Doing Time: Inducing Temporal Graphs

by

Philip James Bramsen

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Masters of Engineering in Computer Science and Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2006

Author . . . .
Department of Electrical Engineering and Computer Science
June 22, 2006

Certified by . . . .
Regina Barzilay
Assistant Professor
Thesis Supervisor

Accepted by . . . .
Arthur C. Smith
Chairman, Department Committee on Graduate Students

Doing Time: Inducing Temporal Graphs

by

Philip James Bramsen

Submitted to the Department of Electrical Engineering and Computer Science
on June 22, 2006, in partial fulfillment of the
requirements for the degree of
Masters of Engineering in Computer Science and Engineering

Abstract

We consider the problem of constructing a directed acyclic graph that encodes temporal relations found in a text. The unit of our analysis is a temporal segment, a fragment of text that maintains temporal coherence. The strength of our approach lies in its ability to simultaneously optimize pairwise ordering preferences and global constraints on the graph topology. Our learning method achieves 83% F-measure in temporal segmentation and 84% accuracy in inferring temporal relations between two segments.

Thesis Supervisor: Regina Barzilay
Title: Assistant Professor

Acknowledgements

This project was the brainchild of my thesis advisor, Professor Regina Barzilay. It was her foresight that set me on the way: My thesis proposal from August '05 is a startling accurate description of the work done. It was her determination which kept this project rolling. Professor Peter Szolovits was also an important part of the early stages of this undertaking.

Special thanks goes to my lab mates Pawan Deshpande and Yoong Keok Lee. Midway through my first term, Repetitive Strain Injury (RSI) began to make life difficult. By January, I had the humbling experience of truly needing assistance. But working side-by-side with Pawan late January through February was a excellent teamwork experience. As paper deadlines approached, both Pawan and Yoong Keok dug in to the project, making the papers possible. And so a very special team is born, one I will remember.

Other lab mates were also helpful. Igor Malioutov was always willing to offer feedback and thoughtful discussion. It was he who first suggested using Integer Linear Programming to solve the global ordering problem. Ben Snyder was also helpful because of his experience with annotation and his academically unbiased personality. Kevin Simler was especially helpful in his willingness to cheerfully jump into a problem with me and think it through.

Beyond the lab, I would like to thank my family: Frank, Annette, Peter and Sarah, and my future family: Diane. Without their faithful support, love, and understanding during my years at MIT, this thesis would not have been possible.

I would like to also thank MIT friends and advisors, Tony Eng, Jim Glass, and Danny Park.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Understanding the temporal flow of discourse is a significant aspect of text comprehension. Consequently, temporal analysis has been a focus of linguistic research for quite some time. Temporal interpretation encompasses levels ranging from the syntactic to the lexico-semantic [22, 19] and includes the characterization of temporal discourse in terms of rhetorical structure and pragmatic relations [10, 24, 20, 16].

Besides its linguistic significance, temporal analysis has important practical implications. In multidocument summarization, knowledge about the temporal order of events can enhance both the content selection and the summary generation processes [2]. In question answering, temporal analysis is needed to determine when a particular event occurs and how events relate to each other. Some of these needs can be addressed by emerging technologies for temporal analysis [26, 18, 15, 4].

This paper characterizes the temporal flow of discourse in terms of *temporal segments* and their ordering. We define a temporal segment to be a fragment of text that does not exhibit abrupt changes in temporal focus [25]. A segment may contain more than one event or state, but the key requirement is that its elements maintain temporal coherence. For instance, a medical case summary may contain segments describing a patient's admission, his previous hospital visit, and the onset of his original symptoms. Each of these segments corresponds to a different time frame, and is clearly delineated as such in a text.

Our ultimate goal is to automatically construct a graph that encodes ordering between temporal segments. The key premise is that in a coherent document, temporal progression is reflected in a wide range of linguistic features and contextual dependencies. In some cases, clues to segment ordering are embedded in the segments themselves. For instance, given a pair of adjacent segments, the temporal adverb *next day* in the second segment is a strong predictor of a precedence relation. In other cases, we can predict the right order between a pair of segments by analyzing their relation to other segments in the text. The interaction between pairwise ordering decisions can easily be formalized in terms of constraints on the graph topology. An obvious example of such a constraint is prohibiting cycles in the ordering graph. We show how these complementary sources of information can be incorporated in a model using global inference.

We evaluate our temporal ordering algorithm on a corpus of medical case summaries. Temporal analysis in this domain is challenging in several respects: a typical summary exhibits no significant tense or aspect variations and contains few absolute time markers. We demonstrate that humans can reliably mark temporal segments and determine segment ordering in this domain. Our learning method achieves 83% F-measure in temporal segmentation and 84% accuracy in inferring temporal relations between two segments.

Our contributions are twofold:

Temporal Segmentation We propose a fully automatic, linguistically rich model for temporal segmentation. Most work on temporal analysis is done on a finer granularity than proposed here. Our results show that the coarse granularity of our representation facilitates temporal analysis and is especially suitable for domains with sparse temporal anchors.

Segment Ordering We introduce a new method for learning temporal ordering. In contrast to existing methods that focus on pairwise ordering, we explore strategies for global temporal inference. The strength of the proposed model lies in its ability to simultaneously optimize pairwise ordering preferences and global constraints on graph topology. While the algorithm has been applied at the segment level, it can be used with other temporal annotation schemes.

Chapter 2

Related Work

Temporal ordering has been extensively studied in computational linguistics [20, 25, 14, 16, 17]. Prior research has investigated a variety of language mechanisms and knowledge sources that guide interpretation of temporal ordering, including tense, aspect, temporal adverbials, rhetorical relations and pragmatic constraints. In recent years, the availability of annotated corpora, such as TimeBank [21], has triggered the use of machine-learning methods for temporal analysis [18, 15, 4]. Typical tasks include identification of temporal anchors, linking events to times, and temporal ordering of events.

Since this paper addresses temporal ordering, we focus our discussion on this task. Existing ordering approaches vary both in terms of the ordering unit — it can be a clause, a sentence or an event — and in terms of the set of ordering relations considered by the algorithm. Despite these differences, most existing methods have the same basic design: each pair of ordering units (i.e., clauses) is abstracted into a feature vector and a supervised classifier is employed to learn the mapping between feature vectors and their labels. Features used in classification include aspect, modality, event class, and lexical representation. It is important to note that the classification for each pair is performed independently and is not guaranteed to yield a globally consistent order.

In contrast, our focus is on globally optimal temporal inference. While the importance of global constraints has been previously validated in symbolic systems for temporal analysis [11, 27], existing corpus-based approaches operate at the local level. These improvements achieved by a global model motivate its use as an alternative to existing pairwise methods.

