

# Extending HeidelTime for Temporal Expressions Referring to Historic Dates

Jannik Strötgen, Thomas Bögel, Julian Zell, Ayser Armiti, Tran Van Canh, Michael Gertz

Institute of Computer Science, Heidelberg University, Germany  
{stroetgen, thomas.boegel, zell, ayser.armiti, canh.tran.van, gertz}@informatik.uni-heidelberg.de

## Abstract

Research on temporal tagging has achieved a lot of attention during the last years. However, most of the work focuses on processing news-style documents. Thus, references to historic dates are often not well handled by temporal taggers although they frequently occur in narrative-style documents about history, e.g., in many Wikipedia articles. In this paper, we present the AncientTimes corpus containing documents about different historic time periods in eight languages, in which we manually annotated temporal expressions. Based on this corpus, we explain the challenges of temporal tagging documents about history. Furthermore, we use the corpus to extend our multilingual, cross-domain temporal tagger HeidelTime to extract and normalize temporal expressions referring to historic dates, and to demonstrate HeidelTime’s new capabilities. Both, the AncientTimes corpus as well as the new HeidelTime version are made publicly available.

**Keywords:** Computational history, temporal tagging, HeidelTime

## 1. Introduction

Temporal information is very frequent in many types of documents such as news articles or Wikipedia documents. Furthermore, for many natural language processing and understanding tasks (e.g., topic detection and tracking, document summarization, and machine translation) it is important to identify and understand temporal information occurring in the documents that are processed.

The first step of the full task of temporal information extraction is temporal tagging, i.e., the extraction and normalization of temporal expressions occurring in text documents. In the last years, there has been a lot of research on temporal information extraction in general, and on temporal tagging in particular, reflected by the TempEval series for instance (Verhagen et al., 2009; Verhagen et al., 2010; UzZaman et al., 2013). However, despite some exceptions, e.g., the development of a temporally annotated corpus containing narratives (Mazur and Dale, 2010), most of the research deals with documents from the news domain. These documents typically contain many temporal expressions referring to current times and dates.

In this paper, our subject of interest is temporal tagging documents about history or historic events. When processing such documents, many temporal taggers fail to detect and correctly normalize temporal expressions referring to historic dates. Especially machine learning-based temporal taggers that are trained on news documents but also rule-based approaches not considering documents about history do not correctly extract and normalize such temporal expressions. For example, our temporal tagger HeidelTime (Strötgen and Gertz, 2013) – despite the fact that it was the first temporal tagger applying domain-sensitive temporal tagging strategies – so far did not detect and normalize date expressions referring to years before the year 1000 or even to years before Christ (BC).

While one could easily add a couple of rules to detect some date expressions referring to historic dates, the following challenges have to be kept in mind if one wants to complete the task more accurately.

- a) Temporal expressions can be ambiguous:
  - Depending on the context, expressions such as “in the year 90” can either refer to 90 AD (in documents about history) or to 1990 (e.g., in current newspaper articles).
  - “BC” is not always expressed explicitly. Thus, “in the year 90” could even refer to 90 BC, e.g., in “In 95 BC ... In the year 90 ...”
- b) Not only explicit, but also relative and underspecified expressions should be normalized correctly, independent of whether they refer to current or historic dates.
- c) Capabilities for extracting and normalizing temporal expressions referring to historic dates should not decrease the temporal tagging quality on documents containing mainly temporal expressions referring to current dates – even though occurring expressions may be ambiguous as described above.
- d) In different languages, such date expressions are often expressed in similar ways, but there are also language-specific differences, as will be discussed later.<sup>1</sup>

In this paper, we study the challenges of temporal tagging documents about history, and explain how we extend our temporal tagger HeidelTime (Strötgen and Gertz, 2013) to address these challenges. For this, we created the multilingual AncientTimes corpus containing Wikipedia articles covering different time periods, and manually annotated all occurring temporal expressions. The corpus consists of documents in the eight languages currently supported by HeidelTime (English, German, Dutch, Spanish, French, Italian, Arabic, and Vietnamese). Based on this corpus and the identified challenges, we extend HeidelTime to also address temporal expressions referring to historic dates. In addition to releasing a new HeidelTime version, we make the multilingual AncientTimes corpus publicly available.

<sup>1</sup>Language-specific challenges for temporal tagging of Spanish, Italian, Arabic, and Vietnamese in general are described in (Strötgen et al., 2014).

The remainder of the paper is structured as follows: After a brief survey of related work in Section 2, details about the AncientTimes corpus and the challenges of temporal tagging documents about history are presented in Section 3. In Section 4, we describe how we extended HeidelTime to cover temporal expressions referring to historic dates. Finally, we compare the evaluation results of HeidelTime’s previous and new versions in Section 5.

