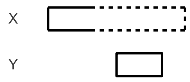
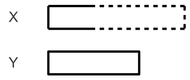
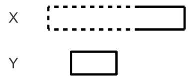
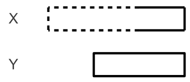
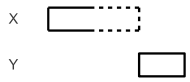
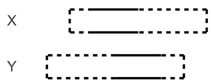
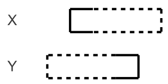
Relation	Illustration	TimeML relType disjunction
X is *older* than Y Y is *younger* than X		X [BEFORE, IBEFORE, ENDED_BY, INCLUDES, DURING] Y
X is *head to head* with Y		X [BEGINS, SIMULTANEOUS, IDENTITY, BEGUN_BY] Y
X *survives* Y Y is *survived by* X		X [INCLUDES, BEGUN_BY, IAFTER, AFTER] Y
X is *tail to tail* with Y		X [ENDED_BY, SIMULTANEOUS, IDENTITY, ENDS] Y
X *precedes* Y Y *succeeds* X		X [BEFORE, IBEFORE, ENDED_BY, INCLUDES, DURING_INV] Y
X is a *contemporary* of Y		X [INCLUDES, IS_INCLUDED, BEGUN_BY, BEGINS, DURING, DURING_INV, SIMULTANEOUS, IDENTITY, ENDS, ENDED_BY] Y
X is *born before death* of Y Y *dies after birth* of X		X [IS_INCLUDED, ENDS, DURING_INV, BEFORE, IBEFORE, INCLUDES, DURING, ENDED_BY] Y

Table 4: Freksa semi-interval relations; adapted from Freksa (1992). The superset of relations is omitted here but related there.

of TimeML full-interval relations suggested by our interpretation of Reichenbach's framework always match one of the groups of Allen relations used to represent a Freksa semi-interval relation.

For example, for two events E_1 and E_2, if the tense arrangement suggests that E_1 starts before E_2 (for example, E_1 is simple past and E_2 simple future), the available relation types for E_1 / E_2 are BEFORE, IBEFORE, DURING, ENDED_BY and INCLUDES.

The ambiguity of one interval bound in a semi-interval relation gives rise to a disjunction of possible interval relation types. For example, given that $E_{1s} < E_{2s}$, and $E_s < E_f$ for any proper interval event (e.g. its start is before its finish), the arrangement of E_1 and E_2's finish points is left unspecified. The disjunction of possible interval relation types is as follows:

- $E_{1f} < E_{2s}$: before;
- $E_{1f} = E_{2s}$: ibefore;
- $E_{1f} > E_{2s}$, $E_{1f} < E_{2f}$: during;
- $E_{1f} = E_{2f}$: ended_by;
- $E_{1f} > E_{2f}$: includes.

In each case, these disjunctions correspond to the Freksa semi-interval relation E_1 YOUNGER E_2.

5.3 Linking TimeML Events Using Reichenbach's Framework

Reichenbach's framework suggests temporal relation constraints based on the tenses and aspects of a pair of verbs. Given permanence of the reference point between the verbs, these constraints can be described using semi-interval relations. Accordingly, the full TimeML tense/aspect event-event interaction matrix according to this paper's interpretation of the framework is given in Table 5.

e1 ↓ e2 →	PAST-NONE	PAST-PROG.	PAST-PERF.	PRESENT-NONE	PRESENT-PROG.
PAST-NONE	*all*	contemporary	succeeds	survived by	survived by
PAST-PROGRESSIVE	contemporary	*contemporary*	survives	older	all
PAST-PERFECTIVE	precedes	survived by	*all*	precedes	survived by
PRESENT-NONE	survives	younger	succeeds	*contemporary*	contemporary
PRESENT-PROGRESSIVE	survives	all	survives	contemporary	*contemporary*
PRESENT-PERFECTIVE	all	all	succeeds	survived by	survived by
FUTURE-NONE	succeeds	younger	after	succeeds	younger
FUTURE-PROGRESSIVE	survives	dies after birth	survives	younger	dies after birth
FUTURE-PERFECTIVE	after	younger	after	younger	younger

e1 ↓ e2 →	PRESENT-PERF.	FUTURE-NONE	FUTURE-PROG.	FUTURE-PERF.
PAST-NONE	all	precedes	survived by	before
PAST-PROGRESSIVE	all	older	born before death	older
PAST-PERFECTIVE	precedes	before	survived by	before
PRESENT-NONE	survives	precedes	older	older
PRESENT-PROGRESSIVE	survives	older	born before death	older
PRESENT-PERFECTIVE	*all*	before	survived by	before
FUTURE-NONE	after	*all*	contemporary	survived by
FUTURE-PROGRESSIVE	survives	contemporary	*contemporary*	survives
FUTURE-PERFECTIVE	after	survived by	survived by	*all*

Table 5: TimeML tense/aspect pairs with the Freksa semi-intervals relations they suggest, according to this paper's interpretation of Reichenbach's framework. These semi-intervals correspond to disjunctions of TimeML interval relations.