Chapter 3

Representation of Temporal Flow

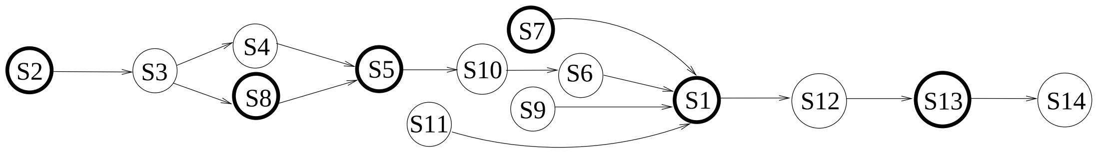
3.1 Introduction

Our goal is to learn the chronological order of events found in narrative text. To do so, we propose a two-layer annotation scheme for text that specifies temporal order.

Temporal Segments We see the text as a sequence of *temporal segments*, where the temporal focus within each temporal segment is constant, but there are abrupt changes in temporal focus between segments. The first layer of annotation is the segment boundaries marked in the text.

Temporal Directed Acyclic Graphs (TDAGs) Building on temporal segments as nodes, we represent chronological precedence relations found in a text with a Temporal Directed Acyclic Graph (TDAG). If two temporal segments have a chronological order, the TDAG corresponding to the text has a directed edge from the earlier to the later segment. Unrelated temporal segments do not have a connection in the TDAG.

This chapter begins with a detailed definition of the representation and annotation scheme. In the second section, we review other temporal annotation methodologies. We conclude with a comparison between our approach and the pre-existing approaches, as well as discussion of the strengths and weaknesses of our representation and annotation.



S1	A 32-year-old woman was admitted to the hospital because of left subcostal pain...
S2	The patient had been well until four years earlier,
S5	Three months before admission an evaluation elsewhere included an ultrasonographic examination, a computed tomographic (CT) scan of the abdomen...
S7	She had a history of eczema and of asthma...
S8	She had lost 18 kg in weight during the preceding 18 months.
S13	On examination the patient was slim and appeared well. An abdominal examination revealed a soft systolic bruit... and a neurologic examination was normal...

Figure 3-1: An example of the transitive reduction of a TDAG for a case summary. A sample of segments corresponding to the nodes marked in bold is shown in the table.

3.2 Representation

3.2.1 Temporal Segments

We claim that blocks of temporally related, textually sequential clauses — which we call temporal segments — are the language vehicle for communicating narratives. We view text as a sequence of *temporal segments*. The length of a segment can range from a single clause to a sequence of adjacent sentences. Temporal focus is retained within a segment, but fundamentally changes between segments. We argue that for the purposes of the communication of the text to the reader, events within a segment of occur at single time.

When writers communicate a narrative, they do not time-stamp every event or state-clause with an explicit reference to time. Rather, for the majority of events, they demarcate chronological progression with grammar or discourse structure that relates clauses to more temporally explicit statements.

Consider the following example:

> A 12-year-old boy was evaluated in the hematology clinic of this hospital... For one week, he had had fatigue that had caused him to miss school, and his appetite had decreased. He also had intermittent cramping, pain in his arms and

legs, and anterior chest pain that was worse with movement, deep breathing, and coughing. (Case 19-2004, NEJM Corpus)

Based on the above passage, readers can understand that the boy was evaluated at some time in the past. They can also interpret that, for a week before the clinic visit, the boy had the symptoms described in the latter lines of the excerpt. Although the statements about symptoms contain some temporal information (e.g. they are all in the past tense), the strongest evidence for their position in the order of events can be determined from the words "for one week, he had had..."

The second and third sentences of the example form a *temporal segment*: they are a set of textually contiguous phrases that happened at the same time. However, note that the event clauses within the temporal segment do not necessarily start, end or progress together. In fact, we could clarify the text (without contradicting it) with additional text to the effect that the symptoms had occurred in some specific order: "his decreased appetite caused the fatigue," "the cramping symptoms only appeared on the last day before his evaluation at the clinic," or "he missed the second and third day of school that week." All we mean by calling the sentences a temporal segment is that the author of the text is not communicating to us information that specifies and order between the events; the author grouped the events to share a temporal reference.

The first sentence ("A 12 year-old boy was evaluated") also qualifies as a temporal segment, if we do not consider the context of the excerpt. It is distinct from the temporal segment of symptoms because it clearly happens at another time.

Consider these illustrative examples of where temporal segment boundaries occur. Each segment boundary is marked with "####":

- "Periumbilical pain had begun the day before the initial admission, #### after the patient had eaten cookies and an apple" (Case 21-2000). "After" indicated that eating the cookies is in a different temporal segment from the pain that followed.

- "#### The abdominal pain worsened." (Case 21-2000). The verb indicates a change in pain from whatever the pain intensity was before. A new temporal segment begins.

- “The temperature was 37.8 degrees Celsius, the pulse was... #### On physical examination, the patient appeared to be comfortable at rest” (Case 20-1997). This boundary is based on world knowledge: A segment boundary belongs here because patient’s vital signs are taken first, before physical examination.

In the following section, on the chronological ordering graph, there are more examples of temporal segmentation.

Annotators received detailed instructions (see Appendix A). The annotation guidelines first introduce the concept of temporal segments, giving an intuitive explanation. The directions also include prototypical examples, common misconceptions, and default annotation procedure for certain ambiguous situations.

3.2.2 Temporal Directed Acyclic Graphs

We represent the order of events as a Temporal Directed Acyclic Graph (TDAG) one per narrative document. We define a TDAG as follows: Edges capture chronological precedence relations between nodes, where nodes represent text from the document. However, for a TDAG, precedence means that two nodes can be ordered if the start of one node clearly precedes start of the other. Because the graph encodes an order, cycles are prohibited. We do not require the graph to be fully connected if the precedence relation between two nodes is not specified in the text, then the corresponding nodes are not to be connected; where the contents of two TDAG nodes cannot be ordered, then there should be no edge between them.

We use the nodes of the TDAG to represent temporal segments. The combined representation is a meaningful abstraction: it is the temporal order between segments of text, each of which has a unifying internal temporal focus. For properly segmented text, all clauses within each segment correctly inherit from their parent segments the chronological relationships that their parent segments have.

Returning to the isolated examples above, the order required in the TDAG is clear. We will indicate order with an arrow.

- “Periumbilical pain had begun the day before the initial admission, after the patient had eaten cookies and an apple" (Case 21-2000).

- "The temperature was 37.8 degrees Celsius, the pulse was... On physical examination, the patient appeared to be comfortable at rest" (Case 20-1997).

