Recognition and Normalization of Time Expressions: ITC-irst at TERN 2004

Document Number	WP3.7
Project ref.	IST-2001-34460
Project Acronym	MEANING
Project full title	Developing Multilingual Web-scale Language Technologies
Project URL	http://www.lsi.upc.es/˜nlp/meaning/meaning.html
Availability	Public
Authors: Matteo Negri and Luca Marseglia (ITC-irst)	

INFORMATION SOCIETY TECHNOLOGIES

Project ref.	IST-2001-34460
Project Acronym	MEANING
Project full title	Developing Multilingual Web-scale Language Technologies
Security (Distribution level)	Public
Contractual date of delivery	
Actual date of delivery	February 2, 2005
Document Number	WP3.7
Type	Report
Status & version	v Final
Number of pages	8
WP contributing to the deliberable	WP3
WPTask responsible	ITC-irst
Authors	Matteo Negri and Luca Marseglia (ITC-irst)
Other contributors	
Reviewer	
EC Project Officer	Evangelia Markidou

Authors: Matteo Negri and Luca Marseglia (ITC-irst)

Keywords: Temporal expression recognition

Abstract: This paper presents the Chronos system developed at ITC-irst to participate in the English Full Task organized within the TERN 2004 evaluation. Chronos extends the capabilities of a rule-based multilingual (English/Italian) Named Entity Recognition System, allowing for the recognition and normalization of temporal expressions within an input text. To this aim, the system is designed to provide the automatic annotation of textual data with the TIMEX2 tag, which includes attributes for expressing the normalized, intended meaning or value of a broad range of temporal expressions.

Recognition and Normalization of Time Expressions: ITC-irst at TERN 2004

Matteo Negri and Luca Marseglia
ITC-irst, Centro per la Ricerca Scientifica e Tecnologica
Via Sommarive 18, 38050
Povo (TN), Italy
[negri,marseglia]@itc.it

ABSTRACT

This paper presents the Chronos system developed at ITC-irst to participate in the English Full Task organized within the TERN 2004 evaluation. Chronos extends the capabilities of a rule-based multilingual (English/Italian) Named Entity Recognition System, allowing for the recognition and normalization of temporal expressions within an input text. To this aim, the system is designed to provide the automatic annotation of textual data with the TIMEX2 tag, which includes attributes for expressing the normalized, intended meaning or value of a broad range of temporal expressions.

1. INTRODUCTION

In recent years, inspired by the success of MUC evaluations, a growing number of initiatives (*e.g.* TREC[1], CLEF[2], CoNLL[3], Senseval[4], etc.) have been developed to boost research towards the automatic understanding of textual data. Identifying and categorizing *entities* belonging to some predefined categories, retrieving an actual *answer* instead of a set of documents in response to an input question, recognizing *semantic roles* in a given sentence, or determining the correct *sense* of a word in a context, are just some examples of the many different research issues covered by these initiatives. Since 1999, the Automatic Content Extraction (ACE) program[5] has contributed to broaden the varied scenario of evaluation campaigns by proposing three main tasks, namely the recognition of *entities*, *relations*, and *events*. This year, the Timex2 Detection and Recognition task[6] (also known as TERN, for Time Expression Recognition and Normalization) has been added to the ACE program in order to make the whole evaluation exercise more complete. The main goal of the new ACE task is to foster research on systems capable to automatically detect and normalize temporal expressions present in a given source language data, with respect to the "2003 Standard for the Annotation of Temporal Expressions" [?].

Often, the research issues raised by the previously mentioned evaluation exercises are related to each other. Some tasks, in fact, are more general and dependent on the solution of problems addressed by the more specific ones. For instance, a major part of correctly recognizing *relations* is correctly recognizing the *arguments* (entities) that are related by the relation. Both these tasks, moreover, play a crucial role in the open-domain Question Answering (QA) scenario, where answers to natural language questions usually refer to entities belonging to particular *semantic classes* (*e.g.* PERSON, LOCATION, etc.), involved in particular *relations* (*e.g.* INVENTOR, CAPITAL, etc.). Under the same QA perspective, the effective treatment of *temporal expressions* has a strong impact on the capability of a system to deal with specific classes of questions, such as those asking for the DATE of a particular event (*e.g.* "*When did J.R.R. Tolkien retire from his professorship at Oxford?*"). The normalization of *relative* temporal expressions (*i.e.* the "translation" of time expressions such as "*two years later*", "*three years after*", or "*the following year*" into the real date "*1959*", see Table1) could in fact provide a QA system with a number of possible candidate answers that otherwise would be ignored ([?],[?]). Furthermore, the implementation of more sophisticated temporal reasoning mechanisms becomes necessary when dealing with complex questions concerning temporal properties or the ordering of events. For instance, as pointed out by [?], current QA systems still fall short from adequately addressing questions such as "*Is Bill Clinton* ***currently*** *the President of the United States*".

