Tense Interpretation in the Context of Narrative

Fei Song and Robin Cohen
Dept. of Computer Science, Univ. of Waterloo
Waterloo, Ontario, Canada N2L 3G1
{fsong,rcohen}@watdragon.uwaterloo.ca

Abstract

This paper presents an algorithm which makes use of tense interpretation to determine the intended temporal ordering between the states and events mentioned in a narrative. This is done by maintaining a temporal focus and interpreting the tense of each new statement of the narrative with respect to this focus. In particular, we propose heuristics for determining the temporal ordering and constraints for characterizing coherent tense sequences. The algorithm is further defended through experiments with naturally occurring examples.

Introduction

Tense interpretation plays an important role in determining the temporal ordering between the states and events mentioned in a narrative. Following Mourelatos [1978], we generally call states and events "situations." Determining temporal ordering is useful for many research problems of artificial intelligence. In story understanding, for example, knowing the temporal ordering allows us to answer questions like "what happened after a particular event occurred?"

Both tense and aspect are important expressions that contribute to the determination of temporal ordering ([Passonneau, 1987], [Dowty, 1986], [Comrie, 1985])[1]. This paper focuses on tense interpretation and adopts a simplified treatment of aspect as proposed in [Passonneau, 1987]. A more detailed treatment of aspect can be found in [Song, 1990].

Although tense has long been studied by linguists ([Reichenbach, 1947], [Dowty, 1986], [Comrie, 1985]), it is fairly recently that people have started to construct computational models to interpret it ([Hinriches, 1987], [Passonneau, 1987], [Webber, 1987], [Moens and Steedman, 1987]). Among these researchers, Webber [1987] is the first to extend tense interpretation from individual utterances to a whole discourse. Webber's main contributions include: recognizing the similarities between tense and other referential expressions such as pronoun and definite noun phrases, introducing the concept of temporal focus to maintain a dynamically changing entity, and presenting a set of heuristics on the possible movements of the temporal focus to interpret the tense of a new utterance. However, Webber allows all the heuristics to be applied in parallel and does not elaborate further on how the most plausible interpretation can be decided. Also, Webber did not consider the effects of aspect on the determination of the temporal ordering between situations.

In this paper, we extend Webber's work in several respects. First, we propose more detailed heuristics for determining the temporal ordering between situations. Second, we suggest constraints for capturing coherent tense sequences; only coherent sequences are further processed. Last, we arrange the heuristics and constraints in a fixed order to get a processing algorithm. The algorithm works for a restricted set of narratives which we call "simple narratives," but it can be made more general when more knowledge from discourse processing is added.

Representation Issues

We can use Vilain and Kautz's point algebra [Vilain and Kautz, 1986] to represent the temporal ordering between situations. We can also use the same algebra to describe the underlying structure of an English tense. Similar to Reichenbach's account [Reichenbach, 1947], we still use the three points: S (the speech time), R (the reference time, a theoretical entity used to describe some complex tenses and distinguish between certain tenses), and E (the event time). However, in addition to assuming "precedes" and "coincides", we also allow "follows", the inverse relation of "precedes", to be used to describe a tense structure. As a result, we can have the following list of SRE triples for describing English tenses[2]:

Simple Present	$[S = R = E]$
e.g., John runs.	
Simple Past	$[S > R = E]$

[1] Of course, other indicators such as temporal adverbials and connectives, discourse clues, and in general, real world knowledge of events also contribute to the analysis.

[2] Here, $<$, $=$, and $>$ stand for "precedes", "coincides" and "follows" respectively.

e.g., John ran.

Simple Future	[S < R = E]
John will run.	
Present Perfect	[S = R > E]
John has run.	
Past Perfect	[S > R > E]
John had run.	
Future Perfect	[S < R > E]
John will have run.	
Present Prospective	[S = R < E]
John is going to run.	
Past Prospective	[S > R < E]
John was going to run.	
Future Prospective	[S < R < E]
John will be going to run.	

The above list covers the same number of tenses as Reichenbach's account does, but it is both unambiguous and precise. For example, a Future Perfect would be described by three structures in Reichenbach's account, while by only one SRE triple in our description above. A situation, as argued in [Allen, 1983], typically holds or occurs over a time interval, which we denote as ET. It is not clear in Reichenbach's account how the E point is related to the ET interval. Our description is precise in that the relationships between E and ET are clearly specified according to the aspect of a situation.