## 2. Related Work

There are two frequently used standards for annotating temporal expressions in text documents, TIDES TIMEX2 (Ferro et al., 2005) and TIMEX3 as part of the TimeML markup language (Pustejovsky et al., 2005). Although TIMEX2 and TIMEX3 are quite similar in many aspects, there are also important differences. For instance, TimeML covers not only temporal expressions but also events and temporal relations so that events are not part of TIMEX3 expressions (e.g., “<TIMEX3>three years</TIMEX3> after the revolution”). In contrast, there are so-called event-anchored temporal expressions annotated in TIMEX2 (e.g., “<TIMEX2>three years after the revolution</TIMEX2>”).

For several languages, there are publicly available corpora in which temporal expressions are manually annotated either with TIMEX2 or TIMEX3 tags. English temporally annotated corpora include the ACE TERN 2004 and 2005 data sets developed in the context of the Automatic Content Evaluation competitions<sup>2</sup>, the TimeBank (Pustejovsky et al., 2003) and the Aquaint corpora as well as the TE-3 platinum corpus, which have been used as training and test sets in the TempEval-3 challenge (UzZaman et al., 2013), respectively. For other languages, there are, for instance, the French and Spanish TimeBank corpora (Bittar et al., 2011; Saurí and Badia, 2012) and the Italian I-CAB corpus (Magnini et al., 2006).

Note however, that almost all publicly available temporally annotated corpora contain news-style documents with temporal expressions that mostly refer to current dates. In contrast, the English WikiWars corpus (Mazur and Dale, 2010) and its German counterpart WikiWarsDE (Strötgen and Gertz, 2011) contain narrative-style documents, namely parts of Wikipedia articles about important wars in history. In Table 1, we show the number of temporal expressions annotated in several publicly available corpora, and how many of these expressions are dates referring to either BC times or to AD times before the year 1000. Expressions referring to AD times before 1000 hardly occur except in the ACE TERN 2004 corpus. Here, however, almost all of these expressions occur in the same document, and most of them are expressed in very similar ways. In contrast, the WikiWars and WikiWars corpora contain several temporal expressions referring to BC dates. In all other corpora, there are no such expressions besides four exceptions.

It is crucial to note that there are significant challenges in temporal tagging documents of different domains. In previous work, we showed that different normalization strategies should be applied by a temporal tagger depending

corpus	temporal expressions		
	total	BC dates	AD < 1000
ACE TERN 2004	8,938	4	27
ACE 2005 English	5,266	0	0
TimeBank (TE-3)	1,426	0	0
Aquaint (TE-3)	652	0	0
TE-3 platinum	158	0	0
WikiWars	2,681	135	0
Italian I-CAB	4,573	0	2
Spanish TimeBank	1,322	0	0
French TimeBank	641	0	0
WikiWarsDE	2,240	78	0

Table 1: Temporal expressions referring to historic dates in several English and non-English publicly available corpora.

on whether news- or narrative-style documents are processed (Strötgen and Gertz, 2012; Strötgen and Gertz, 2013). A main difference is that in narrative-style documents, the reference time of underspecified temporal expressions (e.g., “November”) has to be detected in a document’s text, while in news-style documents, the document creation time is usually the reference time.

## 3. The AncientTimes Corpus

In this paper, we focus on temporal expressions referring to historic dates. Since such expressions are rare in publicly available temporally annotated corpora, we developed the AncientTimes corpus, which we present in the following. At the end of this section, we summarize the challenges of temporal tagging documents about history.

### 3.1. Annotation Standard

As described in Section 2, there are two widely used standards for annotating temporal expressions: TIDES TIMEX2 (Ferro et al., 2005) and TIMEX3 as part of the TimeML markup language for temporal annotations (Pustejovsky et al., 2005). Since TimeML is the annotation standard mainly used in more recent works, e.g. in the TempEval challenges (Verhagen et al., 2009; Verhagen et al., 2010; UzZaman et al., 2013), and because our temporal tagger HeidelTime also annotates temporal expressions with TIMEX3 tags (Strötgen and Gertz, 2013), we chose TimeML’s TIMEX3 tags to annotate temporal expressions in the AncientTimes corpus.

While TIMEX3 tags contain several attributes, the most important ones are the *value* and *type* attributes. The latter specifies whether a temporal expression is of the type date, time, duration, or set. The value attribute covers the most important semantics of temporal expressions – in particular of date expressions – in a normalized way. For example, the value attribute of the date expression “October 15, 2013” is set to “2013-10-15”.

Examples of date expressions we are mainly interested in in this work are “February 280” and “312 BC”. Here, the normalization standard requires that the years are written as four-digit numbers and that expressions referring to BC times get the prefix “BC” within their value attributes. Thus, the values of the two example expressions are to be normalized to “0280-02” and “BC0312”, respectively.

<sup>2</sup><http://www.itl.nist.gov/iad/mig/tests/ace/>.

### 3.2. Document Selection

A main requirement for the AncientTimes corpus is that it contains text documents about different time periods. So far, our temporal tagger HeidelbergTime did not correctly extract and normalize temporal expressions referring to years before the year 1000. Due to the different challenges briefly introduced in Section 1, we chose four time periods that the documents should cover: (i) 100–999 AD, (ii) 1–99 AD, (iii) 99–1 BC, and (iv) 100 BC or earlier.