Figure 3-1, shows a sample of several temporal segments from a medical case summary. Consider the segment **S13** of this text. This segment describes an examination of a patient, encompassing several events and states (i.e., an abdominal and neurological examination). All of them belong to the same time frame; the chronological order between these events is not explicitly outlined in the text.

We can infer the order between most segments by transitivity. For example, **S2** in Figure 3-1precedes all events. **S8**, which started 18 months ago, precedes **S5**, which began three months ago. **S13**, which happened at the hospital, follows admission, **S1**, and all events that came before admission. The figure also includes segments that are not easy to order, even though they are textually and temporally close. For instance, consider the segments **S5** and **S7**, which describe the patient's previous tests and her history of eczema. Any order between the two events is consistent with our interpretation of the text, therefore we cannot determine the precedence relation between the segments **S5** and **S7**.

For our task, we add another component to the TDAG definition: some statements about the patient in the document can not be ordered with respect to other events. For example, "the patient gardened" is somewhat "out of time". It is typically impossible to read from the text the starting time of these statements. However, we do know that these fact-statements were included because they were relevant for the hospital stay: therefore, they must predate the hospital admission time.

Another way we look at these "out of time" statements is as being related to what we will call the central event time of the case history. This is distinct to speech time, reference time, and document time [22], which were discussed in Chapter 2. The central event of our medical case records is the primary hospital admission. Using hospital admission time was sensible from standpoint of Medical Informatics: Patient attributes that have no reference time are most relevant as attributes from the time of admission tasks for which TDAGs could be useful: the patient attributes are relevant to medical community readers of the case records because the patient has the attributes when she is admitted.

An example is in Figure 3-1, **S7**, is interpreted as a statement about the patient, all we know is that the history of eczema was present before admission. Annotators were given a simple description of the TDAG representation. They were also given several example documents and their associated TDAGs. Annotators claimed the task was easy to understand.

3.2.3 Other Temporal Representations

In contrast to many existing temporal representations [1, 21], TDAG is a coarse annotation scheme: it does not capture interval overlap and distinguishes only a subset of commonly used ordering relations. Our choice of this representation, however, is not arbitrary. The selected relations are shown to be useful in text processing applications [27] and can be reliably recognized by humans. Moreover, the distribution of event ordering links under a more refined annotation scheme, such as TimeML, shows that our subset of relations covers a majority of annotated links [21].

3.3 Strengths and Weaknesses of Temporal Segments & TDAGs

In this section, we first discuss criteria for evaluating temporal representation schemes and annotation. Having established the criteria, we analyze the strengths and the weaknesses of our approach, paying particular attention to how it compares with other temporal representations.

3.3.1 Standards for a Temporal Annotation Scheme

There are several criteria to keep in mind. Mani, Wilson, Ferro, and Sundheim in (ref) suggest the following evaluation criteria for an annotation scheme:

- Simplicity with Precision: Simple enough to be used by humans, yet precise for natural language processing tasks.

- Naturalness: Able to reflect the full range of temporal distinctions which humans are capable of annotating.
- Expressiveness: Able to fully express the information present in the text, to whatever extent humans are capable of annotating.
- Reproducibility: Consistent examples are available for annotators and are easy to create.

For practicality, we amend these standards with one more requirement:

- Computability: The representation should be computationally tractable for whatever task it is intended.

As noted by Mani and his colleagues, a fully natural annotation scheme will be beyond the reach of current automatic systems. Ideally, our annotation scheme would be a subset of the more complete schema, but it is not easy to extract our representation of temporal segments and TDAGs from more intricate temporal annotation schemes. To satisfy the computability criteria for our statistical NLP approach, we sacrifice some naturalness and expressiveness. However, we will argue that the temporal relations we can extract with our approach are significant and useful.

3.3.2 Strengths of our Representation and Annotation Scheme

In this section we consider the strengths of our approach. The annotation represents the most valuable information in a concise, computable form. Annotators demonstrated strong comprehension the annotation task and we have high inter-annotator agreement. In addition, our work has ramifications for linguistics.

The Representation Effectively Reveals the Most Valuable Information

Paul Grice [12], first stated the cooperative principal of Pragmatics, ”Make your contribution such as it is required, at the stage at which it occurs, by the accepted purpose or direction of the talk exchange in which you are engaged.” The cooperative principal claims that people tend to form their communications to effectively and efficiently transfer information. The cooperative principal gives rise to Grice's four maxims of conversation, widely recognized as foundational in Semantics Pragmatics (ref prof): the maxims of quality, quantity, relevance, and clarity.

The maxims give credence to the concept of temporal segments

Quantity and Relevance: make your communication as informative as required, and not more so. Writers communicate in temporal segments because they are trying to communicate the information that is most important, not all the information associated with every event. Temporal Segments and TDAGs are an attempt to represent the information that authors are trying to communicate.

- Clarity: be brief, unambiguous, and orderly. Temporal segments are orderly; segments are an efficient mechanism authors use to be brief without being ambiguous.

- Quality: do not say that for which you do not have complete evidence. Linguists claim that the Gricean Maxims can be viewed from the listeners' standpoint. Temporal Segmentation and TDAGs are minimalist and do not try to extract information that is not clearly delineated in the text. Temporal Segmentation is an attempt to put text with uncertain temporal order into blocks, because the blocks can be ordered with more certainty.

Naturalness — Ease of Annotation

Annotators found the segmentation and segment ordering tasks straightforward. Upon hearing an intuitive explanation of the representation and seeing an example document, annotators understood immediately what temporal segments and precedence graphs represented. For temporal segmentation, a page of specific instructions was required to make the annotators aware of non-obvious and easily missed segment boundary decision factors. Take for example, "the child's mother said he had run a high fever." The relevant event is the high fever, not the mother's speech. Subtleties like this one can have an effect on the segmentation.

Reproducibility — Inter-Annotator Agreement

As we will show with formal agreement results in Chapter 6, there was high inter-annotator agreement. Inter-annotator agreement is proof we have reproducibility.

Computability — For Accuracy

Insofar as segmentation of text into temporal segments is accurate, segmentation can improve temporal ordering accuracy. If the segments are linguistically correct, the aggregation of information from distinct atomic event-clauses into temporal segments means there is more information on which to base ordering decisions. When we did attempt to apply our temporal ordering methods to an unsegmented text, the methods performed very poorly see Chapter 7.

Computability — For Speed

Without segmenting into temporally consistent chunks, a time precedence graph among all events must order each event in the text with respect to all other events. For the documents in our medical records corpus, chunking into temporal segments dramatically reduces the number of units that must be ordered (from an average of well over a hundred per document, to under thirty). This difference increases by an order of magnitude when relation between all node pairs is known after full transitive closure is applied to the TDAGs. To expose enough information for learning, transitive closure is required.