1. In **1957**, Tolkien was to travel to the United States to accept honorary degrees from Marquette, Harvard, and several other universities, and to deliver a series of addresses, but the trip was cancelled due to the ill health of his wife Edith. He retired **two years later** from his professorship at Oxford.
2. "The Adventures of Tom Bombadil" was published in **1962**, **three years after** Tolkien retired his professorship at Oxford.
3. ...Tolkien makes a brief allusion to the future of Middle-earth in a letter written in **1958**. **The following year**, after his retirement from teaching at Oxford, he ...

Table 1: Text fragments containing *relative* temporal expressions

The participation of ITC-irst to the TERN evaluation is mainly motivated by the implications of an effective treatment of temporal expressions in open-domain QA, which is

[1] http://trec.nist.gov
[2] http://clef.iei.pi.cnr.it
[3] http://cnts.uia.ac.be/conll2004
[4] http://www.senseval.org
[5] http://www.nist.gov/speech/tests/ace/index.htm
[6] http://timex2.mitre.org/tern.html

in the mainstream of the group's research activity.

The paper is structured as follows. Section 2 shortly introduces the TERN English Full task, presenting the general architecture of the Chronos system. Section 3 provides a description of the rule-based component in charge of detecting and recognizing the extent of time expressions present in a given input text. Section 4 addresses the problem of normalizing the detected temporal expressions with respect to the TIMEX2 annotation standard. Section 5 reports the results achieved by Chronos in the TERN 2004 evaluation, summarizing strengths and limitations of the system, and introducing possible directions for future improvements.

2. TASK DEFINITION AND SYSTEM ARCHITECTURE

Systems participating to the TERN English Full Task were required to automatically *detect*, *bracket*, and *normalize* relevant time expressions mentioned in the English source data. The required output is the annotation of such time expressions with TIMEX2 tags, according to the annotation guidelines specified in [**?**]. Markable time expressions include both *absolute* expressions (*e.g.* "July 17, 1999", "the summer of '69") and *relative* expressions (*e.g.* "three years ago", "last week", "yesterday night"). Also markable are durations (*e.g.* "two weeks", "the 13 years of Milosevic's rule"), event-anchored expressions (*e.g.* "two days before departure"), and sets of times (*e.g.* "every week", "daily"). ***Detection*** refers to systems' capability to recognize time expressions within an input text (at least one overlapping character in the extent of the reference and the system output was required for tag alignment). ***Bracketing*** concerns systems' capability to correctly determine the extension of a detected time expression. ***Normalization*** refers to the ability of the system to correctly assign the normalization attribute values for all the correctly detected time expressions. These attributes include:

- **VAL**: contains a normalized form of the date/time.
e.g. VAL="2004-09-06T12:30-04" and VAL="P6D", would be used for "*September 6th, 2004, at 12:30 a.m. EDT*", and "*six days*" respectively.
- **MOD**: captures temporal modifiers.
e.g. MOD="BEFORE", MOD="MORE_THAN", and MOD="START", would be used for "*more than a decade ago*", "*more than three hours*", and "*the early '70s*" respectively.
- **ANCHOR_VAL**: Contains a normalized form of an anchoring date/time.
e.g. ANCHOR_VAL="2004-08-13" would be the assigned to the time expression "*six months* in "*They worked on the system for six months, before submitting results on August 13, 2004*".
- **ANCHOR_DIR**: Captures the relative direction /orientation between VAL and ANCHOR_VAL.
e.g. Assuming that the reference time of a document is August 13, 2004, ANCHOR_DIR="AFTER" and ANCHOR_DIR="BEFORE" would be assigned to the time expression "*two weeks*" in "*I will be on vacantion for two weeks*" and "*I was in vacation for two weeks*" respectively.
- SET: Identifies expressions denoting sets of times.
e.g. SET="YES" would be assigned to expressions such as "*every 10 years*", "*annually*", and "*almost daily*".