In this paper, we follow [Passonneau, 1987] and treat aspect as the type of a situation. According to [Passonneau, 1987], situations can be classified into four types: states, temporally unbounded processes, temporally unspecified processes and transition events, based on the verb type and whether the tense is progressive. Then, the relationships between E and ET can be specified as follows,

state:	Start < E < End
ub-process:	Start < E < End
us-process:	Start $\leq$ E $\leq$ End
event:	Start < E = End

where Start and End denote the start and end points of the interval ET. Given "John is reading a book", for example, we can decide an ub-process, since "read" is an event-verb and its grammatical aspect is "progressive." As in [Passonneau, 1987], we can also decide that E should be marked as an interior point of the ET interval.

Tense Interpretation for Narratives

Temporal Focus Structure

Tense is widely regarded as anaphoric: its interpretation is usually linked to some time or situation derived from context. Webber [1987], following ([Bauerle, 1979], [Steedman, 1982], [Hinriches, 1986]), argues that it is the R point in a Reichenbach's tense description that is anaphoric. Webber suggests that one needs to maintain a dynamically changing entity as the temporal focus, denoted as TF, which is usually the E point of a previous utterance and is most likely to be used as the referent of the R point of the next utterance.

However, as the following example implies, treating only R as anaphoric is not enough, especially when the same tense is used to describe several situations.

a. John went to a hospital.
   (Sa > Ra = Ea)
b. He had fallen on a patch of ice
   (Sb > Rb > Eb)
c. and had twisted his ankle.
   (Sc > Rc > Ec)

Intuitively, we should be able to decide that Ea > Eb, Ea > Ec, and Eb < Ec. Following Webber's approach, we can decide Ea > Eb after the interpretation of utterance (b). Now, for the current TF, we can either maintain Ea as the TF or establish Eb as a new TF. If we take Ea as the referent for Rc, then we can only decide Ea > Ec, without knowing the relation between Eb and Ec. Alternatively, if we take Eb as the referent for Rc, then we can conclude Eb > Ec, a contradiction to Eb < Ec above. To get the right interpretation, we need to take Rb as the referent for Rc and a point after Eb as the referent for Ec. In other words, both R and E in a Reichenbach description should be treated as anaphoric.

After taking both R and E as anaphoric, we must consider how to decide the referents for them. Webber's one-point focus is not enough since R and E may not coincide for some tenses, and therefore, cannot refer to the same point. To get around this problem, we introduce the concept of temporal focus structure, denoted as TFS, to help interpret the R and E of a new situation. A TFS is also in the form of a SRE triple. It is different from a tense structure in that it is a variable — the values referred to by R and E can be changed from time to time[3]. In fact, TFS is an extension of Webber's one point TF: it not only contains the focus point for interpreting R, but also the point for interpreting E in a new utterance. A tense structure can be interpreted if it shares the same ordering relations between S, R, and E with a TFS. This is done by taking the values of R and E of the TFS (at a specific time, of course) as the referents for the R and E of the given tense structure.

As the above example indicates, tense can maintain an existing TFS, as is the case from (b) to (c) (similar to using a pronoun to maintain a noun in focus). Further, tense can create a new TFS based on an existing TFS, as is the case from (a) to (b) (similar to using a definite noun phrase to create a new focus). However, unlike the static objects referred to by pronoun

[3]The reason for including S in a TFS is that the speech time will shift forward for the on-line description of events, as illustrated in "John is making a phone call. Now, he has finished." In this paper, however, we assume that the difference between the S's is negligable, since in a simple narrative most of the events occur either in the past or future.

or definite noun phrases, the time referred to by tense is a dynamic entity, which typically shifts forward, as is the case from (b) to (c).