Advantage for Linguistics

Our work is a real world test of a concept of temporal segments similar to Webber's concept of temporal focus [24]. Success of the representation constitutes a confirmation of a theory in Discourse Linguistics. To a lesser extent, success of the TDAG representation also benefit linguistics: Features that prove useful for learning the temporal ordering could help linguists understand what linguistic cues help people order events.

3.3.3 Disadvantages

The annotation scheme of temporal segments and TDAGs has one major shortcoming: it is coarse. Compared to other annotation standards, our annotation scheme is less expressive.

Limited Relations

While other schemes are expressive and natural, with broad coverage of temporal relations, we represent only one. We could argue that temporal segments give us some knowledge of the relations between clauses within the temporal segment. However, temporal segments often contain both events ordered as by the narrative-progressive (precedence) relation and events that are truly contemporaneous. In reality, temporal segments only tell us that there is not enough evidence to order the events within the temporal segment. The only strong claim is that for the purposes of the text, we can view the contents of temporal segments as concurrent and we know that the clauses within two temporal segments have the same temporal order as their parent segments.

Further Limitations

Intervals are represented in virtually all other schemas. Some also represent event repeat frequency. We only guarantee the start time of each temporal segment. In some situations interval annotation could be very useful: “The patient had a high fever for two weeks” tells us much more than “the patient had a high fever for two days.”

However, our representation does not eliminate any information about temporal order, frequency, or intervals. Therefore, as a pre-processing step, our representation and method would only help, not hinder, a system designed to extract more information.

3.3.4 Unresolved Questions Regarding the Annotation Scheme

Cross-Language Portability Some other temporal annotation schemes have been evaluated for language portability. Though we strongly suspect that temporal segments and TDAGs will generalize well to other languages, we have not investigated language portability.

Chapter 4

Method for Temporal Segmentation

Our first goal is to automatically predict shifts in temporal focus that are indicative of segment boundaries. Linguistic studies show that speakers and writers employ a wide range of language devices to signal change in temporal discourse [3]. For instance, the presence of the temporal anchor *last year* indicates the lack of temporal continuity between the current and the previous sentence. However, many of these predictors are heavily context-dependent and, thus, cannot be considered independently. Instead of manually crafting complex rules controlling feature interaction, we opt to learn them from data.

We model temporal segmentation as a binary classification task. Given a set of candidate boundaries (e.g., sentence boundaries), our task is to select a subset of the boundaries that delineate temporal segment transitions. To implement this approach, we first identify a set of potential boundaries. Our analysis of the manually-annotated corpus reveals that boundaries can occur not only between sentences, but also within a sentence, at the boundary of syntactic clauses. We automatically segment sentences into clauses using a robust statistical parser [6]. Next, we encode each boundary as a vector of features. Given a set of annotated examples, we train a classifier[1] to predict boundaries based on the following feature set:

Lexical Features Temporal expressions, such as *tomorrow* and *earlier*, are among the strongest markers of temporal discontinuity [20, 3]. In addition to a well-studied set of domain-independent temporal markers, there are a variety of domain-specific temporal

[1] BoosTexter package [23].

markers. For instance, the phrase *initial hospital visit* functions as a time anchor in the medical domain.

To automatically extract these expressions, we provide a classifier with n-grams from each of the candidate sentences preceding and following the candidate segment boundary.

Topical Continuity Temporal segmentation is closely related to topical segmentation [5]. Transitions from one topic to another may indicate changes in temporal flow and, therefore, identifying such transitions is relevant for temporal segmentation.

We quantify the strength of a topic change by computing a cosine similarity between sentences bordering the proposed segmentation. This measure is commonly used in topic segmentation [13] under the assumption that change in lexical distribution corresponds to topical change.

Positional Features Some parts of the document are more likely to exhibit temporal change than others. This property is related to patterns in discourse organization of a document as a whole. For instance, a medical case summary first discusses various developments in the medical history of a patient and then focuses on his current conditions. As a result, the first part of the summary contains many short temporal segments. We encode positional features by recording the relative position of a sentence in a document.

Syntactic Features Because our segment boundaries are considered at the clausal level, rather than at the sentence level, the syntax surrounding a hypothesized boundary may be indicative of temporal shifts. This feature takes into account the position of a word with respect to the boundary. For each word within three words of the hypothesized boundary, we record its part-of-speech tag along with its distance from the boundary. For example, NNP_{+1} encodes the presence of a proper noun immediately following the proposed boundary.

Chapter 5

Learning to Order Segments

Our next goal is to automatically construct a graph that encodes ordering relations between temporal segments. One possible approach is to cast graph construction as a standard binary classification task: predict an ordering for each pair of distinct segments based on their attributes alone. If a pair contains a temporal marker, like *later*, then accurate prediction is feasible. In fact, this method is commonly used in event ordering [18, 15, 4]. However, many segment pairs lack temporal markers and other explicit cues for ordering. Determining their relation out of context can be difficult, even for humans. Moreover, by treating each segment pair in isolation, we cannot guarantee that all the pairwise assignments are consistent with each other and yield a valid TDAG.

Rather than ordering each pair separately, our ordering model relies on global inference. Given the pairwise ordering predictions of a local classifier[1], our model finds a globally optimal assignment. In essence, the algorithm constructs a graph that is maximally consistent with individual ordering preferences of each segment pair and at the same time satisfies graph-level constraints on the TDAG topology.

In Section 5.2, we present three global inference strategies that vary in their computational and linguistic complexity. But first we present our underlying local ordering model.

[1] The perceptron classifier.

5.1 Learning Pairwise Ordering

Given a pair of segments (i, j), our goal is to assign it to one of three classes: forward, backward, and null (not connected). We generate the training data by using all pairs of segments (i, j) that belong to the same document, such that i appears before j in the text.

The features we consider for the pairwise ordering task are similar to ones used in previous research on event ordering [18, 15, 4]. Below we briefly summarize these features.

Lexical Features This class of features captures temporal markers and other phrases indicative of order between two segments. Representative examples in this category include domain-independent cues like *years earlier* and domain-specific markers like *during next visit*. To automatically identify these phrases, we provide a classifier with two sets of n-grams extracted from the first and the second segments. The classifier then learns phrases with high predictive power.

Temporal Anchor Comparison Temporal anchors are one of the strongest cues for the ordering of events in text. For instance, medical case summaries use phrases like *two days before admission* and *one day before admission* to express relative order between events. If the two segments contain temporal anchors, we can determine their ordering by comparing the relation between the two anchors. We identified a set of temporal anchors commonly used in the medical domain and devised a small set of regular expressions for their comparison.[2] The corresponding feature has three values that encode preceding, following and incompatible relations.