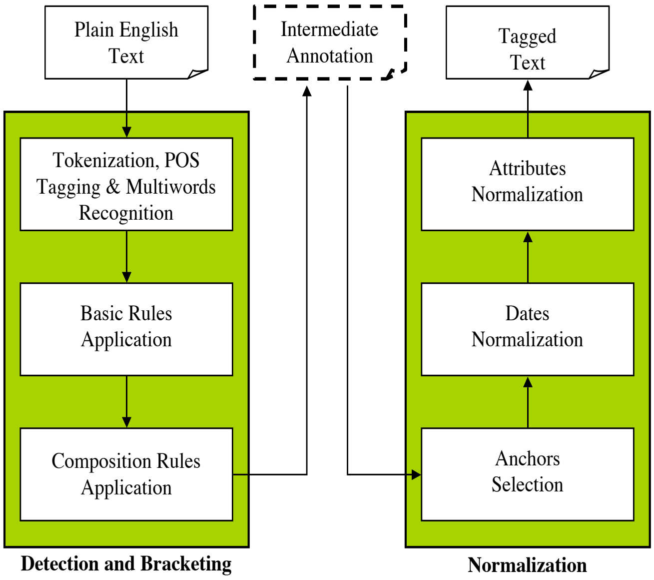

Figure 1: The architecture of Chronos

The architecture of Chronos, depicted in Figure1, relies on two main components: the *detection and bracketing* component, and the *normalization* component. The **detection and bracketing** component is in charge of the linguistic analysis of the input English text, and the production of an intermediate annotation of such text. The intermediate annotation contains all the relevant information required in the following normalization phases. The **normalization** component exploits the intermediate annotation to assign correct values to the TIMEX2 attributes of each detected time expression.

The following sections provide a detailed description of the two main components of the Chronos system.

3. DETECTION AND BRACKETING OF TIME EXPRESSIONS

Detection and bracketing of time expressions have been addressed following a rule-based approach. The resulting set of rules extends the capabilities of a multilingual (English/Italian) Named Entity Recognition (NER) System ([**?**],[**?**]) developed at ITC-irst. The process is carried out in three phases.

1. **Linguistic processing**. In the first phase, the input text is tokenized and words are disambiguated with their lexical category by means of a statistical part of speech tagger. Also multiwords recognition is carried out at this stage: multiwords expressions are recognized considering a list of about five thousand multiwords (*i.e.* collocations, compounds, and complex

terms) that have been automatically extracted from WordNet [?].

2. **Basic rules application**. In the second phase, a set of approximately 1000 hand-crafted basic rules[7] is used for the following three main purposes:

   (*i*) detect all the possible time expressions present in the input text;

   (*ii*) determine their extent;

   (*iii*) gather all the contextual information that is relevant for the following normalization phases.

3. **Composition rules application**. In the third phase, a set of higher-level rules is used to resolve ambiguities between possible multiple taggings.

The detection and bracketing component returns an intermediate annotation of the input text in which a TIMEX2 tag is assigned to each detected time expression. At this stage of the process, however, not all the information included in the tags is in the form of actual TIMEX2 attribute/value pairs. Four additional *temporary attributes* (*i.e.* the attributes "TYPE", "T-CAT", "QUANT", and "OP") are used by the system to store relevant information for the following normalization phases.

Basic rules and composition rules are described more in detail in the following two sections.

3.1 Basic Rules

Basic rules are regular expressions that check for different features of the input text. These are the presence of particular word senses, lemmas, parts of speech, symbols, or strings satisfying specific predicates. Considering such kind of information, basic rules have three main purposes: detection, bracketing, and information gathering.

Detection. Markable expressions are detected considering the presence in the input text of *lexical triggers*. Lexical triggers are words or particular configurations of numeric expressions that convey a meaning related to the concepts of time, date, and duration. Possible triggers considered by the system include: (*i*) **nouns** (*e.g.* "second", "hour", "day", "night", "week", "month", "season", "semester", "year", "decade", "century", "Seventies"), (*ii*) **proper names** (*e.g.* "Friday", "August", "Christmas"), (*iii*) **adverbs** (*e.g.* "today", "daily", "weekly", "monthly", "yearly"), and (*iv*) **numeric expressions** (*e.g.* "08/13/2004", "12:30 GMT'", "1970s", "1989").

