

Predicting Unknown Time Arguments based on Cross-Event Propagation

Prashant Gupta

Indian Institute of Information
Technology Allahabad
Allahabad, India, 211012
greatprach@gmail.com

Heng Ji

Computer Science Department, Queens College and
the Graduate Center, City University of New York
New York, NY, 11367, USA
hengji@cs.qc.cuny.edu

Abstract

Many events in news articles don't include time arguments. This paper describes two methods, one based on rules and the other based on statistical learning, to predict the unknown time argument for an event by the propagation from its related events. The results are promising – the rule based approach was able to correctly predict 74% of the unknown event time arguments with 70% precision.

1 Introduction

Event time argument detection is important to many NLP applications such as textual inference (Baral et al., 2005), multi-document text summarization (e.g. Barzilay et al., 2002), temporal event linking (e.g. Bethard et al., 2007; Chambers et al., 2007; Ji and Chen, 2009) and template based question answering (Ahn et al., 2006). It's a challenging task in particular because about half of the event instances don't include explicit time arguments. Various methods have been exploited to identify or infer the implicit time arguments (e.g. Filatova and Hovy, 2001; Mani et al., 2003; Lapata and Lascarides, 2006; Eidelman, 2008).

Most of the prior work focused on the sentence level by clustering sentences into topics and ordering sentences on a time line. However, many sentences in news articles include multiple events with different time arguments. And it was not clear how the errors of topic clustering techniques affected the inference scheme. Therefore it will be valuable to design inference methods for more fine-grained events.

In addition, in the previous approaches the linguistic evidences such as verb tense were mainly applied for inferring the exact dates of implicit time expressions. In this paper we are interested

in those more challenging cases in which an event mention and all of its coreferential event mentions do not include any explicit or implicit time expressions; and therefore its time argument can only be predicted based on other related events even if they have different event types.

2 Terminology and Task

In this paper we will follow the terminology defined in the Automatic Content Extraction (ACE)¹ program:

entity: an object or a set of objects in one of the semantic categories of interest: persons, locations, organizations, facilities, vehicles and weapons.

event: a specific occurrence involving participants. The 2005 ACE evaluation had 8 types of events, with 33 subtypes; for the purpose of this paper, we will treat these simply as 33 distinct event types. In contrast to ACE event extraction, we exclude generic, negative, and hypothetical events.

event mention: a phrase or sentence within which an event is described.

event argument: an entity involved in an event with some specific role.

event time: an exact date normalized from time expressions and a role to indicate that an event occurs before/after/within the date.

For any pair of event mentions $\langle EM_i, EM_j \rangle$, if:

- EM_i includes a time argument *time-arg*;
- EM_j and its coreferential event mentions don't include any time arguments;

The goal of our task is to determine whether *time-arg* can be propagated into EM_j or not.

3 Motivation

The events in a news document may contain a temporal or locative dimension, typical about an unfolding situation. Various situations are evolving, updated, repeated and corrected in different event mentions. Here later information may override earlier more tentative or incomplete

¹ <http://www.nist.gov/speech/tests/ace/>

events. As a result, different events with particular types tend to occur together frequently, for example, the chains of “Conflict→Life-Die/Life-Injure” and “Justice-Convict → Justice-Charge-Indict/Justice-Trial-Hearing” often appear within one document. To avoid redundancy, the news writers rarely provide time arguments for all of these events. Therefore, it’s possible to recover the time argument of an event by gleaning knowledge from its related events, especially if they are involved in a pre-cursor/consequence or causal relation. We present two examples as follows.

• Example 1

For example, we can propagate the time “Sunday (normalized into “2003-04-06”)” from a “Conflict-Attack” EM_i to a “Life-Die” EM_j because they both involve “Kurdish/Kurds”:

[Sentence including EM_i]

Injured Russian diplomats and a convoy of America's *Kurdish* comrades in arms were among unintended victims caught in **crossfire** and friendly fire **Sunday**.

[Sentence including EM_j]

Kurds said 18 of their own **died** in the mistaken U.S. air **strike**.

• Example 2

This kind of propagation can also be applied between two events with similar event types. For example, in the following we can propagate “Saturday” from a “Justice-Convict” event to a “Justice-Sentence” event because they both involve arguments “A state security court/state” and “newspaper/Monitor”:

[Sentence including EM_i]

A *state security court* suspended a *newspaper* critical of the government **Saturday** after **convicting** it of publishing religiously inflammatory material.

[Sentence including EM_j]

The **sentence** was the latest in a series of *state* actions against the *Monitor*, the only English language daily in Sudan and a leading critic of conditions in the south of the country, where a civil war has been waged for 20 years.

4 Approaches

Based on these motivations we have developed two approaches to conduct cross-event propagation. Section 4.1 below will describe the rule-based approach and section 4.2 will present the statistical learning framework respectively.

4.1 Rule based Prediction

The easiest solution is to encode rules based on constraints from event arguments and positions of two events. We design three types of rules in this paper.