Segment Adjacency Feature Multiple studies have shown that two subsequent sentences are likely to follow a chronological progression [3]. To encode this information, we include a binary feature that captures the adjacency relation between two segments.

5.2 Global Inference Strategies for Segment Ordering

Given the scores (or probabilities) of all pairwise edges produced by a local classifier, our task is to construct a TDAG. In this section, we describe three inference strategies that aim

[2]We could not use standard tools for extraction and analysis of temporal anchors as they were developed on the newspaper corpora, and are not suitable for analysis of medical text [26].

to find a consistent ordering between *all* segment pairs. These strategies vary significantly in terms of linguistic motivation and computational complexity. Examples of automatically constructed TDAGs derived from different inference strategies are shown in Figure 7.

5.2.1 Greedy Inference in Natural Reading Order (NRO)

The simplest way to construct a consistent TDAG is by adding segments in the order of their appearance in a text. Intuitively speaking, this technique processes segments in the same order as a reader of the text. The motivation underlying this approach is that the reader incrementally builds temporal interpretation of a text; when a new piece of information is introduced, the reader knows how to relate it to already processed text.

This technique starts with an empty graph and incrementally adds nodes in order of their appearance in the text. When a new node is added, we greedily select the edge with the highest score that connects the new node to the existing graph, without violating the consistency of the TDAG. Next, we expand the graph with its transitive closure. We continue greedily adding edges and applying transitive closure until the new node is connected to all other nodes already in the TDAG. The process continues until all the nodes have been added to the graph.

5.2.2 Greedy Best-first Inference (BF)

Our second inference strategy is also greedy. It aims to optimize the score of the graph. The score of the graph is computed by summing the scores of its edges. While this greedy strategy is not guaranteed to find the optimal solution, it finds a reasonable approximation [7].

This method begins by sorting the edges by their score. Starting with an empty graph, we add one edge at a time, without violating the consistency constraints. As in the previous strategy, at each step we expand the graph with its transitive closure. We continue this process until all the edges have been considered.

5.2.3 Exact Inference with Integer Linear Programming (ILP)

We can cast the task of constructing a globally optimal TDAG as an optimization problem. In contrast to the previous approaches, the method is not greedy. It computes the optimal solution within the Integer Linear Programming (ILP) framework.

For a document with N segments, each pair of segments (i, j) can be related in the graph in one of three ways: forward, backward, and null (not connected). Let $s_{i \to j}$, $s_{i \leftarrow j}$, and $s_{i \not\leftrightarrow j}$ be the scores assigned by a local classifier to each of the three relations respectively. Let $I_{i \to j}$, $I_{i \leftarrow j}$, and $I_{i \not\leftrightarrow j}$ be indicator variables that are set to 1 if the corresponding relation is active, or 0 otherwise.

The objective is then to optimize the score of a TDAG by maximizing the sum of the scores of all edges in the graph:

$$\max \sum_{i=1}^{N} \sum_{j=i+i}^{N} s_{i \to j} I_{i \to j} + s_{i \leftarrow j} I_{i \leftarrow j} + s_{i \not\leftrightarrow j} I_{i \not\leftrightarrow j} \tag{5.1}$$

subject to:

$$I_{i \to j}, I_{i \leftarrow j}, I_{i \not\leftrightarrow j} \in \{0, 1\} \quad \forall\, i, j = 1, \ldots N, i < j \tag{5.2}$$

$$I_{i \to j} + I_{i \leftarrow j} + I_{i \not\leftrightarrow j} = 1 \quad \forall\, i, j = 1, \ldots N, i < j \tag{5.3}$$

We augment this basic formulation with two more sets of constraints to enforce validity of the constructed TDAG.

Transitivity Constraints The key requirement on the edge assignment is the transitivity of the resulting graph. Transitivity also guarantees that the graph does not have cycles. We enforce transitivity by introducing the following constraint for every triple (i, j, k):

$$I_{i \to j} + I_{j \to k} - 1 \leq I_{i \to k} \tag{5.4}$$

If both indicator variables on the left side of the inequality are set to 1, then the indicator variable on the right side must be equal to 1. Otherwise, the indicator variable on the right can take any value.

Connectivity Constraints The connectivity constraint states that each node i is connected to at least one other node and thereby enforces connectivity of the generated TDAG. We introduce these constraints because manually-constructed TDAGs do not have any disconnected nodes. This observation is consistent with the intuition that the reader is capable to order a segment with respect to other segments in the TDAG.

$$(\sum_{j=1}^{i-1} I_{i \not\leftrightarrow j} + \sum_{j=i+1}^{N} I_{j \not\leftrightarrow i}) < N - 1 \tag{5.5}$$

The above constraint rules out edge assignments in which node i has null edges to the rest of the nodes.

Solving ILP Solving an integer linear program is NP-hard [9]. Fortunately, there exist several strategies for solving ILPs. We employ an efficient Mixed Integer Programming solver *lp_solve*[3] which implements the Branch-and-Bound algorithm. It takes less than five seconds to decode each document on a 2.8 GHz Intel Xeon machine.

[3] `http://groups.yahoo.com/group/lp_solve`

Chapter 6

Evaluation Set-Up

We first describe the corpora used in our experiments and the results of human agreement on the segmentation and the ordering tasks. Then, we introduce the evaluation measures that we use to assess the performance of our model.

6.1 Corpus Characteristics

We applied our method for temporal ordering to a corpus of medical case summaries. The medical domain has been a popular testbed for methods for automatic temporal analyzers [8, 27]. The appeal is partly due to rich temporal structure of these documents and the practical need to parse this structure for meaningful processing of medical data.

We compiled a corpus of medical case summaries from the online edition of The New England Journal of Medicine.[1] The summaries are written by physicians of Massachusetts General Hospital. A typical summary describes an admission status, previous diseases related to the current conditions and their treatments, family history, and the current course of treatment. For privacy protection, names and dates are removed from the summaries before publication.

The average length of a summary is 47 sentences. The summaries are written in the past tense, and a typical summary does not include instances of the past perfect. The summaries do not follow a chronological order. The ordering of information in this domain

[1] `http://content.nejm.org`

is guided by stylistic conventions (i.e., symptoms are presented before treatment) and the relevance of information to the current conditions (i.e., previous onset of the same disease is summarized before the description of other diseases).