[7]The whole set of basic rules for the recognition of time expressions has been manually created in a few weeks (around one person month) considering, as training data, the development corpora (around 317K words) of English newswire texts provided by the TERN organizers. For the other named entity categories treated by the original NER system (*i.e.* PERSON, LOCATION, ORGANIZATION, MEASURE, MONEY, CARDINAL, PERCENT), a total of around 600 rules have been written for each language. The number of TIMEX2 English rules exceeds the number of rules for all the other categories due to the richer normalized annotation required by the task.

Bracketing. Extent recognition is carried out looking at the context surrounding the detected lexical triggers. To this aim, relevant information considered by the basic rules is represented by: (*i*) **nouns** (*e.g.* "beginning", "end", "start"), (*ii*) **adjectives** (*e.g.* "next", "previous", "preceding", "following", "every", "each", "more"), (*iii*) **adverbs** (*e.g.* "ago", "shortly", "after", "before", "just", "once"), (*iv*) **prepositions** (*e.g.* "during"), and (*v*) **numbers** (*e.g.* "3", "five", "sixth").

Information gathering. Considering the detected triggers and their surrounding lexical context, basic rules are also in charge of gathering relevant information required in the following normalization phases for a complete annotation with the TIMEX2 tag (*i.e.* to fill the "VAL", "MOD", "ANCHOR_VAL", "ANCHOR_DIR", and "SET" fields). For each detected time expression, such information concerns:
(i) the presence of modifiers (*e.g.* "*more than*", "*early*", "*approximately*"). This information will be used to fill the "MOD" attribute;
(ii) the presence of expressions denoting sets of times (*e.g.* "*every*", "*twice a*") . This information will be used to fill the "SET" attribute;
(iii) the presence of clues (*e.g.* "*before*", "*later*", "*ago*", "*after*", "*during*") concerning the most likely "ANCHOR_DIR" value for relative time expressions.

Often, however, the superficial form of a time expression does not provide enough information for a correct normalization. For instance, the "VAL" attribute of relative time expressions such as "*early today*" or "*last year*" cannot be determined simply by considering the triggers and their modifiers. In these cases, some degree of reasoning considering the information provided by the lexical context in which the expressions occur is necessary. For this reasoning purpose, the temporary attributes "TYPE", "OP", "QUANT", and "T-CAT" have been introduced. For each detected time expression, they are used to store additional information concerning:
(iv) its *type*. The temporary attribute "TYPE" is filled with one of the two possible values "T-ABS" or "T-REL". The first is used to indicate *absolute* time expressions (*e.g.* "*January 27, 1972*", " *the late 1970s*", "*the early morning of 6 June 1944*"), which will be normalized in virtue of their superficial form. The second is used for relative time expressions (*e.g.* "*three years later*", "*Two weeks ago*", "*Today*"), whose normalization requires additional information (see Section 4);
(v) the *operator* to be applied for the calculation of its final "VAL" (only for relative time expressions). The attribute "OP" may assume one of the values "+", "-", or "=". For instance, the "OP" attributes assigned to the relative time expressions "*three years later*", "*Two weeks ago*", and "*Today*" will have values "+", "-", and "=" respectively;
(vi) the *quantity* that has to be added or subtracted for the calculation of its final "VAL" (only for relative time expressions). Such quantity is expressed by an integer ($n \geq 0$) assigned to the "QUANT" attribute. For instance, the "QUANT" attributes assigned to the time expressions reported in the previous examples will be filled with "3", "2", and "0" respectively;
(vii) its *granularity* (only for relative time expressions). Possible values of the "T-CAT" attribute are obtained map-

ping the detected lexical triggers to the categories [second, minute, hour, day, month, ..., millennium]. The normalization component will use this information to determine the correct anchors of relative time expressions. For instance, the "T-CAT" attributes associated to the previous examples will be "year", "week", and "day" respectively. The normalization component will select anchors with the same granularity to fill their "VAL" attributes.