Detailed Heuristics for Tense Interpretation

In order to describe the heuristics for tense interpretation, we organize all nine SRE triples into the tense hierarchy in figure 1. Here, a thin link from a father to its

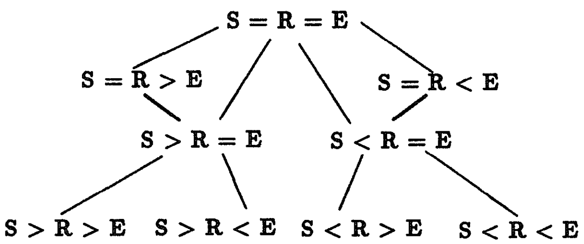

Figure 1: Tense Hierarchy in English

son denotes a creation case, where the father is taken as the current TFS and the son as the tense structure of a new utterance, and a thick link or the repetition of the same structure suggests a maintenance case.

In the following, we assume that from prior processing, we have (1) set the current TFS, and (2) determined ST(n) and TS(n), the situation type and the tense structure of the current utterance. Also, we use S(n), R(n), and E(n) to denote all the points in TS(n), and S(f), R(f), and E(f) all the points in the current TFS.

There are two rules for the maintenance case: (1) progression rule, applicable when the same tense is used to described several situations; (2) elaboration rule, applicable when the tense sequence is from a Present Perfect to a Simple Past or from a Present Prospective to a Simple Future, marked by the two thick links in our tense hierarchy.

```
procedure maintain(TS(n), TFS)
begin
  if TS(n) = TFS then     /* progression rule */
    if ST(n) is a state or ub-process
    then record E(n) = E(f)
    else record E(n) > E(f);
    if R(n) ≠ E(n) then set R(n) = R(f);
    if there exists m such that m ≠ n and
      (E(f) < E(m) or E(f) ≤ E(m)) then
      if E(f) ≠ E(m) then
        replace E(f) < E(m) with
          E(n) < E(m)
      else replace E(f) ≤ E(m) with
          E(n) ≤ E(m);
  else   /* elaboration rule */
    record E(f) ≥ E(n)
end
```

The progression rule captures the forward shifts of time from situation to situation, depending on the type of a situation, as time typically shifts for events and bounded processes, but stays the same for states and unbounded processes ([Dowty, 1986], [Hinriches, 1986]). Also in the progression rule, we check for a prior situation E(m) such that E(f) occurs earlier than E(m), and if such a situation exists, we replace the ordering relation with E(n) located earlier than E(m). This step is intended to update the global representation of the ordering relations in the narrative, by collapsing certain binary relations. In contrast, the elaboration rule shifts time backwards in order to add details to a previously introduced situation. For example, it is often the case that a speaker uses a Present Perfect to introduce an event in the past and then uses several Simple Pasts to elaborate the event in detail.

There are also two rules for the creation case: R-creation and E-creation. The former can be applied to the sequence from a Simple Present to a Simple Past or a Simple Future, and the latter to the other thin links in our tense hierarchy, i.e., sequences from a Simple tense to a Perfect tense or a Prospective tense.

```
procedure create(TS(n), TFS)
begin
  if R(n) = E(n) then    /* R-creation rule */
    if S(n) < R(n) then record E(f) < E(n)
    else record E(f) > E(n)
  else   /* E-creation rule */
    set R(n) = E(f)
    if R(n) < E(n) then record E(f) < E(n)
    else record E(f) > E(n)
end
```

Constraints on Coherent Tense Sequences

In the previous subsection, we assumed that the current TFS was given for interpreting the tense structure of a new utterance. Now, we need to consider how to set and maintain the current TFS, in particular, what to use as the initial TFS and how to save the old TFS for later resumption every time a new TFS is created.

Since from the current TFS, we can either maintain the TFS or create a new TFS based on the TFS, it is natural to take the tense structure [S=R=E] at the root of our tense hierarchy as the initial TFS. Another reason is that all points in this structure refer to the speech time which is obvious to both the speaker and the hearer. In [Comrie, 1985], the speech time is also called the deictic center, since the speaker can always use clue words like "now", "at present", to direct the hearer's attention to this time. Then, starting from this initial TFS, we can either maintain the TFS or create one of the four new structures: [S=R>E], [S>R=E], [S=R<E], and [S<R=E].

However, there are cases where a narrative starts with a Past Perfect. Such a Past Perfect is often used to set up the background in the past and from then on, more past situations can be given to make up the narrative. That is, the deictic center is actually moved to a time in the past and we can take [S>R=E], the structure of a Simple Past, as the initial TFS.

