A Heuristic Approach to Order Events in Narrative Texts

Farid Nouioua
LIPN, UMR 7030 du C.N.R.S. – Institut Galilée
Université Paris 13, 99 avenue J.B.Clément, F 93430, Villetaneuse
Farid.Nouioua@lipn.univ-paris13.fr

Abstract

We describe a heuristic method to order the events evoked in texts describing road accidents. The method consists first in generating, by non monotonic inference rules, a set of precedence and simultaneity constraints between temporal references linked to particular words of the text. Then, the proposed algorithm updates the narration order of the events to satisfy these constraints and give the real order of the events described in this text. The temporal information is represented according to a linear temporal model which is enhanced in a first order reified language.

1. Introduction

Knowledge about time plays an important role in Natural Language Understanding (NLU) and in Artificial Intelligence (AI). The study of temporality in NL texts consists generally in identifying the temporal nature of the different events evoked in a text as well as the relations linking them. For French, [2] and [3] describe a method that computes the aspecto-temporal values of the different parts of a text by determining aspect and temporal operators as well as modalities and then applying them to the propositional content of the text to obtain processes. Another important work for French is the Sdt model (semantic of temporality) [5] which proposes four types of intervals: an enunciation interval for each sentence, a process interval, a reference interval and an interval representing the portion of the temporal axis delimited by each time complement or time subordinate. An implementation of this model can be found in [10]. These approaches are not dedicated to trigger reasoning about texts.

In AI, several models of time have been proposed to reason about actions and change. The Allen's generalized interval theory [1] is a well known example. It proposes a naïve interval-based temporal reasoning framework. This model defines 13 explicit topological relations between intervals and studies the constraints between them and their propagation. Another model to represent time for reasoning about actions is McDermott's chronicles [7]. It uses a set of axioms which define its structure and properties. It considers an infinite set of partially ordered states; each one has a date which is a real number. A chronicle is a convex set of totally ordered states whose dates match the real axis. Chronicles can branch in future, which allows representing the free will. This model has been applied to planning and causality.

This paper proposes a temporal model for causal reasoning. From a text describing an accident, we want to construct automatically a semantic representation which conveys enough features to trigger a causal reasoning to detect the cause of this accident.

This work is a part of a larger architecture of a complete NLU system which detects the cause of a road accident from its textual description [8]. In this system, a parser[1] followed by some post-treatments extracts from the text the "linguistic literals": syntactic relations between the words of the text[2]. The linguistic literals trigger a first non monotonic reasoning process: the "linguistic reasoning". Its goal is to construct the "semantic literals" that are first-order literals able to express the events that really happened according to the author. The semantic literals feed in turn a second non monotonic causal reasoning which infers the cause of the accident [6]. The linguistic reasoning translates a representation using NL words into another one using "concepts" including time. The temporal reasoning is then a part of the linguistic reasoning on which this paper is focused.

Section 2 is devoted to the logical language used to represent the semantic content of a text. Section 3 discusses the reasoning used to extract temporal constraints and the algorithm which uses them to order the events of a text. The implementation and the experimental results are presented in section 5. Section 6 concludes and draws directions for future work.

[1] Developed by Daniel Kayser for the texts of our corpus.

[2] The relations used are: subject/verb, object/verb, verb/complement, noun/complement, verb/qualification, noun/qualification and relations between couples of verbs.

1530-1311/08 $25.00 © 2008 IEEE
DOI 10.1109/TIME.2008.13

IEEE computer society

2. The target representation language

2.1. A first order reified logic

The reification technique allows quantifying over predicates while staying within a first order framework. For example, instead of writing *P(X,Y)*, we write *Holds(P, X, Y)* in which *P* becomes an argument (henceforth, we denote by "predicate" the predicate *Holds* of the reified language and by "properties" the predicates of the original). Since the predicate *Holds* must have a fixed arity (3 in this work) whereas the reified properties may have different arities, we use a specific binary function *combine* to construct new arguments from simpler ones. For example, the reification of the predicate *follows(A, B, T)* (*A* follows *B* at time *T*) is : *Holds(combine(follows, B), A, T)*.