A second requirement for the AncientTimes corpus is that it contains documents written in all languages currently supported by HeidelbergTime. A reasonable source of documents available in multiple languages about content covering the above mentioned time periods is Wikipedia. After determining potential documents in one language, Wikipedia's interlanguage links<sup>3</sup> can be used to validate if the documents are also available in other languages.

As one of the first works addressing temporal tagging non-news documents, Mazur and Dale (2010) developed the WikiWars corpus containing temporally annotated parts of English Wikipedia articles about important wars in history (c.f. Section 2). While most of the documents are war descriptions about more recent wars such as the Soviet war in Afghanistan (1979–1989) or World War I and II (1914–1918 and 1939–1945, respectively), there are also two documents about earlier wars containing temporal expressions referring to historic dates: the Greco-Persian Wars (499–450 BC) and the Punic Wars (264–146 BC). Thus, as a starting point, we use these two documents and the corresponding documents of the German counterpart of WikiWars, WikiWarsDE. Since Wikipedia contains both documents in all eight languages with one exception (there is no Punic Wars article in Vietnamese), the time period “100 BC or earlier” is well covered by these documents.

To find Wikipedia documents about similar content covering the other three time periods, we checked the Wikipedia page listing wars before the year 1000,<sup>4</sup> and selected those documents being available in most of the eight languages: “Bar Kokhba revolt” (132-136 AD), “Year of the Four Emperors” (49 AD), and “Julius Caesar's Gallic Wars” (58-50 BC). Since some of the documents were not available in Vietnamese or Arabic or were too short to be valuable, we selected alternative articles for the respective documents. In addition, since some documents were very long, we only used parts of the documents in some languages.

Details about the documents, such as their lengths and the number of annotated temporal expressions, are presented in the following section.

### 3.3. Annotation Process and Corpus Statistics

In all documents, we manually annotated the extents of occurring temporal expressions with TIMEX3 tags and specified normalization information (the type and value attributes). Note that WikiWars and WikiWarsDE are annotated according to TIDES TIMEX2 (Ferro et al., 2005). Although the differences between TIMEX2 and TIMEX3 are rather minor (c.f. Section 2), we manually translated the

annotations in the four WikiWars/WikiWarsDE documents into TIMEX3. In all languages, the documents were annotated by a language expert. Then, all annotations were discussed by the first annotator and a TimeML expert.

In summary, the AncientTimes corpus contains 39 TIMEX3-annotated documents (4-5 documents in 8 languages), and the details about the AncientTimes corpus are provided in Table 2. Although the number of tokens and temporal expressions varies between time periods and languages of the documents, we were able to study the challenges of temporal tagging documents about history based on this corpus, as will be described in the following section. Thus, the AncientTimes corpus can be regarded as a valuable multilingual resource for research on temporal tagging content about history.

### 3.4. Characteristics of Temporal Expressions Referring to Historic Dates

Based on our annotations in the AncientTimes corpus, we analyzed the characteristics of temporal expressions referring to historic dates in the different languages. In addition, when detecting language-specific writings in the documents of the AncientTimes corpus for one language, we searched for similar constructions in the other languages outside the corpus, in particular in further Wikipedia articles. Finally, we studied alternative language-specific writings not occurring in the corpus at all. By applying this exploratory strategy, we aimed at detecting as many different writings of temporal expressions referring to historic dates as possible. This is crucial to achieve high coverage when extending HeidelbergTime as will be detailed in Section 4. Our main findings about how temporal expressions referring to historic dates are expressed in the different languages can be summarized as follows:

- Date expressions referring to years between 1 and 999 can be written in ways very similar to date expressions referring to current dates in the corresponding language.
- There are language-specific writings to explicitly refer to AD and BC years.
- The closer to the year 1 an expression refers to, the more likely it is that it is explicitly stated whether the year expression refers to the year AD or BC. However, explicitly referring to BC times is more common than explicitly referring to AD times.
- In English, the typical expressions to refer to such dates are “Anno Domini” and “Before Christ” with their often used abbreviations (AD/A.D. and BC/B.C.).
- In addition, the terms “Common Era” (Christian Era, Current era) and “Before the Common Era” (Before the Christian Era, Before the Current Era) or their abbreviations (CE, BCE) can be used as alternatives.
- In German, Spanish, Dutch, Italian, French, Arabic, and Vietnamese similar expressions exist, e.g., “vor/nach Christus” (v. Chr. and n. Chr.) in German.