An example of simple basic rule is presented in Table 2. This rule matches any POS tagged sequence "t1 t2 t3", where:
- "t1" is recognized as a determiner;
- the lemma of "t2" is "early";
- "t3" is a lexical trigger satisfying the predicate "*decade-p*" (*e.g.* "1990s", "Nineties").

For instance, it will match ***absolute*** time expressions such as "*the early 1990s*" in "*The first such micro-RNA was discovered in **the early 1990s***". As can be seen in the OUTPUT row of the table, the presence of the modifier "early" close to the lexical trigger "1990s" determines the assignment of the value "START" to the "MOD" attribute.

PATTERN	t1 t2 t3
t1	[pos = "DT"]
t2	[lemma = "early"]
t3	[pred = decade-p]
OUTPUT	<TIMEX2 val="?" type="T-ABS" mod="START"> t1 t2 t3 <\TIMEX2>

Table 2: A basic rule matching with "The early 1990s"

The result of the application of this rule will be the tagged text, where the "?" is assigned to the TIMEX2 attribute that still has to be filled:

<TIMEX2 val="?" type="T-ABS" mod="START">
the early 1990s
<\TIMEX2>

Another example is presented in Table 3, which describes a rule matching any ***relative*** time expression represented by the POS tagged sequence "t1 t2 t3 t4" (*e.g.* "*nearly three years later*" in "*The first such micro-RNA was discovered **nearly three years later** by Dr. Victor Ambros*") where:
- "t1" satisfies the predicate "*less_than-p*" (*e.g.* "almost", "nearly");
- "t2" satisfies the predicate "*number-p*" (*e.g.* "3", "five");
- the lemma of "t3" is the lexical trigger "year";
- the lemma of "t4" is "later".

In addition to the values assigned to the TIMEX2 attributes "MOD" and "ANCHOR_DIR", this rule's output contains the temporary attributes TYPE, T-CAT, OP, and QUANT. These attributes respectively provide the normalization component with information concerning: the type (*i.e.* "T-REL") of the detected time expression, its granularity (*i.e.* year), the operator that has to be applied to determine the value of "VAL" (*i.e.* "+"), the quantity that has to be added (in this case "t2", which is equal to 3).

PATTERN	t1 t2 t3 t4
t1	[pred = approx-p]
t2	[pred = number-p]
t3	[lemma = "year"]
t4	[lemma = "later"]
OUTPUT	<TIMEX2 val="?" anchor_val="?" type="T-REL" mod="LESS_THAN" t-cat="year" quant="t2" op="+" anchor_dir="ENDING" t1 t2 t3 t4 <\TIMEX2>

Table 3: A basic rule matching with "Nearly three years later"

The result of the application of this rule will be the following tagged text[8], where the "?" is assigned to the TIMEX2 attributes that still have to be filled:

<TIMEX2 val="?" anchor_val="?" type="T-REL" mod="LESS_THAN" anchor_dir="ENDING" t-cat="year" quant="3" op="+">
nearly three years later
<\TIMEX2>

According to the approach described in [?] and [?], a large part of the knowledge required to accomplish the named entity recognition task can be mined from WordNet. Lexical triggers for some named entity categories such as PERSON, LOCATION, and ORGANIZATION, can in fact be affectively captured by means of semantic predicates defined on the WordNet hierarchy[9]. Also for time expressions, even though most of the triggers have been manually selected generalizing from the training data, such WordNet-based approach has been applied. As an example, the rule represented in Table 4 utilizes the predicate "*WN-date-p*", which is satisfied by any of the 261 proper name hyponyms of the synset `calendar_day#1` (*e.g.* "Sunday", "Bastille Day", "Hanukkah", "April Fools'", etc.) mined from WordNet 1.6.

3.2 Composition Rules

The output of the basic rules application phase is processed by a set of composition rules. These rules are in charge of handling conflicts between possible multiple taggings. Such

[8]The underlined attributes, are **NOT** TIMEX2 attributes, and will appear only in the intermediate annotation. They contain temporary information that will be used by the normalization component to fill the "VAL" and "ANCHOR_VAL" fields.

[9]For instance, the predicates "*location-p*" and "*proper-location-p*", satisfied respectively by nouns related to the concept of location (*e.g.* "capital", "river") and proper location names (*e.g.* "Lisbon", "Nile"), can be defined over the high level synsets `location#1`, `mandate#2`, `road#1`, `solid_ground#1`, `body_of_water#1`, `geological_formation#1`, and `celestial_body#1`.