2.2. The temporal model

There are three times in our texts: the linear time of narration, the linear time of events (the real course), and the branching time which takes into account also the unrealized possible futures. The last model is undoubtedly the richest one but we found, for our particular application, that only the second time needs an explicit representation. This information together with the effective course of events allows the causal reasoning to detect violations of norms which represent the potential causes of the accident [8]. Therefore, we use integers to represent temporal states. The general form of a semantic literal is: *[¬]holds(P, A, T)* where *P* is a property, *A* is an agent or an object and *T* is the index of a temporal state. Another hypothesis of our work is that, as far as causal reasoning is concerned, it suffices to consider only the precedence and simultaneity relations that link the events of the text.

Let us now explain what states are. The scene described in a text is seen as sequence of successive pictures taken from a continuous situation. A temporal state corresponds to an interval characterized by its *"persistent effects"* : the set of propositions that keep the same truth value during it. A transition between two states *t* and *t+1* is performed by two ways :

- An agent executes successfully, in state *t*, an action *Act* which allows to transit to state *t'= t+1* by changing the truth value of at least one proposition. For example, if an agent *A* brakes at state *t* then, in state *t'=t+1*, A's speed decreases or *A* stops.
- An external event *evt* occurs and causes the transition from a state *t* to a state *t'=t+1* in which its effects hold. For example, meeting a glaze plaque in the road may cause transition from state *t* where a vehicle is under control to state *t'=t+1* where it is no more under control.

An action *act* or an event *evt* occur in a transition between two states *t* and *t+1*. However, to homogenize the representation, we associate to an action or an event only the index of the first state. Hence, a state *t* is characterized, in addition to persistent effects, by actions and/or events occurring in the transition *t, t+1*.

For example, consider the following text (B52) : « *Après m_1'être arrêté au_1 feu_1 rouge, je suis reparti au_2 feu_2 vert et c'est à ce moment que le véhicule B est venu me_2 percuter fortement $à_3$ l'arrière*[3]» *(after being stopped at red light, I moved off at the green light and that's when the vehicle B struck me heavily at the rear*). We distinguish in this text four states: in state 1, the author is stopped at the red light : *holds(stops, author,* ***1****), holds(red_light, author,* ***1****);* in state 2, the light turns green: *holds(green_light, author,* ***2****);* in state 3 the author moves off: *holds(moves_off, author,* ***3****)* and in state 4, vehicle B strikes the author's vehicle at the rear: *holds(combine(strikes, author), veh_B,* ***4****), holds(combine(shock_position, rear), author,* ***4****).*

3. The method

We use Reiter's default logic [11] to write the inference rules. We have material implications of the form: $A_1 \wedge ... \wedge A_n \rightarrow B$; normal defaults of the form: $\frac{A_1 \wedge ... \wedge A_n : B}{B}$, *(denoted :* $A_1 \wedge ... \wedge A_n : B$*) and* semi-normal defaults of the form : $\frac{A_1 \wedge ... \wedge A_n : B[C_1, ..., C_m]}{B}$ (denoted : $A_1 \wedge ... \wedge A_n : B\ [C_1, ..., C_m]$) where $A_1, ..., A_n$, B, C_1, ..., C_k are first order literals.

3.1. Inferring the atemporal semantic literals

We first infer "intermediate semantic literals" which come close in form to final semantic literals but which still handle words and not concepts. The input of this task are *subject/verb*, *object/verb* and *verb/complement* relations that are indeed at the root of most semantic literals. The form of intermediate semantic predicates is: *Holds_I(P_Prop_P, P_Ag, t_ref(M))*. *Prop_P* and *Ag_P* indicate respectively "potential property" and "potential agent". *Temp_ ref(M)* denotes: "a temporal reference associated to word *M*". It will potentially be used to compute the effective temporal parameter of a semantic predicate. Here are two examples of the intermediate semantic literals extracted from text B52 :

holds_I(arrêter, auteur, t_ref(arrêter)),
holds_I(combine(percuter,auteur),veh_B,t_ref(percuter))

[3] Indices distinguish different occurrences of a same word.