6.2 Annotating Temporal Segmentation

Our approach for temporal segmentation requires annotated data for supervised training. We first conducted a pilot study to assess the human agreement on the task. We employed two annotators to manually segment a portion of our corpus. The annotators were provided with two-page instructions that defined the notion of a temporal segment and included examples of segmented texts. Each annotator segmented eight summaries which on average contained 49 sentences. Because annotators were instructed to consider segmentation boundaries at the level of a clause, there were 877 potential boundaries. The first annotator created 168 boundaries, while the second — 224 boundaries. We computed a Kappa coefficient of 0.71 indicating a high inter-annotator agreement and thereby confirming our hypothesis about the reliability of temporal segmentation.

Once we established high inter-annotator agreement on the pilot study, one annotator segmented the remaining 52 documents in the corpus.[2] Among 3,297 potential boundaries, 1,178 (35.7%) were identified by the annotator as segment boundaries. The average segment length is three sentences, and a typical document contains around 20 segments.

6.3 Annotating Temporal Ordering

To assess the inter-annotator agreement, we asked two human annotators to construct TDAGs from five manually segmented summaries. These summaries consist of 97 segments, and their transitive closure contain a total of 1,331 edges. We computed the agreement between human judges by comparing the transitive closure of the TDAGs. The annotators achieved a surprisingly high agreement with a Kappa value of 0.98.

After verifying human agreement on this task, one of the annotators constructed TDAGs

[2] It took approximately 20 minutes to segment a case summary.

for another 25 summaries.[3] The transitive reduction of a graph contains on average 20.9 nodes and 20.5 edges. The corpus consists of 72% forward, 12% backward and 16% null segment edges inclusive of edges induced by transitive closure. At the clause level, the distribution is even more skewed — forward edges account for 74% edges, equal for 18%, backward for 3% and null for 5%.

6.4 Evaluation Measures

We evaluate temporal segmentation by considering the ratio of correctly predicted boundaries. We quantify the performance using F-measure, a commonly used binary classification metric. We opt not to use the P_k measure, a standard topical segmentation measure, because the temporal segments are short and we are only interested in the identification of the exact boundaries.

Our second evaluation task is concerned with ordering manually annotated segments. In these experiments, we compare an automatically generated TDAG against the annotated reference graph. In essence, we compare edge assignment in the transitive closure of two TDAGs, where each edge can be classified into one of the three types: forward, backward, or null.

Our final evaluation is performed at the clausal level. In this case, each edge can be classified into one of the four classes: forward, backward, equal, or null. Note that the clause-level analysis allows us to compare TDAGs based on the automatically derived segmentation.

[3] It took approximately one hour to build a TDAG for each segmented document.

Chapter 7

Results

We evaluate temporal segmentation using leave-one-out cross-validation on our corpus of 60 summaries. The segmentation algorithm achieves a performance of 83% F-measure, with a recall of 78% and a precision of 89%.

To evaluate segment ordering, we employ leave-one-out cross-validation on 30 annotated TDAGs that overall contain 13,088 edges in their transitive closure. In addition to the three global inference algorithms, we include a majority baseline that classifies all edges as forward, yielding a chronological order.

Our results for ordering the manually annotated temporal segments are shown in Table 7.1. All inference methods outperform the baseline, and their performance is consistent with the complexity of the inference mechanism. As expected, the ILP strategy, which supports exact global inference, achieves the best performance — 84.3%.

An additional point of comparison is the accuracy of the pairwise classification, prior to the application of global inference. The accuracy of the local ordering is 81.6%, which is lower than that of ILP. The superior performance of ILP demonstrates that accurate global inference can further refine local predictions. Surprisingly, the local classifier yields a higher accuracy than the two other inference strategies. Note, however, the local ordering procedure is not guaranteed to produce a consistent TDAG, and thus the local classifier cannot be used on its own to produce a valid TDAG.

Table 7.2 shows the ordering results at the clausal level. The four-way classification is computed using both manually and automatically generated segments. Pairs of clauses

Algorithm	Accuracy
Integer Linear Programming (ILP)	**84.3**
Best First (BF)	78.3
Natural Reading Order (NRO)	74.3
Baseline	72.2

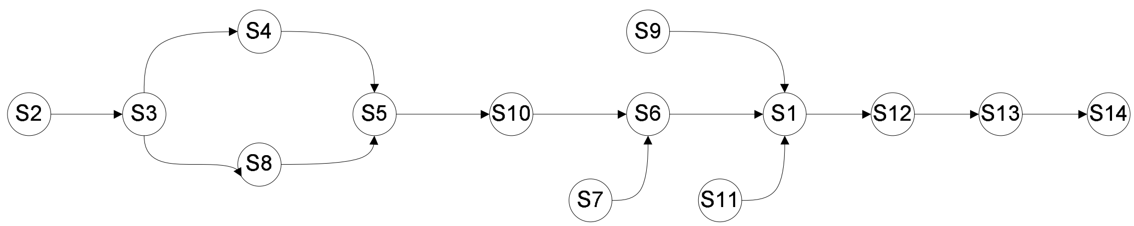
Table 7.1: Accuracy for 3-way ordering classification over manually-constructed segments.

that belong to the same segment stand in the equal relation, otherwise they have the same ordering relation as the segments to which they belong.

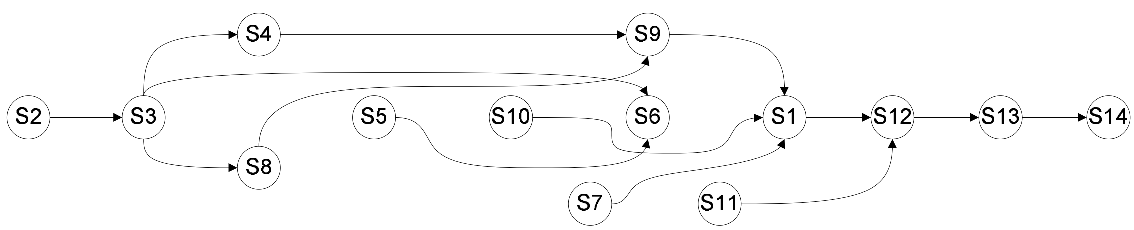
On the clausal level, the difference between the performance of ILP and BF is blurred. When evaluated on manually-constructed segments, ILP outperforms BF by less than 1%. This unexpected result can be explained by the skewed distribution of edge types — the two hardest edge types to classify (see Table 7.3), backward and null, account only for 7.4% of all edges at the clause level.

When evaluated on automatically segmented text, ILP performs slightly worse than BF. We hypothesize that this result can be explained by the difference between training and testing conditions for the pairwise classifier: the classifier is trained on manually-computed segments and is tested on automatically-computed ones, which negatively affects the accuracy on the test set. While all the strategies are negatively influenced by this discrepancy, ILP is particularly vulnerable as it relies on the score values for inference. In contrast, BF only considers the rank between the scores, which may be less affected by noise.