PATTERN	t1
t1	[pred = WN-date-p]
OUTPUT	<TIMEX2 type="T-ABS"> t1 <\TIMEX2>

Table 4: A basic rule matching with "Christmas"

conflicts may occur when a recognized time expression contains, overlaps, or is adjacent to one or more other detected time expressions. As an example, given the sentence "*I travelled for **the whole Monday night***", the basic rules application phase recognizes the following three time expressions: "*the whole Monday*", "*Monday night*", and "*the whole Monday night*". Simple composition rules considering the start/end position of the tags are used to deal with these problems. For instance, the rule represented in Table 5 selects, among two or more nested tagged expressions, the one with the largest extent (in this case "*the whole Monday night*"). Similar rules are applied to handle adjacent and overlapping tags.

PATTERN	T-EXP1 T-EXP2
T-EXP1	[start = n] [end = m]
T-EXP2	[start = n≤o<m] [end = o<p≤m]
OUTPUT T-EXP1	 [start = n] [end = m]

Table 5: A composition rule for handling inclusions

4. NORMALIZATION

The normalization component takes as input the intermediate annotation, determines the correct values for each attribute of the detected time expressions, and produces an output tagged text compliant with the TIMEX2 annotation formalism. This process is carried out in three steps: anchors selection, dates normalization, and attributes normalization.

Anchors selection. Anchors selection represents the first step of the normalization process, as it is crucial for the correct resolution of relative time expressions (*i.e.* time expressions whose "TYPE" attribute in the intermediate annotation has been set to "T-REL"). This phase is in charge of connecting each detected relative time expression (*e.g.* "*three years later*") to an absolute time expression (*e.g.* "*2001*"), which is called its *anchor*. Besides providing the correct value for the "ANCHOR_VAL" attribute, such anchor is necessary to determine the correct value to be assigned to the "VAL" attribute (in this case "*2004*"). Starting from the beginning of the document, the process is carried out trying to determine the anchors of each relative time expressions one at a time. This is due to the fact that the anchor of a relative time expression is often represented by the "VAL" assigned to a relative expression previously found in the document. In practice, each relative expression may represent the anchor of the following ones.

The current version of the system carries out the anchors selection process following two main strategies: "CR-DATE" and "PR-DATE". The "CR-DATE" heuristic associates to a relative time expression the document's creation date found at the beginning of the document[10]. The "PR-DATE" heuristic associates to a relative time expression the value of the nearest previous absolute time expression *with a compatible granularity*. According to this granularity constraint, the selected anchor must have the same or a higher degree of specificity with respect to the relative expression (*e.g.* if the "T-CAT" of the relative time expression is "month", the granularity of the anchor can be "month", "week", or "day", but not "year"). For instance, given the examples of Table1, the "PR-DATE" heuristic will select "1957", "1962", and "1958" as anchors for the resolution of the three relative time expressions "*two years later*", "*three years after*", and "*the following year*" (which have the same "t-cat=year" attribute). The selection of the heuristic is carried out considering either the lexical triggers, or their combination with the surrounding context, following the criteria outlined in Table 6.

Dates normalization. The dates normalization step is in charge of filling the "VAL" attribute of each detected time expression. Absolute time expressions are normalized in virtue of their superficial form. In this simple rewriting process, when the "TYPE" attribute associated to a time expression in the intermediate annotation is set to "T-ABS", the regular expression itself is translated in the correct normalized form by means of simple regular expressions (*e.g.* "*1990s*", "*6 June 1944*", and "*the early morning of 6 June 1944*" are respectively transcribed in transcribed in "199", "1944-06-06", and "1944-06-06TMO"). Such normalized form is then used to fill the "VAL" attribute.

Filling the "VAL" attribute of relative time expressions (whose "TYPE" attribute in the intermediate annotation is set to "T-REL") requires additional information provided by the context surrounding their lexical trigger. Such information is represented by the value of the anchor, the operator ("+", "-", or "=") to be applied to calculate the final value of the attribute, and the quantity (n≥0) that has to be added or subtracted to the value of the anchor. For instance, given the first example of Table1, once determined that:
(i) the "PR-DATE" heuristic has to be applied (since we are in the situation "*trigger*+later" described in Table 6);
(ii) the operator to be used is "+" (since "*later*" modifies the trigger "*years*");
(iii) the quantity to be added is "2" (since "*two*" modifies the trigger "*years*");
the system will correctly fill the "VAL" attribute with the value "1959".