The following step is to infer the "atemporal semantic literals" that are identical to final semantic literals, except for the temporal parameter which remains unresolved (for more details about the inference rules of this level, see [8]). The atemporal semantic literals extracted from the text B52 are:

$$\begin{gathered} holds(stops,\ author,\ t\_ref(arreter)),\ holds(red\_light, \\ author,\ t\_ref(a_1)),\ \ holds(green\_light,\ author, \\ t\_ref(a_2)),\ holds(moves\_off,\ author,\ t\_ref(repartir)), \\ holds(combine(strikes,\ author),\ veh\_B, \\ t\_ref(percuter)),\ holds(combine(shock\_\ position,\ rear), \\ author,\ t\_ref(a_3)) \end{gathered}$$

3.3. Inference of temporal constraints

While inferring the atemporal semantic literals, a temporal reasoning generates constraints of the form: the temporal reference associated to a word M_1 must precede directly (resp. be simultaneous with) that associated to word M_2 : $prec(t\_ref(M_1),\ t\_ref(M_2))$ (resp. $simul(t\_ref(M_1),\ t\_ref(M_2))$). The default order of events given in narration must then be updated to satisfy these constraints. The temporal inference rules are based on simple interpretation of the temporal significance of some words and syntactical constructions in addition to general common sense knowledge. Here are examples of these inference rules:

Rule (1) treats expressions describing situations where an agent is stopped at a stopping cause (red light, stop panel, …). It infers the simultaneity of the stopping state and the presence of the stopping cause.

$$\begin{gathered} type(A,\ vehicle) \wedge type(X,\ stopping\_cause) \\ \wedge prep(Pr) \wedge sem\_rep(V,\ arreter) \wedge holds\_I(V,\ A,\ H) \\ \wedge holds\_I(combine(combine(Pr,V),X),\ A,t\_ref(Pr)) \\ \rightarrow simul(H,\ t\_ref(Pr)) \qquad (1) \end{gathered}$$ [4]

We use the rule (2) to extract from expressions telling that a vehicle receives a shock in some position (rear, front, …) the simultaneity relation between the shock event and the position of this shock.

$$\begin{gathered} Type(A,\ vehicle) \wedge type(B,\ vehicle) \wedge type(X, \\ position) \wedge prep(Pr) \wedge sem\_rep(V,\ heurter) \wedge \\ holds\_I(combine(V,\ A),\ B,\ t\_ref(V)) \wedge \\ holds\_I(combine(combine(Pr,V),X),B,t\_ref(Pr)) \wedge \\ \rightarrow simul(H,\ t\_ref(Pr)) \qquad (2) \end{gathered}$$

We use the rule (3) to express that a light turning green precedes the moving off of the driver.

$$\begin{gathered} Type(A,\ vehicle) \wedge prep(Pr) \wedge sem\_rep(V,\ demarrer) \\ \wedge sem\_rep(X,\ feu\_vert) \wedge holds\_I(V,\ A,\ H) \wedge \\ holds\_I(combine(combine(Pr,V),X),A,t\_ref(Pr)) \\ \rightarrow prec(H,\ t\_ref(Pr)) \qquad (3) \end{gathered}$$

The rule (4) is used to infer, if a vehicle moves off at the green light after being stopped at the red light, that necessarily, the passage of the light to green happens after the state when it is red.

$$\begin{gathered} type(A,vehicle) \wedge prep(Pr_1) \wedge prep(Pr_2) \\ \wedge sem\_rep(V,arreter) \wedge sem\_rep(W,demarrer) \wedge \\ sem\_rep(X,feu\_rouge) \wedge sem\_rep(Y,feu\_vert) \wedge \\ sem\_rep(Z,apres) \wedge holds\_I(combine(combine(Pr_1,V), \\ X),\ A,\ t\_ref(Pr_1)) \wedge holds\_I(combine(combine(Pr_2, \\ W),Y),A,t\_ref(Pr_2)) \wedge relation(Z,V,W) \rightarrow \\ prec(t\_ref(Pr_1),\ t\_ref(Pr_2)) \qquad (4) \end{gathered}$$

By applying the above inference rules on our example, we obtain the following constraints:

$$\begin{gathered} simul(t\_ref(arreter),t\_ref(a_1)), \\ simul(t\_ref(percuter),t\_ref(a_3)), \\ prec(t\_ref(a_2),t\_ref(repartir)),\ prec(t\_ref(a_1),\ t\_ref(a_2)) \end{gathered}$$

Other rules are used to infer other temporal constraints in different other situations. Some of them still rather specific to the application domain, but some others use more general linguistic or common sense knowledge. For example we have rules that reason about grammatical tenses of verbs[5] or use causal relations to state that a cause always precedes its effect.