6 Validating the Framework

So far, this paper has discussed the temporal relation typing problem, the differing tense representations provided by Reichenbach and TimeML, and an interpretation of Reichenbach's framework that permits temporal relation type constraint in TimeML. This section details the method for and presents results of validating Reichenbach's framework.

6.1 Context Modelling

Temporal context (detailed in Section 3.2) is defined as a set of events that have a common reference time, where the grammatical rule of sequence of tenses is followed. Lacking tools for annotating temporal context, it may instead be modelled in a variety of ways, based on arrangements of speech time and reference time, and the sentence-distance between a given pair of verb events.

Based on the hypothesis that events in a single temporal context will generally be distributed closely to one another in a text, proximity-based modelling of temporal context is evaluated. This assumes that all verbs within a certain proximity bound are considered to be in the same context. This is tested for single-sentence (e.g. all verbs in the same sentence are in the same temporal context, and no others), and for adjacent-sentence (verbs in adjacent sentences are in the same temporal context).

Because permanence of the reference point requires a shared reference time, for tenses to be meaningful in their context, the speech time must remain static. The "same SR" context refers to modelling of temporal context as a situation where the ordering of reference and speech times remains constant (in terms of one preceding, occurring with or following the other). This simple same-ordering constraint on S and R does not preclude situations where speech or reference time move, but still remain in roughly the same order (e.g. if reference time moves from 9pm to 9.30pm when speech time is 3pm), which are in fact changes of temporal context (either because R is no longer shared or because S has moved).

6.2 Results

Results are given in Table 6. In this table, a "consistent TLINK" is one where the relation type given in the ground truth is a member of the disjunction of relation types suggested by Reichenbach's framework. Separate figures are provided for performance including and excluding cases where the disjunction of all link types is given. This is because consistency given "no constraint" is not useful.

Context model	TLINKs	Consistent	Non-“all”	Non-“all” consistent
None (all pairs)	1 167	81.5%	481	55.1%
Same sentence, same SR	300	88.0%	95	62.1%
Same sentence	600	71.2%	346	50.0%
Same / adjacent sentence, same SR	566	**91.9%**	143	**67.8%**
Same / adjacent sentence	913	78.3%	422	53.1%

Table 6: Consistency of temporal orderings suggested by Reichenbach’s framework with ground-truth data. The non-all columns refer to cases in which there was relation constraint, e.g., the framework did not suggest “all relation types possible”.

6.3 Analysis

Interpreted in this way, Reichenbach’s framework is generally consistent with TimeBank, supporting the framework’s suggestions of event-event ordering among pairs of tensed verb events.

Although the proportion of inconsistent links (ignoring unconstrained cases) is noticeable – 32.2% in the best scenario – it is sufficiently strong to support the framework. The magnitude of inconsistency is comparable with inter-annotator *dis*agreement on TimeBank’s temporal relation labels (0.23) when the crudeness of the proposed temporal context models is taken into account. IAA for tense and aspect labels in TimeBank – critical to correct application of Reichenbach’s framework – are much higher (see Section 4.3). The fact that temporal context is derived from models and not explicit gold-standard annotation is also likely a significant source of noise in agreement.

The “same SR” context yields good results, though has limited applicability (e.g., it halves the set of considered same-sentence pairings). As both arguments having the same S and R occurs when they have the same TimeML tense, the only effective variant in these cases – in terms of data that contributes to Reichenbachian interpretation – is the TimeML aspect value. When given the constraint that both arguments have the same TimeML tense, the measured consistency of the framework increases. This hints that the ordering and positions of S and R are critical factors in relating tensed events, and considering them may lead to improvements in temporal relation labelling techniques that rely on aspect, such as that of Costa and Branco (2012).

Enlarging the context “window” to include adjacent sentences improves consistency. It may be that linked events within sentences are often between main events and embedded clauses or reported speech. It is also possible that single sentences may contain repositionings of the reference point that persist in following sentences, so that a single sentence does not exhibit internal permanence but permanence exists between it and following sentences. Close investigation into the typical scoping and switching of temporal context could help explain this phenomenon and lead to better models of temporal context.

The results suggest Reichenbach’s framework is accurate and capable of temporally ordering events.