If EM_i has an event type $type_i$ and includes an argument arg_i with role $role_i$, while EM_j has an event type $type_j$ and includes an argument arg_j with role $role_j$, they are not from two temporally separate groups of Justice events {Release-Parole, Appeal, Execute, Extradite, Acquit, Pardon} and {Arrest-Jail, Trial-Hearing, Charge-Indict, Sue, Convict, Sentence, Fine}², and they match one of the following rules, then we propagate the time argument between them.

• Rule1: Same-Sentence Propagation

EM_i and EM_j are in the same sentence and only one time expression exists in the sentence; This follows the within-sentence inference idea in (Lapata and Lascarides, 2006).

• Rule2: Relevant-Type Propagation

arg_i is coreferential with arg_j ;
 $type_i = \text{“Conflict”}$, $type_j = \text{“Life-Die/Life-Injure”}$;
 $role_i = \text{“Target”}$ and $role_j = \text{“Victim”}$, or
 $role_i = role_j = \text{“Instrument”}$.

• Rule3: Same-Type Propagation

arg_i is coreferential with arg_j , $type_i = type_j$, $role_i = role_j$, and they match one of the *Time-Cue* event type and argument role combinations in Table 1.

Event Type _i	Argument Role _i
Conflict	Target/Attacker/Crime
Justice	Defendant/Crime/Plaintiff
Life-Die/Life-Injure	Victim
Life-Be-Born/Life-Marry/Life-Divorce	Person/Entity
Movement-Transport	Destination/Origin
Transaction	Buyer/Seller/Giver/Recipient
Contact	Person/Entity
Personnel	Person/Entity
Business	Organization/Entity

Table 1. Time-Cue Event Types and Argument Roles

The combinations shown in Table 1 above are those informative arguments that are specific enough to indicate the event time, thus they are

² Statistically there is often a time gap between these two groups of events.

called “Time-Cue” roles. For example, in a “Conflict-Attack” event, “Attacker” and “Target” are more important than “Person” to indicate the event time. The general idea is similar to extracting the cue phrases for text summarization (Edmundson, 1969).

4.2 Statistical Learning based Prediction

In addition, we take a more general statistical approach to capture the cross-event relations and predict unknown time arguments. We manually labeled some ACE data and trained a Maximum Entropy classifier to determine whether to propagate the time argument of EM_i to EM_j or not. The features in this classifier are most derived from the rules in the above section 4.1.

Following Rule 1, we build the following two features:

- **Feature1: Same-Sentence**

$F_SameSentence$: whether EM_i and EM_j are located in the same sentence or not.

- **Feature2: Number of Time Arguments**

$F_TimeNum$: if $F_SameSentence = true$, then assign the number of time arguments in the sentence, otherwise assign the feature value as “Empty”.

For all the Time-Cue argument role pairs in Rule 2 and Rule 3, we construct a set of features:

- **Feature Set3: Time-Cue Argument Role Matching**

$F_CueRole_{ij}$: Construct a feature for any pair of Time-Cue role types $Role_i$ and $Role_j$ in Rule 2 and 3, assign the feature value as follows:

if the argument arg_i in EM_i has a role $Role_i$
and the argument arg_j has a role $Role_j$:
if arg_i and arg_j are coreferential then
 $F_CueRole_{ij} = Coreferential$,
else $F_CueRole_{ij} = Non-Coreferential$.
else $F_CueRole_{ij} = Empty$.

5 Experimental Results

In this section we present the results of applying these two approaches to predict unknown event time arguments.

5.1 Data and Answer-Key Annotation

We used 47 newswire texts from ACE 2005 training corpora to train the Maximum Entropy classifier, and conduct blind test on a separate set of 10 ACE 2005 newswire texts. For each document we constructed any pair of event mentions

$\langle EM_i, EM_j \rangle$ as a candidate sample if EM_i includes a time argument while EM_j and its coreferential event mentions don’t include any time arguments. We then manually labeled “Propagate/Not-Propagate” for each sample. The annotation for both training and test sets took one human annotator about 10 hours. We asked another annotator to label the 10 test texts separately and the inter-annotator agreement is above 95%. There are 485 “Propagate” samples and 617 “Not-Propagate” samples in the training set; and in total 212 samples in the test set.

5.2 Overall Performance

Table 2 presents the overall Precision (P), Recall (R) and F-Measure (F) of using these two different approaches.