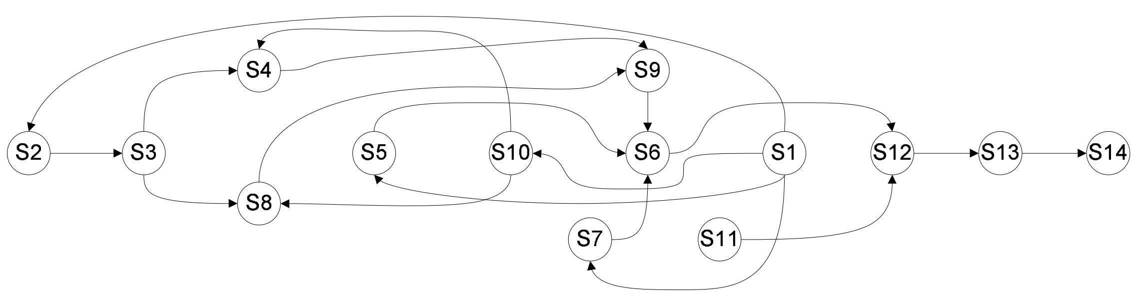
We advocate a two-stage approach for temporal analysis: we first identify segments and then order them. A simpler alternative is to directly perform a four-way classification at the clausal level using the union of features employed in our two-stage process. The accuracy of this approach, however, is low — it achieves only 74%, most likely due to the sparsity of clause-level representation for four-way classification. This result demonstrates the benefits of a coarse representation and a two-stage approach for temporal analysis.



(a) Reference TDAG



(b) ILP generated TDAG with an accuracy of 84.6%



(c) BF generated TDAG with an accuracy of 71.4%; NRO produces the same graph for this example.

Figure 7-1: Examples of automatically constructed TDAGs with the reference TDAG.

Algorithm	Manual Seg.	Automatic Seg.
ILP	**91.9**	84.8
BF	91.0	**85.0**
NRO	87.8	81.0
Baseline	73.6	73.6

Table 7.2: Results for 4-way ordering classifi cation over clauses, computed over manually and automatically generated segments.

Algorithm	Forward	Backward	Null
ILP	**92.5**	**45.6**	**76.0**
BF	91.4	42.2	74.7
NRO	87.7	43.6	66.4

Table 7.3: Per class accuracy for clause classifi cation over manually computed segments.

S1	A 32-year-old woman was admitted to the hospital because of left subcostal pain. . .
S2	The patient had been well until four years earlier,
S3	when she began to have progressive, constant left subcostal pain, with an intermittent increase in the temperature to 39.4°C, anorexia, and nausea. The episodes occurred approximately every six months and lasted for a week or two;
S4	they had recently begun to occur every four months.
S5	Three months before admission an evaluation elsewhere included an ultrasonographic examination, a computed tomographic (CT) scan of the abdomen. . .
S6	Because of worsening pain she came to this hospital.
S7	The patient was an unemployed child-care worker. She had a history of eczema and of asthma. . .
S8	She had lost 18 kg in weight during the preceding 18 months.
S9	Her only medications were an albuterol inhaler, which was used as needed,
S10	and an oral contraceptive, which she had taken during the month before admission.
S11	There was no history of jaundice, dark urine, light stools, intravenous drug abuse, hypertension, diabetes mellitus, tuberculosis, risk factors for infection with the human immunodefi ciency virus, or a change in bowel habits. She did not smoke and drank little alcohol.
S12	The temperature was 36.5°C, the pulse was 68, and the respirations were 16. . .
S13	On examination the patient was slim and appeared well. . . An abdominal examination revealed a soft systolic bruit. . . and a neurologic examination was normal. . .
S14	A diagnostic procedure was performed.

(d) An example of a case summary

Table 7.4: Text corresponding to Figure 7

Chapter 8

Conclusions

This paper introduces a new method for temporal ordering. The unit of our analysis is a temporal segment, a fragment of text that maintains temporal coherence. After investigating several inference strategies, we concluded that integer linear programming and best first greedy approach are valuable alternatives for TDAG construction.

In the future, we will explore a richer set of constraints on the topology on the ordering graph. We will build on the existing formal framework [11] for the verification of ordering consistency. We are also interested in expanding our framework for global inference to other temporal annotation schemes. Given a richer set of temporal relations, the benefits from global inference can be even more significant.

Chapter 9

Appendix

9.1 Temporal Segment Annotation —Guidelines for Annotating the NEJM Corpus

9.1.1 Introduction

When we communicate a narrative, we do not put a time stamp on every event. Rather, we explicitly communicate temporal information in some statements and use sentence structure and discourse structure to connect other statements to the statements that are more explicit.

Consider the following example from the NEJM corpus of cases:

> **A 12-year-old boy was evaluated in the hematology clinic of this hospital..** For one week, he had had fatigue that had caused him to miss school, and his appetite had decreased. He also had intermittent cramping, pain in his arms and legs, and anterior chest pain that was worse with movement, deep breathing, and coughing. (Case 19-2004)

As readers, we understand that the boy was evaluated, at some time in the past. Also, for a week before the clinic visit, the boy had the symptoms described in the latter lines of the excerpt. Though the statements about symptoms do have some temporal information (e.g. all past), they actually get their place in the order of events from the phrase "for one week".

The second and third sentences form a *temporal segment*: they all occur "at the same time". This doesn't mean that they all started together, ended together or must have progressed together. In fact, we could clarify the text (without contradicting it) with more text saying that the symptoms had occurred in some specific order. All we mean by calling the sentences a temporal segment is that the author of the text is not communicating to us information that would
the author grouped the events to share a temporal reference.

9.1.2 Annotation Task

The task is to define the segments by placing a mark at the beginning of each segment (which does not have to be after a period or even after a punctuation mark). There are some guidelines to keep in mind:

1. You will sometimes see *insertion*, where almost everything in a section of text is in the same temporal segment, except for a short bit referring to another time. You might also find nesting, where there is a big temporal segment with another temporal segment "inside" it (e.g. A segment that is "yesterday" with a segment referring to "four o'clock" inside it). If you see nesting or insertion, place a temporal segment marker at every boundary. Put temporal segment markers where you would otherwise nest or insert (if there were markers for nesting or inserting); this means that [a[b]] becomes [a][b]. Likewise, [a[b]a] becomes [a][b][c].

2. In general, resist the temptation to join statements into a single segment when the statements explicitly state an order: "Last week he grew tired, then became ill, then became unconscious" is three temporal segments, not one.

3. Place a boundary between events that do not have explicit words telling you their order, but absolutely must follow one another. (e.g. She went to another hospital, where she was told that her electrocardiogram was normal...)