<sup>3</sup>[http://en.wikipedia.org/wiki/Help:Interlanguage\\_links](http://en.wikipedia.org/wiki/Help:Interlanguage_links)

<sup>4</sup>[http://en.wikipedia.org/wiki/List\\_of\\_wars\\_before\\_1000](http://en.wikipedia.org/wiki/List_of_wars_before_1000)

language		Greco-Persian (499–450 BC)	Punic Wars (264–146 BC)	Gallic Wars (58–50 BC)	4 Emperors (69 AD)	Bar Kokhba (132–136 AD)	total
English	token	7,326	3,443	4,097	1,789	2,188	18,843
	timex	126	54	40	51	40	311
German	token	3,973	3,760	2,247	695	897	11,572
	timex	36	67	56	21	16	196
Spanish	token	4,110	5,546	5,313	1,573	1,178	17,720
	timex	40	69	49	37	22	217
Italian	token	3,119	2,908	5,985	7,163	910	20,085
	timex	38	54	76	57	9	234
French	token	10,769	2,901	7,690	1,168	791	23,319
	timex	130	25	95	27	13	290
Dutch	token	910	460	2,878	1,358	948	6,554
	timex	15	11	40	53	11	130
Arabic	token	624	1,458	902*	520*	1,152*	4,656
	timex	15	32	22*	6*	30*	105
Vietnamese	token	–	1,880	635	1916	2,447*	6,878
	timex	–	33	8	29	50*	120

Table 2: Statistics of the AncientTimes corpus. Some documents were too short to be valuable or did not exist at all in some languages. We used alternative documents with content about the same time periods (respective numbers marked with \*).

- In all languages, expressions referring to AD dates often do not contain explicitly stated hints (e.g., “AD”). However, there are also many expressions without explicitly stated hints that refer to BC dates.
- Understanding the context often helps to identify such elliptical constructions: For example, in “In 250 BC, ... In 249,” the second expression clearly refers to 249 BC although the “BC” is not explicitly stated.
- In some documents, it is only stated explicitly at the beginning of the document that the document is about something happening BC. Almost all following expressions referring to dates do not contain explicit hints that they refer to dates BC.
- Negative numbers can be used to refer to BC years instead of terms or abbreviations such as “BC”. While in the AncientTimes corpus only one French document contains such expressions, such writings also occur in other languages in other texts.
- Sometimes phrases such as “in the year” are used in combination with two- and three-digit numbers to refer to specific years instead of just using “in” followed by the respective number (e.g., “in the year 99” vs. “in 99”). These phrases help to decide whether numbers are used as year expressions or for counting purposes.
- current or previous century. For example, in a news article from 2005, the expression “May 99” is likely to refer to the May of 1999.
- In the AncientTimes corpus, and in documents about history in general, such expressions typically refer to the years in the 1st century BC or AD even if no phrase such as “AD” or “BC” is explicitly stated.
- In Table 3, we show the characteristics of the date expressions occurring in the AncientTimes corpus for four of the eight languages. Although in the English and Spanish documents of the AncientTimes corpus, explicitly referring to BC years is quite common (column 1 vs. column 2 in Table 3), the corresponding numbers for French and German show that assuming that expressions without explicit mentions of BC-terms refer to dates in AD times would result in many incorrect normalizations.
- The temporal expressions in the AncientTimes corpus referring to AD years only sometimes contain explicit hints such as “AD” (columns 3 and 4 in Table 3).
- Determining whether temporal expressions in documents about history refer to dates BC or AD is challenging since elliptical constructions occur frequently.

These characteristics of temporal expressions referring to historic dates result in several challenges for temporal tagging content about history as will be detailed next.

### 3.5. Challenges for Temporal Tagging

In addition to the fact that patterns and phrases to explicitly refer to BC and AD times have to be known by a temporal tagger, the following challenges should also be kept in mind when addressing temporal tagging content about history.

- In news-style documents, temporal expressions with two-digit years usually refer to the specific year in the
- Incorrect normalizations of expressions often result in further normalization errors, because relative expressions (e.g., one year later) and underspecified expressions (e.g., in November) are quite frequent in all languages (column 5 in Table 3). While in news-style documents the reference time of underspecified expressions is usually the document creation time, in narrative-style documents such as the ones of the AncientTimes corpus, the reference times of both types of expressions have to be detected in the documents’ texts (Strötgen and Gertz, 2012).

	dates with explicit year information				dates without year information	
	referring to BC times		referring to AD times		relative / underspec.	past-, present future-ref
	explicitly	not explicitly	explicitly	not explicitly		
English	112	11	17	32	66	17
German	75	25	3	30	29	14
Spanish	81	4	8	21	20	27
French	66	71	0	38	28	29

Table 3: Date expressions in the AncientTimes corpus (four languages) separated into dates with explicit year expressions with and without explicit hints to “BC” and “AD”, and into dates without explicit year information, i.e., relative and underspecified expressions as well as expressions that are to be normalized to PAST\_REF, PRESENT\_REF, or FUTURE\_REF.

- It is important that the reference times are correctly normalized. Otherwise, there is no chance to normalize the underspecified or relative expressions correctly.
- Elliptical constructions and expressions with years before 100 and in particular before 32 (due to ambiguities with days) are challenging for temporal taggers. For example, “January 20” can refer to 0020-01, BC0020-01, or to the 20th of January in any other year. In the latter case, the reference year would have to be detected to fully normalize the expression (e.g., to 2013-01-20, assuming that the reference year is 2013).