Additional processing is required when the relative time expression does not explicitly specify the quantity to be added or subtracted to the selected anchor, as in:

[10]When the creation date is not explicitly reported, its value is induced from the document's name in the TERN corpus. For instance, given the input document "NYT20001025.1839.0279.sgm", in absence of any other information within the text the document's creation date is set to "2000-10-25".

Heuristic	*trigger*	*trigger + context*
PR-DATE		following+*trigger*, previous+*trigger*, same+*trigger*, that+*trigger*, *trigger*+before, *trigger*+later
CR-DATE	yesterday, today, tonight Monday, ..., Sunday, January, ..., December	this+*trigger*, last+*trigger* next+*trigger*, past+*trigger* *trigger*+ago

Table 6: Heuristic selection strategies for the resolution of relative time expressions

"*He started studying on March 30 2004, and passed the exam* ***the following Friday***".
Here, the normalization component has to determine how many days must be added to the "ANCHOR_VAL" (*i.e.* "2004-03-30"), in order to assign a correct value to the "VAL" attribute of the relative time expression. To this aim, first the system determines that "March 30 2004" indicates a Tuesday; then applying the operator "+" (since we are in the situation "following+*trigger*") it adds the three days which separate the anchor and "the following Friday". At the end, it calculates the final date associated to the relative time expression, filling its "VAL" attribute with the correct value "2004-04-02".

Attributes normalization. This normalization phase is in charge of producing the final tagged text, removing the temporary attributes introduced in the intermediate annotation, and adding the normalized TIMEX2 attributes "ANCHOR_VAL" and "ANCHOR_DIR". The "ANCHOR_VAL" attribute is assigned to all the time expressions denoting durations (in this case, the granularity of its value will be homogenous with the granularity of the time expression), and to expressions such as "now", "currently", "future", "former", recognized by means of regular expressions. Also the "ANCHOR_DIR" attribute is assigned to each relative expression by means of regular expressions considering either the triggers' superficial form or their surrounding context.

5. CHRONOS AT TERN 2004

Results obtained by Chronos in the English Full Task are presented in Table 7. Evaluation was carried out comparing the annotations produced by the system with a reference test corpus (around 50K words) of English news.

The first two columns, POSS and ACT, report the number of items in the reference (POSS= CORR + INCO + MISS) and the number of items in the system output (ACT= CORR + INCO + SPUR). The number of *correct* (CORR), *incorrect* (INCO), *missing* (MISS), and *spurious* (SPUR) items is also reported, both in terms of detection/bracketing (TIMEX2 and TIMEX2:TEXT rows), and in terms of normalization capabilities (all the other rows). Finally, the overall system's performance is summarized by the *precision* (PREC), *recall* (REC), and *F-measure* (F) scores reported in the last three columns of the table.

As can be observed, results demonstrate a good overall performance. However, a preliminary error analysis revealed that there is still room for improvement.

5.1 What worked

Undoubtedly, within the TERN tasks scenario, rule-based approaches are still effective and competitive with respect to machine learning techniques. Even though it is often regarded as a tedious and time-consuming work, writing rules to model the complex and variable use of temporal expressions required just around one person month. Moreover, looking at future improvements, the adoption of such approach simplifies the task of extending the system's coverage, or modifying its behavior over already covered time expressions. At the level of rules, in fact, extensions and modifications can be easily made with a direct control on the output returned by the system.

5.2 What did not work

Even though system's performance on the easiest subtasks, in particular on detection, is characterized by a very high precision score (97.6%), recall values could be higher (88% on detection, 79.8% on bracketing). The relatively high number of missing TIMEX2 tags (219, *i.e.* 11.9% of the total detectable time expressions in the reference) is only partially due to the difficulty of covering all the possible ways to express time expressions. An underestimated source of errors was in fact the conflict resolution mechanism implemented by means of composition rules (see Section 3.2), which hampered the correct tagging of embedded time expressions such as:
"<***A***>*The eve of* <***B***>*the new year*</***A***></***B***>"
"<***A***>*Sixty years ago* <***B***>*today*</***A***></***B***>".
In these cases, the removal of inclusions performed by the composition rule described in Table 5 determines the wrong annotation of only one (*i.e.* the largest possible) time expression.