4. Ignore verbs about how the events of the narrative are communicated. "The patient's mother noted that he felt warm" should be interpreted as only being about the warmth

of the patient. The time of the mother's report is not relevant.

5. Statements about the document, such as references to Tables, typically communicate something that was observed in the surrounding temporal segment. They also sometimes refer to the time when you (the reader) can see the document. Ignore the temporal information not relevant to the narrative; do not include references to the time of reading in your decision about whether or not to place a TCB marker. This is similar to the guideline above (4) because the fact that you *currently* see "the patient's heart-rate with respect to time sugar levels in Figure 4" has no relevance to the temporal segments in the narrative. Upon occasion, references to time-of-reading observations do not mention helpful words like "Table" or "Figure".

6. Ignore temporal statements regarding routine medical procedures that do not have bearing on the order of events in the narrative about the patient: "A CT scan of the thorax, obtained after the oral administration of contrast material, showed a new nodule." This should be one segment because we consider the temporal information to be part of the routine explanation of the parameters of the medical procedure.

7. When dealing with statements about the patient or the patients family that deal with past history (the patient was a retired nurse, the patient had a history of asthma, the patient's mother died of Tuberculosis) or general current state (the patient traveled frequently, the patient's siblings were all in good health) but there is no explicit time, mark history, recent history and general time-of-admission state as separate segments. Sequential references that are all the same type of state or history can be viewed as one segment.

8. The defi nition of temporal segment does not require a verb if there are explicit temporal markings. This leads us to mark the following. "The patient reported a four-month history of night sweats, #### a three-month history of 'generalized arthralgias,' #### a two-week history of headache, #### and recent loss of 4 or 5 kg in body weight". ("####" is our boundary marker.) Typically, however, words that share a verb are in the same temporal segment.

9. Typically, there is a long segment where all events are events that happen at the hospital. Unless there is clear order of precedence, consider sections where you cannot be sure of the order to be part of one segment.

10. Information about frequency of events often does not tell you about the boundary between temporal segments. “From then on, the patient experienced a high fever every 2 weeks...” is one segment.

11. These guidelines are by no means complete. Please footnote decisions you make that need a rationale beyond (or not in agreement with) these guidelines.

Bibliography

[1] James F. Allen. Towards a general theory of action and time. *Artificial Intelligence*, 23(2):123–154, 1984.

[2] Regina Barzilay, Noemie Elhadad, and Kathleen McKeown. Inferring strategies for sentence ordering in multidocument news summarization. *Journal of Artificial Intelligence Research*, 17:35–55, 2002.

[3] Yves Bestgen and Wietske Vonk. The role of temporal segmentation markers in discourse processing. *Discourse Processes*, 19:385–406, 1995.

[4] Branimir Boguraev and Rie Kubota Ando. Timeml-compliant text analysis for temporal reasoning. In *Proceedings of IJCAI*, pages 997–1003, 2005.

[5] Wallace Chafe. The flow of thought and the flow of language. In Talmy Givon, editor, *Syntax and Semantics: Discourse and Syntax*, volume 12, pages 159–182. Academic Press, 1979.

[6] Eugene Charniak. A maximum-entropy-inspired parser. In *Proceedings of the NAACL*, pages 132–139, 2000.

[7] William Cohen, Robert Schapire, and Yoram Singer. Learning to order things. *Journal of Artificial Intelligence*, 10:243–270, 1999.

[8] Carlo Combi and Yuval Shahar. Temporal reasoning and temporal data maintenance in medicine: Issues and challenges. *Computers in Biology and Medicine*, 27(5):353–368, 1997.

[9] Thomas H. Cormen, Charles E. Leiserson, and Ronald L. Rivest. *Intoduction to Algorithms*. The MIT Press, 1992.

[10] David R. Dowty. The effects of aspectual class on the temporal structure of discourse: Semantics or Pragmatics? *Linguistics and Philosophy*, 9:37–61, 1986.

[11] R. Fikes, J. Jenkins, and G. Frank. A system architecture and component library for hybrid reasoning. Technical report, Stanford University, 2003.

[12] Hebert P. Grice. Logic and conversation. *The logic of grammar*, pages 64—153, 1975.

[13] Marti Hearst. Multi-paragraph segmentation of expository text. In *Proceedings of the ACL*, pages 9–16, 1994.

[14] Chung Hee Hwang and Lenhart K. Schubert. Tense trees as the ”fi ne structure” of discourse. In *Proceedings of the ACL*, pages 232–240, 1992.

[15] Mirella Lapata and Alex Lascarides. Inferring sentence-internal temporal relations. In *Proceedings of HLT-NAACL*, pages 153–160, 2004.

[16] Alex Lascarides and Nicholas Asher. Temporal interpretation, discourse relations, and commonsense entailment. *Linguistics and Philosophy*, 16:437–493, 1993.

[17] Alex Lascarides and John Oberlander. Temporal connectives in a discourse context. In *Proceeding of the EACL*, pages 260–268, 1993.

[18] Inderjeet Mani, Barry Schiffman, and Jianping Zhang. Inferring temporal ordering of events in news. In *Proceeding of HLT-NAACL*, pages 55–57, 2003.

[19] Mark Moens and Mark J. Steedman. Temporal ontology in natural language. In *Proceedings of the ACL*, pages 1–7, 1987.

[20] Rebecca J. Passonneau. A computational model of the semantics of tense and aspect. *Computational Linguistics*, 14(2):44–60, 1988.

[21] James Pustejovsky, Patrick Hanks, Roser Sauri, Andrew See, David Day, Lissa Ferro, Robert Gaizauskas, Marcia Lazo, Andrea Setzer, and Beth Sundheim. The timebank corpus. *Corpus Linguistics*, pages 647–656, 2003.

[22] Hans Reichenbach. *Elements of Symbolic Logic*. Macmillan, New York, NY, 1947.

[23] Robert E. Schapire and Yoram Singer. Boostexter: A boosting-based system for text categorization. *Machine Learning*, 39(2/3):135–168, 2000.

[24] Bonnie L. Webber. The interpretation of tense in discourse. In *Proceedings of the ACL*, pages 147–154, 1987.

[25] Bonnie L. Webber. Tense as discourse anaphor. *Computational Linguistics*, 14(2):61–73, 1988.

[26] George Wilson, Inderjeet Mani, Beth Sundheim, and Lisa Ferro. A multilingual approach to annotating and extracting temporal information. In *Proceedings of the ACL 2001 Workshop on Temporal and Spatial Information Processing*, pages 81–87, 2001.

[27] Li Zhou, Carol Friedman, Simon Parsons, and George Hripcsak. System architecture for temporal information extraction, representation and reasoning in clinical narrative reports. In *Proceedings of AMIA*, pages 869–873, 2005.