



**Universität
Zürich^{UZH}**

Institut für Computerlinguistik

Automatische Annotation von deutschen
und französischen temporalen Ausdrücken
im Text+Berg-Korpus

Masterarbeit der Philosophischen Fakultät der Universität Zürich
im Studienfach Multilinguale Textanalyse

Referent: Prof. Dr. Martin Volk

Verfasserin:

Katrin Michaela Rettich

Matrikelnummer 08-125-312

Pappelweg 27

3013 Bern

16. Juni 2013

Zusammenfassung

In dieser Masterarbeit im Fach Multilinguale Textanalyse beschäftigen wir uns mit der automatischen Annotation von deutschen und französischen temporalen Ausdrücken im Text+Berg-Korpus. Wir wollen ein Modul für Text+Berg erstellen, in dem alle Informationen über Zeitausdrücke im *TimeML*-Format enthalten sind.

Wir stellen fest, dass es für die beiden Sprachen Deutsch und Französisch bereits bestehende temporale Annotationssysteme gibt und wenden diese zur Extraktion und Auswertung von Zeitausdrücken im Korpus an. Für beide Sprachen zeigen die Evaluationsergebnisse der Annotationen Verbesserungspotenzial auf.

Wir passen das deutsche Annotationssystem an das Text+Berg-Korpus und die spezifische Domäne an und beschäftigen uns mit der Problematik der richtigen Auswertung der Zeitausdrücke. Um die französischen Annotationen zu verbessern, nutzen wir im parallelen Teil des Korpus Vergleichs- und Projektionsmethoden, welche mit den deutschen temporalen Annotationen und den Alignierungsinformationen arbeiten. Mit unseren verschiedenen Ansätzen können wir sowohl die deutschen als auch die französischen temporalen Annotationen verbessern.

Wir können nun Standoff-Dateien mit temporalen Annotationen bereitstellen.

Danksagung

Ich bedanke mich ganz herzlich bei Prof. Dr. Martin Volk für die motivierende Betreuung während der Entwicklung und des Verfassens dieser Arbeit und die mitreissende Begeisterung in der Multilingualen Textanalyse.

Ein Dankeschön geht auch an Jannik Strötgen und Julian Zell von der Universität Heidelberg für die Bereitstellung neuer Module in *HeidelTime* und die Hilfe bei der Installation. Zuletzt bedanke ich mich bei Uwe Rettich und Lena Schwenck für das Korrekturlesen und bei meiner Familie und allen Freunden, die mich immer unterstützt haben.

Inhaltsverzeichnis

Tabellenverzeichnis	v
Abkürzungsverzeichnis	vi
1 Einführung	1
1.1 Motivation	2
1.2 Ziele der Arbeit	3
1.3 Struktureller Aufbau der Arbeit	4
2 Stand der Forschung zur Annotation temporaler Ausdrücke	5
2.1 Annotationsrichtlinien	5
2.1.1 Entwicklung	5
2.1.2 TimeML	7
2.2 Nutzen der temporalen Annotation	10
2.3 Domänenspezifische temporale Annotation	11
3 Annotation deutscher temporaler Ausdrücke	13
3.1 Stand der Forschung zur Annotation deutscher temporaler Ausdrücke	13
3.2 HeidelTime	14
3.2.1 Grundlagen	15
3.2.2 Aufbau und Funktionsweise	16
3.2.3 Zusätzliche Komponenten und Installation	17
3.2.4 Evaluation	18
3.3 HeidelTime als temporaler Tagger für Text+Berg	20
3.3.1 Anpassung	20
3.3.2 Automatische Annotation	29
3.3.3 Ergebnisse und Evaluation für Text+Berg	31

4	Annotation französischer temporaler Ausdrücke	36
4.1	Stand der Forschung zur Annotation französischer temporaler Ausdrücke	36
4.2	Unitex-Graphen und Normalisierung nach Bittar (2009)	38
4.3	Unitex-Graphen als temporaler Tagger für Text+Berg	40
4.3.1	Automatische Annotation	40
4.3.2	Ergebnisse und Evaluation für Text+Berg	41
5	Nutzung der Alignierungen zur Annotation temporaler Ausdrücke	43
5.1	Alignierungen in Text+Berg	44
5.2	Vergleich der deutschen und französischen Annotationen	45
5.3	Projektion der temporalen Annotationen	52
5.4	Kombiniertes Verfahren zur Verbesserung der französischen Annotation	60
6	Konklusion	63
6.1	Zusammenfassung und Ergebnisse	63
6.2	Grenzen und Probleme	66
6.3	Offene Punkte und Ausblick	67
	Bibliographie	68
A	Diagramme	71
B	Zusätzliche Quellenangaben	84
C	Korpusauszug	85