Method	P (%)	R (%)	F(%)
Rule-based	70.40	74.06	72.18
Statistical Learning	72.48	50.94	59.83

Table 2. Overall Performance

The results of the rule-based approach are promising: we are able to correctly predict 74% of the unknown event time arguments at about 30% error rate. The most common correctly propagated pairs are:

- From Conflict-Attack to Life-Die/Life-Injure
- From Justice Convict to Justice-Sentence/
Justice-Charge-Indict
- From Movement-Transport to Contact-Meet
- From Justice-Charge-Indict to Justice-Convict

5.3 Discussion

From Table 2 we can see that the rule-based approach achieved 23% higher recall than the statistical classifier, with only 2% lower precision. The reason is that we don’t have enough training data to capture all the evidences from different Time-cue roles. For instance, for the Example 2 in section 3, Rule 3 is able to predict the time argument of the “Justice-Sentence” event as “Saturday (normalized as 2003-05-10)” because these two events share the coreferential Time-cue “Defendant” arguments “newspaper” and “Monitor”. However, there is only one positive sample matching these conditions in the training corpora, and thus the Maximum Entropy classifier assigned a very low confidence score for propagation. We have also tried to combine these two approaches in a self-training framework – adding the results from the propagation rules as additional training data and re-train the Maximum

Entropy classifier, but it did not provide further improvement.

The spurious errors made by the prediction rules reveal both the shortcomings of ignoring event reporting order and the restricted matching on event arguments.

For example, in the following sentences:

[*Context Sentence*]

American troops stormed a presidential palace and other key buildings in Baghdad as U.S. tanks rumbled into the heart of the battered Iraqi capital on **Monday** amid the thunder of **gunfire** and **explosions**...

[*Sentence including EM_j*]

At the palace compound, Iraqis **shot** *<instrument>small arms</instrument>* fire from a clock tower, which the U.S. tanks quickly destroyed.

[*Sentence including EM_i*]

The first one was on **Saturday** and triggered intense *<instrument>gun</instrument>* **battles**, which according to some U.S. accounts, left at least 2,000 Iraqi fighters dead.

The time argument “Saturday” was mistakenly propagated from the “Conflict-Attack” event “battles” to “shot” because they share the same Time-cue role “instrument” (“small arms/gun”). However, the correct time argument for the “shot” event should be “Monday” as indicated in the “gunfire/explosions” event in the previous context sentence. But since the “shot” event doesn’t share any arguments with “gunfire/explosions”, our approach failed to obtain any evidence for propagating “Monday”. In the future we plan to incorporate the distance and event reporting order as additional features and constraints.

Nevertheless, as Table 2 indicates, the rewards of using propagation rules outweigh the risks because it can successfully predict a lot of unknown time arguments which were not possible using the traditional time argument extraction techniques.

6 Conclusion and Future Work

In this paper we described two approaches to predict unknown time arguments based on the inference and propagation between related events. In the future we shall improve the confidence estimation of the Maximum Entropy classifier so that we could incorporate dynamic features from the high-confidence time arguments which have already been predicted. We also plan to test the effectiveness of this system in textual inference, temporal event linking and event coreference

resolution. We are also interested in extending these approaches to the setting of cross-document, so that we can predict more time arguments based on the background knowledge from related documents.

Acknowledgments

This material is based upon work supported by the Defense Advanced Research Projects Agency under Contract No. HR0011-06-C-0023 via 27-001022, and the CUNY Research Enhancement Program and GRTI Program.

References

- David Ahn, Steven Schockaert, Martine De Cock and Etienne Kerre. 2006. Supporting Temporal Question Answering: Strategies for Offline Data Collection. *Proc. 5th International Workshop on Inference in Computational Semantics (ICoS-5)*.
- Regina Barzilay, Noemie Elhadad and Kathleen McKeown. 2002. Inferring Strategies for Sentence Ordering in Multidocument Summarization. *JAIR*, 17:35-55.
- Chitta Baral, Gregory Gelfond, Michael Gelfond and Richard B. Scherl. 2005. *Proc. AAAI’05 Workshop on Inference for Textual Question Answering*.
- Steven Bethard, James H. Martin and Sara Klingsstein. 2007. Finding Temporal Structure in Text: Machine Learning of Syntactic Temporal Relations. *International Journal of Semantic Computing (IJSC)*, 1(4), December 2007.
- Nathanael Chambers, Shan Wang and Dan Jurafsky. 2007. Classifying Temporal Relations Between Events. *Proc. ACL2007*.
- H. P. Edmundson. 1969. New Methods in Automatic Extracting. *Journal of the ACM*. 16(2):264-285.
- Vladimir Eidelman. 2008. Inferring Activity Time in News through Event Modeling. *Proc. ACL-HLT 2008*.
- Elena Filatova and Eduard Hovy. 2001. Assigning Time-Stamps to Event-Clauses. *Proc. ACL 2001 Workshop on Temporal and Spatial Information Processing*.
- Heng Ji and Zheng Chen. 2009. Cross-document Temporal and Spatial Person Tracking System Demonstration. *Proc. HLT-NAACL 2009*.
- Mirella Lapata and Alex Lascarides. 2006. Learning Sentence-internal Temporal Relations. *Journal of Artificial Intelligence Research* 27. pp. 85-117.
- Inderjeet Mani, Barry Schiffman and Jianping Zhang. 2003. Inferring Temporal Ordering of Events in News. *Proc. HLT-NAACL 2003*.