7 Conclusion

This paper set out to validate Reichenbach’s framework of tense and aspect in the context of event ordering. The framework was found to be supported by human-annotated ground-truth data. This result provides empirical support for this established account of tense and aspect. In its finding, this paper also details a technique for reasoning about the temporal order of verb events in discourse.

Reichenbach’s framework is a powerful tool for ordering events (and times) within a given context. It transparently informs approaches to many complex tasks, including automatic temporal ordering, timex normalisation, machine translation, clinical summarisation, and natural language generation. The approach detailed here requires *temporal context* to exploit the framework. However, it is not yet clear *how* to automatically determine the bounds of temporal contexts. To this end, future work can consider the annotation of temporal context, in order to aid high-precision temporal information extraction from discourse. Further, the argument that semi-interval reasoning is suitable for temporal information from text is supported by this empirical work, prompting more investigation into its use in the linguistic context.

Acknowledgments Thanks is due James Pustejovsky for early discussions on Reichenbach’s framework and TimeML, and to the anonymous reviewers for their helpful feedback.

References

Allen, J. (1983). Maintaining knowledge about temporal intervals. *Communications of the ACM 26*(11), 832–843.

Costa, F. and A. Branco (2012). Aspectual type and temporal relation classification. In *Proc. EACL*, pp. 266–275.

Derczynski, L. and R. Gaizauskas (2011a). An Annotation Scheme for Reichenbach's Verbal Tense Structure. In *Proc. Joint ACL-ISO Workshop on Interoperable Semantic Annotation*.

Derczynski, L. and R. Gaizauskas (2011b). A corpus-based study of temporal signals. In *Proc. Conference on Corpus Linguistics*.

Dowty, D. (1986). The effects of aspectual class on the temporal structure of discourse: semantics or pragmatics? *Linguistics and philosophy 9*(1), 37–61.

Eijck, J. v. and H. Kamp (2010). Discourse representation in context. In J. v. Benthem and A. ter Meulen (Eds.), *Handbook of Logic and Language*, pp. 181 – 252. Elsevier.

Elson, D. and K. McKeown (2010). Tense and Aspect Assignment in Narrative Discourse. In *Proc. International Conference in Natural Language Generation*.

Fillmore, C. (1971). *Lectures on deixis*. CSLI Publications Stanford, California.

Freksa, C. (1992). Temporal reasoning based on semi-intervals. *Artificial intelligence 54*(1), 199–227.

Hepple, M., A. Setzer, and R. Gaizauskas (2007). USFD: preliminary exploration of features and classifiers for the TempEval-2007 tasks. In *Proc. International Workshop on Semantic Evaluation*, pp. 438–441.

Hornstein, N. (1990). *As time goes by: Tense and universal grammar*. MIT Press.

Klein, W. (1994). *Time in language*. Germanic linguistics. London [u.a.]: Routledge.

Llorens, H., L. Derczynski, R. Gaizauskas, and E. Saquete (2012). TIMEN: An Open Temporal Expression Normalisation Resource. In *Proc. International Conference on Language Resources and Evaluation*, pp. 3044–3051.

Mani, I., J. Pustejovsky, and R. Gaizauskas (2005). *The Language of Time: A Reader*. Oxford University Press.

Mazur, P. and R. Dale (2010). WikiWars: A new corpus for research on temporal expressions. In *Proc. EMNLP*, pp. 913–922.

Portet, F., E. Reiter, A. Gatt, J. Hunter, S. Sripada, Y. Freer, and C. Sykes (2009). Automatic generation of textual summaries from neonatal intensive care data. *Artificial Intelligence 173*(7-8), 789–816.

Pustejovsky, J., P. Hanks, R. Sauri, A. See, R. Gaizauskas, A. Setzer, D. Radev, B. Sundheim, D. Day, L. Ferro, and M. Lazo (2003). The TimeBank corpus. In *Proc. Conference on Corpus Linguistics*, pp. 647–656.

Pustejovsky, J., B. Ingria, R. Sauri, J. Castano, J. Littman, R. Gaizauskas, A. Setzer, G. Katz, and I. Mani (2005). The specification language TimeML. In *The Language of Time: A Reader*, pp. 545–557. Oxford University Press.

Reichenbach, H. (1947). The tenses of verbs. In *Elements of Symbolic Logic*. Macmillan.

Verhagen, M., R. Sauri, T. Caselli, and J. Pustejovsky (2010). SemEval-2010 task 13: TempEval-2. In *Proc. International Workshop on Semantic Evaluation*, pp. 57–62.

Webber, B. (1987). The interpretation of tense in discourse. In *Proc. ACL*, pp. 147–154.