Other issues that still remain beyond the scope of the current version of the system are raised by the presence in the input texts of reported speech fragments, anaphoric expressions, ambiguous time expressions and apparent dates.

Reported speech fragments raise an obvious normalization problem. Relative time expressions found in a reported speech context, in fact, require specific heuristics for the selection of the correct "ANCHOR_VAL". For instance, consider the sentence:
"*He concluded the* ***1998*** *annual meeting saying: '****The next year*** *will be* ***the eve of a new era*** *for our company'.*"
Here, the relative time expression "***The next year***", which is usually normalized using the "CR-DATE" heuristic (*i.e.* it is anchored to the document's creation date), should be normalized using the "PR-DATE" heuristic, since referring to the year after **1998**. Starting from the problem of handling reported speech fragments, future improvements of the system will concentrate on developing more accurate heuris-

Tag	*POSS*	*ACT*	*CORR*	*INCO*	*MISS*	*SPUR*	*PREC*	*REC*	*F*
TIMEX2	1828	1648	1609	0	219	39	0.976	0.880	0.926
TIMEX2:ANCHOR_DIR	351	294	245	26	80	23	0.833	0.698	0.760
TIMEX2:ANCHOR_VAL	351	398	272	56	23	70	0.683	0.775	0.726
TIMEX2:MOD	50	43	36	1	13	6	0.837	0.720	0.774
TIMEX2:SET	39	25	22	0	17	3	0.880	0.564	0.688
TIMEX2:TEXT	1828	1648	1458	151	219	39	0.885	0.798	0.839
TIMEX2:VAL	1569	1560	1365	190	14	5	0.875	0.870	0.872

Table 7: Chronos at TERN 2004: evaluation results

tics for anchors selection.

Anaphoric expressions raise a detection problem, due to the fact that they do not contain lexical triggers considered by the system. For instance, consider the following text portion:
"Evelyn Griffin has seen has seen ***80 winters****.* ***This****, she says, was the coldest."*
Since Chronos does not consider pronouns as possible lexical triggers, the relative time expression "***This***" will not be detected. Future investigations will aim to determine when anaphoric expressions found in a text convey some temporal meaning.

Ambiguous expressions (*e.g.* the noun "*April*", which can also be the name of a person) and apparent dates raise the problem of spurious taggings. Apparent dates are proper names designating something other than a temporal entity, but containing lexical triggers. The creation of a "stopword" list of such proper names (*e.g.* "USA Today", "Daily Telegraph", "20th Century Fox", "Black Sabbath") was just a partial (due to its intrinsic incompleteness) solution to a problem that demands for further research.

6. REFERENCES

[1] C. Fellbaum. *WordNet, An Electronic Lexical Database.* The MIT Press, 1998.

[2] L. Ferro, L. Gerber, I. Mani, B. Sundheim, and G. Wilson. 2003 standard for the annotation of temporal expressions. Technical report, MITRE Corporation, 2004.

[3] B. Magnini, M. Negri, R. Prevete, and H. Tanev. A WordNet-based Approach to Named Entity Recognition. In *Proceedings of SemaNet, COLING Workshop on Building and Using Semantic Networks*, Taipei, Taiwan, August 2002.

[4] I. Mani. Recent Developments in Temporal Information Extraction. In t. a. John Benjamins, editor, *Proceedings of the Conference on Recent Advances in Natural Language Processing.* N. Nicolov and K. Mitkov, 2004.

[5] M. Negri and B. Magnini. Using WordNet Predicates for Multilingual Named Entity Recognition. In *Proceedings of the Second International WordNet Conference (GWC 2004)*, Masaryk University, Brno, Czech Republic, January 2004.

[6] J. Pustejovsky, J. Castano, R. Ingria, R. Sauri, R. Gaizauskas, A. Setzer, G. Katz, and D. Radev. Temporal Information Extraction for Temporal Question Answering. In *New Directions in Question Answering, Papers from 2003 AAAI Spring Symposium*, 2003.

[7] E. Sequete, P. Martínez-Barco, R. Muñoz, and J. Vicedo. Splitting Complex Temporal Questions for Question Answering Systems. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL 2004)*, 2004.