In summary, to correctly extract and normalize temporal expressions referring to historic dates, a temporal tagger should (i) be aware of special phrases referring to “BC” and “AD”, (ii) distinguish between news- and narrative-style documents, and (iii) validate whether temporal expressions without explicit phrases for “BC” and “AD” refer to BC or AD dates. Finally, extensions to handle temporal expressions referring to historic dates should not negatively influence a temporal tagger’s extraction and normalization quality on news-style documents, and on documents about more current content in general.

## 4. HeidelTime Adaptations

After having detailed the characteristics of temporal expressions referring to historic dates and the resulting challenges for temporal taggers, in the following, we describe how we extended HeidelTime to deal with such expressions.

For this, we briefly present crucial details of HeidelTime’s architecture. Then, we describe the resources and rules we added and modified to extract and normalize temporal expressions referring to historic dates. Finally, we explain how HeidelTime’s normalization procedure was adapted.

### 4.1. HeidelTime’s Architecture

HeidelTime is a rule-based, multilingual, cross-domain temporal tagger. It uses language-dependent resources (patterns, normalization information, and rules) and language-independent but domain-sensitive normalization strategies.

**Language-dependent Resources.** HeidelTime clearly separates between its algorithmic part and its language-dependent resources being organized outside the source code. The latter ones consist of three types of resources.

(i) Pattern resources contain patterns frequently used to form temporal expressions. For example, for each language, there is a pattern file containing names of months.

(ii) Normalization resources contain mappings from patterns (e.g., names of months) to their normalized values, for instance, that “March” is normalized to “03”. Note that it is neither important for the pattern nor for the normalization resources that “March” is an ambiguous term that can also be used as regular noun without temporal meaning.

(iii) Rule resources contain rules defining how pattern resources are combined for the extraction of temporal expressions, and how normalization resources are combined to normalize the extracted patterns in some standard format. Note that one can make use of regular expressions, define part-of-speech constraints, and specify negative rules, e.g., to determine if ambiguous expressions such as “march” occur without being part of a temporal expression.<sup>5</sup>

To correctly extract and normalize date expressions referring to historic dates, we had to specify patterns and add normalization information for all languages. Furthermore, we had to adapt some of the existing rules and added some new rules, as will be detailed in Section 4.2.

**Different Domains.** For the normalization of temporal expressions, different text domains pose different challenges. In particular, a temporal tagger should apply different strategies for the normalization of underspecified date expressions (c.f. Section 2).

HeidelTime applies domain-dependent normalization strategies (Strötgen and Gertz, 2013). Thus, when the domain-dependent challenges described in the previous section are addressed (e.g., the different handling of 2-digit year expressions in news- and narrative-style documents), domain-dependent issues should be solved within HeidelTime’s algorithmic part, i.e., in the source code, rather than within the language-dependent resources.

In Section 4.3, we explain how we modified HeidelTime’s normalization strategies for addressing temporal expressions referring to historic dates.

### 4.2. Adding Resources and Adapting Rules

For every language, we added pattern files to detect (i) 1- to 4-digit numbers and (ii) expressions explicitly referring to either “BC” or “AD”. For both, we also added normalization information to the normalization resources (e.g., that the normalized value of “220” is “0220”). Using these pattern and normalization resources, we modified some of the original rules and added several new rules. In the following, we describe several examples.

<sup>5</sup>For further information about the language-dependent resources and HeidelTime’s rule syntax, we refer to (Strötgen and Gertz, 2013) and (Strötgen et al., 2014).

**Expressions with BC/AD patterns.** HeidelbergTime’s rule sets already contained several rules requiring a four-digit number referring to a year to detect temporal expressions with fully specified year information (e.g., to detect expressions of the form “month year”, “month day, year”, and “season of year”). To extract and normalize similar expressions referring to historic dates, we copied the rules and replaced the four-digit number with a one- to four-digit number in combination with a BC/AD pattern. With these rules, HeidelbergTime extracts and correctly normalizes most of the expressions that contain explicit references to BC and AD years (c.f. columns 1 and 3 in Table 3).

However, there are still many temporal expressions that are not yet extracted (c.f. columns 2 and 4 in Table 3). Note that such missing expressions also result in wrong normalizations of several relative and underspecified expressions (c.f. column 5 in Table 3) since they are required as reference times (necessary adaptations to HeidelbergTime’s normalization procedure to handle historic dates as reference times will be detailed in Section 4.3). Thus, it is crucial to also address date expressions without explicit hints to whether they refer to AD or BC dates.