3.4. Resolving the temporal constraints

An algorithm receives the set of the temporal references $t\_ref(M_1),\ ...,\ t\_ref(M_n)$ present in the atemporal semantic literals ($M_1,\ ...,\ M_n$ appear in the text in this order) and the set of the temporal constraints $R_1,\ ...,\ R_m$ and associates to each reference $t\_ref(M_i)$ an integer $T_i > 0$ representing its real order. $S = T_1,\ ...,\ T_n$ is called "the temporal structure". Here is the general schema of the algorithm:

- *Initialization:* Initialize the order T_i of each reference $t\_ref(M_i)$ by i (the narration order)
- Let $R_1,\ ...,\ R_m$ be the set of the constraints taken in an arbitrary order[6]. For each R_j, if S does not satisfy R_j (see below), then modify S to satisfy R_j.
- If all constraints are satisfied, then replace each $t\_ref(M_i)$ by T_i else repeat the previous step.

[4] The predicate *type* regroups together words belonging to a same semantic type and the predicate *sem_rep* gives a same representative word to all words considered as synonyms.

[5] One of the rules of this type which is commonly used in our texts, states that if two verbs V_1 and V_2 have the same subject S and that V_1 is at the "imparfait" tense whereas V_2 is in the "passé composé" tense, then generally, the temporal reference associated to V_1 precedes that associated to V_2.

[6] In our application, we have often a limited number of temporal relations. Considering them in any order gives satisfactory results.

A constraint *prec(t_ref(M_j), t_ref(M_i))* is satisfied iff: $T_j = T_i\text{-}1$ (immediate precedence). If the temporal structure *S* does not satisfy *R*, we modify it by "moving" *t_ref(M_j)* to be just before *t_ref(M_i)*, filling the gap left by this move if any and ensuring that the numbering starts always with *1*. The possible cases are treated by the following algorithm:

```
if ($T_i \le T_j$)
  if ($t\_ref(M_j)$ is not alone[7] and $T_i \neq 1$)
     $T_j \leftarrow T_i - 1$ ; (change the order of $t\_ref(M_j)$);
  if ($t\_ref(M_j)$ is alone and $T_i \neq 1$)
     For (each $T_k > T_j$)    $T_k \leftarrow T_k - 1$ ;  (move the
       references situated after $t\_ref(M_j)$ to fill the gap)
     $T_j \leftarrow T_i - 1$ ; (change the order of $t\_ref(M_j)$)
  if ($t\_ref(M_j)$ is not alone and $T_i = 1$),
     For (each $T_k$)    $T_k \leftarrow T_k + 1$ ; (keep the first place
       for $t\_ref(M_j)$)
     $T_j \leftarrow 1$ ;
  if ($t\_ref(Mj)$ is alone  and $Ti = 1$)
     For (each $Tk \ge 1$ and $T_k < T_j$)
        $T_k \leftarrow T_{k+1}$ ; (keep the first place for $t\_ref(M_j)$
        and fill the gap left by $t\_ref(Mj)$)
     $Tj \leftarrow 1$ ;
if ($T_i > T_j$)
  if ($t\_ref(M_j)$ is not alone)
     $T_j \leftarrow T_i - 1$ ; (change the order of $t\_ref(M_j)$)
  if ($t\_ref(M_j)$ is alone)
     For (each $T_k > T_j$)   $T_k \leftarrow T_k - 1$ ; (move the
  references situated after $t\_ref(M_j)$ to fill the gap)
     $T_j \leftarrow T_i - 1$ ; (change the order of $t\_ref(M_j)$)
```

We treat analogously the simultaneity constraints: a relation *simul(t_ref(M_j), t_ref(M_i))* ($T_i \neq T_j$) is satisfied by "moving" the reference having the larger number. The following algorithm treats the possible cases. It considers without loss of generality that $T_i < T_j$.