Once setting up the initial TFS, we can then maintain or create a new TFS based on the TFS. However, at some point of the narrative, the speaker may need to return back to a previously introduced TFS. Following ([Grosz and Sidner, 1986], [Webber, 1988]), we can use a focusing stack to store all the existing TFS's, with the top element always being the current TFS. When a new TFS is created, it will be pushed on top of the stack so that it becomes the current TFS. When a previous TFS is resumed, all the elements above it will be eliminated so that the previous TFS becomes the current TFS again.

Referring to our tense hierarchy, maintenance and creation correspond to the repetition of the same node or the links from a father to its son, while resumption corresponds to the links from a son to its father. In other words, the links in the hierarchy can be seen as bidirectional. However, our heuristics for tense interpretation only apply to the links that go from a father to its son. For example, a switch from a Simple Past to a Simple Present requires us to first resume a previous Present TFS in the focusing stack and then apply the heuristic for the maintenance case to interpret the Present tense.

Using a stack to manage the change of TFS is similar to the management of some other kinds of focuses in discourse processing ([Grosz, 1977], [Sidner, 1983], [McKeown, 1985]). The reason that we prefer the most recent TFS is that a speaker can only create a new TFS based on an existing TFS. Once a TFS is created, the speaker tends to make full use of it before returning to a previous TFS, otherwise the TFS has to be reintroduced into the focusing stack.

The above rules of setting up the initial TFS and managing the existing TFS's form the constraints on coherent tense sequences. Tense sequences that do not satisfy these constraints are said to be incoherent, i.e., where there are no possible links between some tense structure and the existing TFS's. Consider the following example,

a. John is staying at home.
   (Sa = Ra = Ea)
b. He had finished his homework.
   (Sb > Rb > Eb)

After interpreting utterance (a), the current TFS will have the structure [Sa=Ra=Ea]. Now, given utterance (b), we cannot maintain the current TFS since it does not match the tense structure of utterance (b), nor can we create a new TFS to interpret utterance (b) as a Past Perfect is not a son structure of a Simple Present. Therefore, we decide that the given tense sequence is incoherent.

An Algorithm for Tense Interpretation

Based on the detailed heuristics for tense interpretation and the constraints on coherent tense sequences, we can now present a context-based algorithm for tense interpretation. It will terminate since all the rules are arranged in a fixed order and it stops only when all the utterances in a narrative are processed or the tense sequence of the narrative is incoherent.

```
input a list of (n, ST(n), TS(n)), where n is the
   order, and ST(n) and TS(n) are the situation type
   and tense structure of a new situation;
output a network of E points and the ordering
   relations between them;
begin
   if TS(n) = [S(1)>R(1)>E(1)] then
     push [S(0)>R(0)=E(0)] to the focusing stack
   else
     push [S(0)=R(0)=E(0)] to the focusing stack;

   while Input is not empty do
     begin
        get the next TS(n) from Input;
        search through from top of the focusing
          stack for a TFS such that TS(n) = TFS
          or TS(n) = a son of TFS;
        if no such TFS exists then
          report incoherent discourse and stop;
        eliminate all the elements above TFS in
          the stack;
        if TS(n) = TFS or (S(f)=R(f) and
          R(f) ≠ E(f)) then
          call maintain(TF(n), TFS);
          update TFS with TS(n);
        else
          call create(TS(n), TFS);
          push TS(n) onto the focusing stack;
     end
end
```

In order to save space, we choose a small example to illustrate the above algorithm.

(1) John is boiling the fettucini noodles.
(2) He has already made the marinara sauce.
(3) He is going to put them together
    to get a pasta dish.

The corresponding input list can be given as follows:

[(1, [S(1)=R(1)=E(1)], ub-process),
 (2, [S(2)=R(2)>E(2)], event),
 (3, [S(3)=R(3)<E(3)], event)]

At the beginning, we initialize the current TFS to be [S(0)=R(0)=E(0)] since the first utterance is not described in a Past Perfect. Taking the first utterance, we find that its tense structure [S(1)=R(1)=E(1)] matches the current TFS. Following the "maintain" procedure, we record E(1) = E(0) since the given utterance describes a temporally unbounded process. After this interpretation, we update the current TFS with