**Expressions without BC/AD patterns.** For expressions with year information but without explicit BC/AD patterns, we distinguish several cases due to ambiguities of numbers. These occur in text documents either to refer to years or they are used for counting purposes:

- For expressions with 3-digit numbers and further temporal patterns such as names of months or seasons or words like “year”, we added several rules in all languages (similar rules already existed with 4-digit numbers). Due to the other terms with temporal meanings, it is very likely that a 3-digit number refers to a year. The normalization of such expressions is to the respective dates AD. In a post-processing step described in Section 4.3, HeidelbergTime analyzes if it is more likely that the expressions refer to the respective dates BC.
- For expressions with 2-digit numbers referring to years, HeidelbergTime already contains rules in most languages since such expressions are quite frequent to refer to current dates. For instance, “December 99” in current news articles has the meaning “1999-12”. In documents about history, such expressions with 2-digit numbers also occur but refer to the respective dates in the 1st century BC or AD. To fully normalize such expressions, HeidelbergTime so far determined a reference time and used the same century (or the century before or after the century of the reference time) to normalize the respective expression. While no new rules with 2-digit years were added, we explain in Section 4.3 how we modified HeidelbergTime’s normalization procedure to distinguish whether a 2-digit number is an abbreviated year or the full year information.
- Some temporal expressions only consist of a number to refer to a year. For several languages, HeidelbergTime already contains rules to extract 4-digit numbers as temporal expressions, and, in addition, several negative rules to extract 4-digit numbers if they do not re-

fer to a year. For instance, “2000” is extracted as temporal expression in the phrase “in 2000, there were ...” while it is not extracted as temporal expression in the phrase “in 2000 kilometers” due to a negative rule matching a 4-digit number followed by a plural noun. We added similar context-dependent rules to extract 2- and 3-digit numbers as temporal expressions (e.g., if the number is preceded by a preposition), and some rules to match 2- and 3-digit numbers if they do not refer to a year. However, for some languages such rules are less reliable. For instance, in German it is rather difficult to determine context-dependent patterns to distinguish the two cases so that we excluded such rules. While some temporal expressions are thus missed, we avoid to incorrectly match frequently occurring numbers not used to refer to a year.

- For languages with negative rules to match “number + plural noun” patterns (e.g., “in 2000 kilometers”) as not being temporal expressions, a positive duration rule matches expressions such as “2000 years” (with “years” being also a plural noun).

#### 4.3. Adapting HeidelbergTime’s Normalization Procedure

Three main issues were addressed by adapting HeidelbergTime’s normalization procedure.

**Relative and underspecified expressions.** In addition to temporal expressions covered by the modified and new rules, we also have to take care of relative and underspecified expressions (e.g., “the following year” or “November”). While their extraction is identical no matter whether such expressions refer to current dates or historic dates, their normalization becomes more complicated if BC years and years before 1000 come into play. However, not the normalization information as part of the language-dependent rules and resources have to be adapted, but the language-independent normalization procedure implemented in HeidelbergTime’s source code.

For the normalization of relative and underspecified expressions, a reference time has to be detected. In previous HeidelbergTime versions, each reference time had to start with a 4-digit year expression (if the millennium, century, or decade of a reference time was checked, the assumption was that the reference time started with at least a 1-, 2-, or 3-digit number, respectively). Now, we reimplemented HeidelbergTime’s reference time functions so that BC years and years less than 1000 are handled as well. In addition, the functions to calculate values of underspecified and relative expressions were updated accordingly. In Table 4, examples to calculate values of relative expressions such as “the following year” and “the next month” are shown.

**Disambiguating 2-digit Year Expressions.** In news-style documents, we assume that 2-digit year expressions refer to the current, previous or next century related to the document creation time. In narrative-style documents, we distinguish if the reference time of expressions with 2-digit year patterns refers to a date before the 11th century BC/AD. If so, we assume the respective century for the normalization of 2-digit years. Otherwise, we assume that the

expression	normalized reference time	normalized value
the following year	2012	2013
the following year	BC0250	BC0249
the following year	BC0001	0001
the following year	0099	0100
the next month	2012-12	2013-01
the next month	0058-10	0058-11
the next month	BC0250-12	BC0249-01
the next month	BC0001-12	0001-01

Table 4: Examples for calculating values of relative expressions based on the detected reference time.

2-digit years refer to dates in the 1st century BC or AD because abbreviating 3-digit years is uncommon.

This procedure works well for the documents of the AncientTimes corpus without negatively influencing HeidelbergTime’s performance on news-style documents or on narrative-style documents about more recent content.

#### Disambiguating Expressions without BC/AD Patterns.

Since temporal expressions without explicit BC/AD patterns are quite frequent in many documents of the AncientTimes corpus (c.f. Table 3), we added a post-processing step to HeidelbergTime’s normalization procedure. As described in Section 4.2, such expressions are preliminarily normalized to the respective AD dates. However, if there are expression in the same document referring to a BC date, the following decisions are made:

Assuming a date expression  $e_i$ . If one of the five previously mentioned date expressions of  $e_i$  refers to the same century as  $e_i$  – however in BC – then, the normalized value of  $e_i$  is transferred to the respective BC date.<sup>6</sup> For instance, assuming the text “4th century BC ... In May 390 ... in the year 350 ...”, three expressions are extracted and initially normalized:  $e_1$ =“4th century BC” (BC03),  $e_2$ =“May 390” (0390-05), and  $e_3$ =“the year 350” (0350). Since all expressions have the same century information (4th century normalized as “03”), the normalizations of  $e_2$  and  $e_3$  are transferred to the respective BC dates, i.e., to “BC0390-05” and “BC0350”, respectively. However, due to the closeness to the BC/AD boundary, we make the following exception: If both expressions refer to a date in the first or second century, the value of  $e_i$  is only transferred if the previous expression refers to a date chronologically before  $e_i$ .