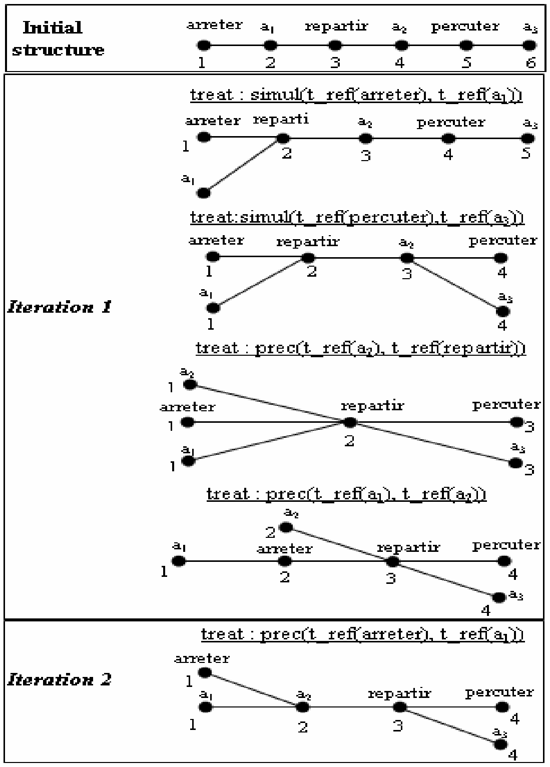
```
if ($t\_ref(M_j)$ is not alone)
  $T_j \leftarrow T_i$  (change the order of $t\_ref(M_j)$)
if ($t\_ref(M_j)$ is alone)
  For (each $T_k > T_j$)
     $T_k \leftarrow T_k - 1$ (move the references situated after
     $t\_ref(M_j)$ to fill the gap)
  $T_j \leftarrow T_i$  (change the order of $t\_ref(M_j)$)
```

3.5. An example

Let us recall that the set of temporal constraints inferred from our example (text B52) are :

simul(t_ref(arreter), t_ref(a_1)), simul(t_ref(percuter), t_ref(a_3)), prec(t_ref(a_2), t_ref(repartir)), prec(t_ref(a_1), t_ref(a_2))

The unfolding of the algorithm on this example is illustrated as follows:



All the other constraints are satisfied by this last structure. Therefore, the list of the "final" semantic literals obtained from text B52 are:

holds(stops, author, 1), holds(red_light, author, 1), holds(green_light, author, 2), holds(moves_off, author, 3), holds(combine(strikes, author), veh_B, 4), holds(combine(shock_position, rear), author, 4)

4. Implementation and results

To implement the system, we used the Answer Set Programming (ASP) paradigm [4]. We chose for that the Smodels language based on stable models semantics (available at: http://www.tcs.hut.fi/Software/smodels/). Our inference rules are translated into extended logic programs [9]. The different parts of the reasoning process are coordinated by a C program.

For evaluation purposes, we split the corpus in two samples: a training sample of 73 texts obtained at the beginning of our study and used for the development of the system, and a validation sample containing the other 87 texts collected later to test the system.

[7] We say that a reference *ref_temp(Mj)* is *alone* iff it is the only reference whose order is T_j.

To test the linguistic reasoning, we determined *a priori*, for each text, the set of its expected semantic literals and we compared them to the literals found by the system. We have already determined at the beginning of our study about 50 semantic predicates considered relevant and sufficient for the causal reasoning. Thus, it becomes easy to determine without ambiguity from a text, its expected semantic literals. To show the robustness of the linguistic reasoning we considered separately the case where its input is corrected manually and that where this input is exactly the output of the post-treatments module. For lack of space, we present only the results at the text level.