[S(1)=R(1)=E(1)]. Taking the second utterance, we find that its tense structure [S(2)=R(2)>E(2)] is a son structure of the current TFS in our tense hierarchy. So we call the "create" procedure to record E(2) < E(1) and push [S(2)=R(2)>E(2)] on top of the focusing stack to get a new TFS. Here, the creation is a case of E-creation. Taking the last utterance, we find that its tense structure [S(3)=R(3)<E(3)] does not match the current TFS; nor can the structure be created from the current TFS, as there is no creation link between them in our tense hierarchy. However, this tense structure can be created from a previous TFS, the one that is obtained from the first utterance. So we eliminate the current TFS in the stack and resume [S(1)=R(1)=E(1)] below it to be the current TFS. Then, we call the "create" procedure to record E(3) > E(1) and push [S(3)=R(3)<E(3)] on top of the stack. Since all of the utterances have been interpreted, our algorithm will now terminate and give us the temporal structure shown in figure 2. Note that E(0) is used as a dummy situation, which is only useful for setting up the initial TFS and is not shown in the figure.

$$\text{———} \; E(2) \xrightarrow{\;<\;} E(1) \xrightarrow{\;<\;} E(3) \longrightarrow$$

Figure 2: Temporal Structure of Example 1

Experiments with Natural Examples

Our rules for tense interpretation are intended to capture the most likely cases. Exceptions to these cases do arise in contexts where other temporal indicators, such as temporal adverbials and connectives, and discourse cue phrases (see [Grosz and Sidner, 1986]) are provided.

To further test our algorithm, we chose a total of twenty examples from the book *Real Stories* [Katz *et al.*, 1975]. Our experiments may go through two possible rounds. First, we test whether our algorithm can produce the expected temporal ordering that would be decided by a human reader.

Our algorithm uses heuristics to prefer certain interpretations, in the absence of other temporal indicators. For example, in the case when the same tense is repeated in a subsequent sentence, our rule would prefer progression over elaboration as a speaker tends to describe situations at the same level of detail and when the speaker wants to add details to some situation, he usually uses cue phrases such as "first" and "for example" to clearly indicate such a shift[4].

[4] Similar heuristics are also used in [Cohen, 1983], [Litman, 1985], and [Carberry, 1986]. The general rule seems to be that we prefer continuation over resumption and prefer resumption over creation.

For examples with interpretations that are inconsistent with the results of round one, we run a second round, allowing a user to provide information suggested by other temporal indicators. If there are such linguistic expressions available, the user provides the focus movement suggested; otherwise, our algorithm simply assumes the heuristics used at the first round analysis. Depending on how many focus movements are explicitly provided by a user, we compute the number of utterances that are correctly interpreted by our tense algorithm.

The number of utterances that are interpreted correctly is shown in the table below. An example that shows the natural flavor of our experiments can be found in Appendix A.

#Stories	#Utterances		
	Average	Tense&Aspect	With User
20	16.15	14.15	16.0

As our results suggest, further work should extend our current algorithm to interact with a module which has additional discourse information.

Summary and Future Directions

We have presented an algorithm that uses tense interpretation in the analysis of simple narratives. Our work can be seen as an extension to Webber's. More specifically, we proposed detailed heuristics for interpreting tense, suggested constraints for capturing coherent tense sequences, and organized these rules into an algorithm for determining the temporal ordering between the situations mentioned in a narrative.

One of our future directions is to provide a detailed treatment of aspect. Readers are referred to [Song, 1990] for more discussion. In addition, temporal adverbials (e.g., yesterday, at three o'clock, in a week) and temporal connectives (e.g., when, before, after, while) are also effective ways of describing ordering relations. The problem with these expressions is that they are not always available and are widely diversified. They may also require a mechanism for combining quantitative temporal information (often incomplete) with qualitative temporal information (usually uncertain) (see [Allen, 1983]).

In short, we believe that our work provides the basis for building more complex algorithms to implement these possible extensions.

References

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

Bauerle, R. 1979. *Temporal Deixis, Temporal Frage.* Gunter Narr Verlag, Tubigen.

Carberry, Sandra 1986. *Pragmatic Modeling in Information System Interfaces.* Ph.D. Dissertation, University of Deleware.