Note that it is important to not assume that all expressions without BC/AD hints refer to BC if there is an earlier mentioned expression with a BC hint. Assume the text “In 57 AD ... in the 5th century BC ... In 58 ...”, the value of “58” would be wrongly transferred to “BC0058”.

Using the described post-processing rules, such incorrect transfers are avoided, and many of the expressions with year information but without explicit BC/AD hints are normalized correctly (c.f. Section 5).

<sup>6</sup>The same procedure is used if one of the five previously mentioned date expressions of  $e_i$  refers to “the same century + 1”, because if both refer to BC dates, they occur close to each other chronologically – a typical assumption for small text passages in narrative texts (Strötgen and Gertz, 2012).

(a) HeidelbergTime’s 1.5 and new versions on WikiWars.

	relaxed			strict			value F1
	P	R	F	P	R	F	
1.5	95.6	82.0	88.3	87.5	75.1	80.8	78.2
new	95.8	85.4	90.3	88.2	78.5	83.1	80.8

(b) HeidelbergTime’s 1.5 and new versions on WikiWarsDE.

	relaxed			strict			value F1
	P	R	F	P	R	F	
1.5	98.4	84.5	90.9	92.6	79.5	85.6	79.4
new	98.9	88.3	93.3	93.0	83.0	87.7	82.2

Table 5: Evaluation results on WikiWars corpora.

## 5. Evaluation

To demonstrate the quality of HeidelbergTime’s new version, we compare it to HeidelbergTime’s previous version (1.5).

**Evaluation Measures.** Precision, recall, and F-score are reported for the extraction task. For the normalization task, we use the “value F1” measure. Here, true positives are only those expressions being (partially) extracted and correctly normalized (value attribute) by the system.

**WikiWars and WikiWarsDE.** In Table 5, evaluation results are shown for the whole WikiWars and WikiWarsDE corpora. Although the changes are mainly relevant for only two of the 22 documents in both corpora, the *value F1* numbers increase by 2.6 and 2.8 percentage points, respectively.

**The AncientTimes corpus.** Table 6 shows the results on our new multilingual AncientTimes corpus. For all languages, the extraction and in particular the normalization of temporal expressions referring to historic dates works very well with HeidelbergTime’s new version.

To demonstrate the value of the post-processing step for disambiguating expressions without explicit BC/AD patterns (c.f. Section 4.3), value F1 scores for HeidelbergTime’s new version without post-processing are also provided in Table 6. For most sub-corpora (languages), the post-processing boosts the normalization performance.

**Other corpora.** To make sure that our modifications do not negatively influence HeidelbergTime’s performance on other kinds of documents (e.g., news articles), we used several publicly available corpora as validation sets to verify our adaptations during the development. HeidelbergTime’s new version achieves high quality extraction and normalization results on the AncientTimes corpus without decreasing the tagging quality on other corpora.<sup>7</sup>

## 6. Summary

In this paper, we presented the AncientTimes corpus containing documents about history from different time periods in eight languages. We studied the challenges of temporal tagging content about history, and we described our adaptation of HeidelbergTime to successfully extract and normalize temporal expressions referring to historic dates.

The AncientTimes corpus as well as the new HeidelbergTime version are publicly available.

<sup>7</sup>HeidelbergTime’s evaluation results on many publicly available corpora are reported at <http://code.google.com/p/heideltime>.

(a) AncientTimes English.								(b) AncientTimes German.									
relaxed				strict				value F1	relaxed				strict				value F1
P	R	F	P	R	F	P	R		F	P	R	F	P	R	F		
1.5	90.4	37.3	52.8	62.4	25.7	36.5	24.3	1.5	91.7	36.3	51.9	55.6	22.1	31.5	16.5		
new	95.2	84.5	89.5	84.4	74.9	79.4	84.3	new	95.3	78.0	85.8	86.6	70.9	78.0	80.4		
w/o p.-p.							83.9	w/o p.-p.							78.6		

  

(c) AncientTimes Spanish.								(d) AncientTimes French.									
relaxed				strict				value F1	relaxed				strict				value F1
P	R	F	P	R	F	P	R		F	P	R	F	P	R	F		
1.5	77.4	60.2	67.7	39.6	30.8	34.7	29.3	1.5	96.8	42.8	59.4	70.6	31.2	43.3	26.8		
new	96.3	85.8	90.7	80.3	71.6	75.7	85.7	new	98.4	84.2	90.7	88.9	76.1	82.0	89.6		
w/o p.-p.							85.2	w/o p.-p.							65.0		