Table 1. **Results at the text level**

	Training sample		**Validation sample**	
	Raw input	Corrected input	Raw input	Corrected Input
Nb. of texts	73		87	
without errors	60 **(82%)**	63 **(86%)**	35 **(40%)**	39 **(45%)**
with expected literals not found	8 (11%)	5 (7%)	38 (44%)	37 (40%)
with additional literals	0	0	4 (5%)	4 (5%)
with literals having erroneous temporal parameter	**5 (7%)**	**5 (7%)**	**12 (14%)**	**12 (14%)**
with literals having erroneous non temporal parameters	2 (3%)	0	8 (9%)	0

Table 1 shows the obtained results. We consider that a text is erroneous if there is an error in at least one of its semantic literals. The linguistic reasoner errors are not concentrated in a part of the corpus: more than one half of the texts contain at least an error in the inferred semantic literals. This remark is crucial for the evaluation of the causal reasoning, which must be able to find correct answers even in the presence of some errors in its input. Temporal errors remain however limited. There is at least an error in temporal parameters for 14% of the validation texts, which can be considered as rather a good result. To improve these results, we think that we have to further generalize our inference rules by working at a more abstract level.

5. Conclusion and Perspectives

We presented in this paper a heuristic approach to order automatically the events evoked in a text describing a road accident. The system aims at producing a semantic representation of the text that contains temporal information about the course of the reported events. This information is represented according to a linear temporal model. This model is well adapted to reasoning about actions and has been used in a causal reasoning to detect automatically the cause of the accident described in the text. The proposed approach is based on a non monotonic reasoning that exploits linguistic and common-sense knowledge to generate precedence and simultaneity constraints. These constraints are then taken by an algorithm to update the default order of narration. This paper shows the feasibility and utility of a reasoning approach to NLU and, by considering NL texts as support of information to reasoning about actions, tries to make a bridge between these two fields of AI.

Our temporal model remains rather simple. For future work we want to enrich it by representing branching time and refining the temporal constraints that may exist between the temporal states which will undoubtedly need to develop more abstract inference rules and to develop more sophisticated algorithms to resolve the resulting constraints.

6. References

[1] J. Allen, H.A. Kautz, “A Model of Naive Temporal Reasoning”, Formal Theories of the Commonsense World, Ablex Series in Artificial Intelligence, J. Hobbs & R.C. Moore (Eds.), 1985, pp. 251-268.

[2] M. De Glas, J.P. Desclés, “Du temps linguistique comme idéalisation d’un temps phénoménal”, *Intellectica,* 1996, vol. 23, pp. 159-192.

[3] J.P. Desclés, D. Maire-Reppert, H. Oh, J. Berry, “Traitement Automatique du temps et de l'aspect pour l'analyse de textes”. TAL, Revue Semestrielle de l'ATALA, 1994, vol. 35(1), pp. 83-95.

[4] M. Gelfond, V. Lifschitz, “Classical negation in logic programs and disjunctive databases”, New Generation Computing, 1991, vol. 9(3-4), pp. 363-385.

[5] L. Gosselin, “Sémantique et temporalité en français Un modèle calculatoire et cognitif du temps et de l'aspect”, Duculot, 1996.

[6] D. Kayser, F. Nouioua, “About Norms and Causes”. International Journal on Artificial Intelligence Tools, 2005, vol. 14(1-2), pp. 7-23.

[7] D.V. McDermott, “A Temporal Logic for Reasoning about Processes and Plans”, Cognitive Science, 1982, vol. 6, pp. 101-155.

[8] F. Nouioua, “Extraction et Utilisation des normes pour un raisonnement causal dans un corpus textuel”, PhD Thesis, University of Paris XIII, 2007.

[9] F. Nouioua, P. Nicolas, “Using Answer Set Programming in an Inference-based Approach to Natural Language Semantics”, 5th International Workshop on Inference in Computational Semantics, 2006, pp. 77-86.

[10] Person C, “Traitement Automatique de la Temporalité du récit : implémentation du modèle linguistique SdT”. PhD Thesis, University of Caen, 2004.

[11] R. Reiter, “A Logic for Default Reasoning”, Artificial Intelligence, 1980, vol. 13(1-2), pp. 81-132.