Cohen, Robin 1983. *A Computational Model for the Analysis of Arguments.* Ph.D. Dissertation, University of Toronto.

Comrie, Bernard 1985. *Tense.* Cambridge University Press.

Dowty, David 1986. The effects of aspectual class on the temporal structure of discourse: Semantics or pragmatics. *Linguistics and Philosophy* 9(1):37–62.

Grosz, Barbara J. and Sidner, Candace L. 1986. Attention, intentions, and the structure of discourse. *Computational Linguistics* 12(3):175–204.

Grosz, Barbara J. 1977. The representation and use of focus in a system for understanding dialogs. In *Proceedings of IJCAI Conference.* 67–76.

Hinriches, Erhard W. 1986. Temporal anaphora in discourses of English. *Linguistics and Philosophy* 9:63–82.

Hinriches, Erhard W. 1987. A compositional semantics of temporal expressions in English. In *Proceedings of the 25th ACL Conference.* 8–15.

Katz, Milton; Chakeres, Michael; and Bromberg, Murray 1975. *Real Stories.* Globe Book Company, New York. Book One, Second Edition.

Litman, Diane 1985. *Plan Recognition and Discourse Analysis: An Integrated Approach for Understanding Dialogues.* Ph.D. Dissertation, University of Rochester.

McKeown, Kathleen R. 1985. *Text Generation: Using Discourse Strategies and Focus Constraints to Generate Natural Language Text.* Cambridge University Press.

Moens, Marc and Steedman, Mark 1987. Temporal ontology in natural language. In *Proceedings of the 25th ACL Conference.* 1–7.

Mourelatos, Alexander P. D. 1978. Events, processes, and states. *Linguistics and Philosophy* 2:415–434.

Passonneau, Rebecca J. 1987. Situations and intervals. In *Proceedings of the 25th ACL Conference.* 16–24.

Reichenbach, Hans 1947. *The Elements of Symbolic Logic.* The Free Press.

Sidner, Candace L. 1983. Focusing in the comprehension of definite anaphora. In Brady, M. and Berwick, R. C., editors 1983, *Computational Models of Discourse.* The MIT Press. 267–330.

Song, Fei 1990. *A Processing Model for Temporal Analysis and its Application to Plan Recognition.* Ph.D. Dissertation, University of Waterloo.

Steedman, Mark 1982. Reference to past time. In Jarvella, R. and Klein, W., editors 1982, *Speech, Place, and Action.* Wiley. 125–157.

Vilain, M. and Kautz, H. 1986. Constraint propagation algorithms for temporal reasoning. In *Proceedings of the Fifth National Conference on Artificial Intelligence.* 377–382.

Webber, Bonnie L. 1987. The interpretation of tense in discourse. In *Proceedings of the 25th ACL Conference.* 147–154.

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

A. One Real Example

The examples used in our experiments are all adopted from *Real Stories* [Katz *et al.*, 1975]. Many of these examples may need to be transcribed so that certain linguistic constructions (such as indirect speech and non-actual situations) are stripped off or restated for the purpose of tense interpretation. One transcribed example is shown as follows:

(1) King is a watchdog at an Air Force base in Tennessee.
(state, simple present)
(2) At one point, King was about to be destroyed.
(state, simple past)
(3) He was too mean to train.
(state, simple past)
(4) He was vicious.
(state, simple past)
(5) He hated everybody, everything that moves and everything touching him.
(state, simple past)

(6) King had been raised by a Spanish-speaking family
(event, past perfect)
(7) before he was sold to the Air Force.
(event, simple past)
(8) All that King wanted was someone to give him his orders in Spanish.
(state, simple past)
(9) Spanish was the only language he knew.
(state, simple past)

(10) King was given a new trainer who speaks Spanish.
(event, simple past)
(11) Now King is happy and
(state, simple present)
(12) the Air Force is happy with him.
(state, simple present)

Here, each utterance is associated with a description indicating the type and the tense of the situation described. Also, an empty line between utterances are used to separate different paragraphs.

Our tense interpretation algorithm can interpret correctly 10 out of the 12 utterances. The two exceptions are utterance (7), where a progression case is indicated by the connective "before", and utterance (10), where a resumption case is suggested by the start of a new paragraph, which can be seen as a clue to discourse structures.