  

(e) AncientTimes Italian.								(f) AncientTimes Dutch.									
relaxed				strict				value F1	relaxed				strict				value F1
P	R	F	P	R	F	P	R		F	P	R	F	P	R	F		
1.5	97.1	43.2	59.8	78.4	34.9	48.2	25.4	1.5*	87.2	60.0	71.1	58.1	40.0	47.4	23.7		
new	97.8	76.0	85.5	86.0	66.8	75.2	81.1	new	94.2	90.4	92.2	81.7	78.4	80.0	88.2		
w/o p.-p.							74.7	w/o p.-p.							88.2		

  

(g) AncientTimes Arabic.								(h) AncientTimes Vietnamese.									
relaxed				strict				value F1	relaxed				strict				value F1
P	R	F	P	R	F	P	R		F	P	R	F	P	R	F		
1.5	70.0	35.0	46.7	32.0	16.0	21.3	25.3	1.5	95.3	52.6	67.8	56.3	31.0	40.0	15.6		
new	93.2	82.0	87.2	84.1	74.0	78.7	81.9	new	97.3	92.2	94.7	85.5	81.0	83.2	89.4		
w/o p.-p.							78.7	w/o p.-p.							89.4		

Table 6: Evaluation results on the AncientTimes corpus comparing HeidelTime 1.5 with HeidelTime’s new version. In addition, we show the value F1 results without the post-processing step for disambiguating expressions without BC/AD patterns (w/o p.-p.). \*Note: A small bug fix in version 1.5 was required (in a rule for Dutch).

## 7. References

- André Bittar, Pascal Amsili, Pascal Denis, and Laurence Danlos. 2011. French TimeBank: an ISO-TimeML Annotated Reference Corpus. In *ACL ’11*, pages 130–134.
- Lisa Ferro, Laurie Gerber, Inderjeet Mani, Beth Sundheim, and George Wilson. 2005. TIDES 2005 Standard for the Annotation of Temporal Expressions. Technical report, The MITRE Corporation.
- B. Magnini, E. Pianta, C. Girardi, M. Negri, L. Romano, M. Speranza, V. Bartalesi Lenzi, and R. Sprugnoli. 2006. I-CAB: the Italian Content Annotation Bank. In *LREC ’06*.
- Pawel Mazur and Robert Dale. 2010. WikiWars: A New Corpus for Research on Temporal Expressions. In *EMNLP ’10*, pages 913–922.
- James Pustejovsky, Patrick Hanks, Roser Sauri, Andrew See, Robert J. Gaizauskas, Andrea Setzer, Dragomir R. Radev, Beth Sundheim, David Day, Lisa Ferro, and Marcia Lazo. 2003. The TIMEBANK Corpus. In *Proceedings of Corpus Linguistics 2003*, pages 647–656.
- James Pustejovsky, Robert Knippen, Jessica Littman, and Roser Sauri. 2005. Temporal and Event Information in Natural Language Text. *Language Resources and Evaluation*, 39(2-3):123–164.
- Roser Sauri and Toni Badia. 2012. Spanish TimeBank 1.0. Technical report, Linguistic Data Consortium (LDC), Philadelphia.
- Jannik Strötgen and Michael Gertz. 2011. WikiWarsDE: A German Corpus of Narratives Annotated with Temporal Expressions. In *GSCL ’11*, pages 129–134.
- Jannik Strötgen and Michael Gertz. 2012. Temporal Tagging on Different Domains: Challenges, Strategies, and Gold Standards. In *LREC ’12*, pages 3746–3753.
- Jannik Strötgen and Michael Gertz. 2013. Multilingual and Cross-domain Temporal Tagging. *Language Resources and Evaluation*, 47(2):269–298.
- Jannik Strötgen, Ayser Armiti, Tran Van Canh, Julian Zell, and Michael Gertz. 2014. Time for More Languages: Temporal Tagging of Arabic, Italian, Spanish, and Vietnamese. *ACM Transactions on Asian Language Information Processing (TALIP)*, 13(1):1–21.
- Naushad UzZaman, Hector Llorens, Leon Derczynski, James Allen, Marc Verhagen, and James Pustejovsky. 2013. SemEval-2013 Task 1: TempEval-3: Evaluating Time Expressions, Events, and Temporal Relations. In *SemEval ’13*, pages 1–9.
- Marc Verhagen, Robert Gaizauskas, Frank Schilder, Mark Hepple, Jessica Moszkowicz, and James Pustejovsky. 2009. The TempEval Challenge: Identifying Temporal Relations in Text. *Language Resources and Evaluation*, 43(2):161–179.
- Marc Verhagen, Roser Sauri, Tommaso Caselli, and James Pustejovsky. 2010. SemEval-2010 Task 13: TempEval-2. In *SemEval ’10*, pages 57–62.