Tabellenverzeichnis

1	TIMEX3: Beispiele	9
2	HeidelTime: Systemkomponenten	16
3	HeidelTime: Evaluation WikiWarsDE und Text+Berg	20
4	HeidelTime: Beispiel für Muster und Normalisierungen	21
5	Anpassung HeidelTime: Evaluation Artikel 3 Jahrbuch 1900	26
6	Verbesserte Disambiguierung: Evaluation Artikel 3 Jahrbuch 1900	29
7	Anzahl temporaler Ausdrücke im Text+Berg-Korpus	31
8	Anpassung HeidelTime: Evaluation für Text+Berg	34
9	Verbesserte Disambiguierung: Evaluation für 800 Sätze	35
10	Tagger für Französisch: Evaluation für Text+Berg	42
11	Vergleich durch Alignierung: Korrespondenz temporale Ausdrücke DE-FR	46
12	Vergleich durch Alignierung: Evaluation französische Annotation Jahrbücher 1960 und 1990	49
13	Vergleich durch Alignierung: Evaluation deutsche Annotation Jahrbücher 1960 und 1990	50
14	Beispiel eines alignierten temporalen Ausdrucks	51
15	Nutzung der Alignierung: Evaluation Verbesserung <code>value</code> der französischen Annotationen	51
16	Projektion durch Alignierung: Evaluation Jahrbuch 1960 und 1990	59
17	Kombinationsverfahren zur französischen Annotation	61

Abkürzungsverzeichnis

AFP	Agence France-Presse, französische Presseagentur
DCT	Document creation time, Zeitpunkt der Dokumenterzeugung
OCR	Optical Character Recognition, Optische Zeichenerkennung
POS	Part of speech, Wortart
UIMA	Unstructured Information Management Architecture

1. Einführung

Das Text+Berg-Korpus (Bubenhofner et al. 2012) entstand im Rahmen des Projekts *Text+Berg digital*¹ an der Universität Zürich. Das Ziel dieses Projekts ist es, alpine Texte des Schweizer Alpenclubs² (SAC) zu erschliessen. Das Korpus umfasst Jahrbücher des SAC von 1864 bis 1923 mit Artikeln in verschiedenen Sprachen, die in Französisch verfassten Publikationen *Echo des Alpes* von 1872 bis 1924, sowie Ausgaben des Magazins *Die Alpen*, welche von 1925 bis 1956 gemischt-sprachlich und von 1957 bis 2011 parallel in deutsch und französisch herausgegeben wurden. Das Korpus umfasst insgesamt 44,6 Millionen Token in 253 Bänden³ und deckt einen Publikationszeitraum von 1864 bis 2011 ab. Die Texte im Korpus behandeln alpine Themen wie Bergbesteigungen, Expeditionen, geographische Beschreibungen, Fauna und Flora der Bergwelt oder die alpine Kultur und Geschichte. Unter den Texttypen im Korpus finden sich Erzählungen, Berichte, Gedichte, Verordnungen oder auch Empfehlungen. Alle Jahrbücher liegen digitalisiert in Form von strukturierten XML-Dateien vor, welche mit linguistischen Metadaten wie Wortart und Lemma annotiert wurden.

Im Text+Berg-Korpus wurden bereits Eigennamen von Bergen, Hütten, Gletschern und Personen automatisch identifiziert und annotiert. Die erkannten Eigennamen sind in einer zum jeweiligen Jahrbuch separaten Datei aufgelistet und enthalten Hinweise zur Identifikation der betreffenden Token im Korpus. Diese automatischen Annotationen von Eigennamen stellen wichtige Informationen für die semantische Auswertung und inhaltliche Aufbereitung des Korpus dar. Ein noch ausstehender Bereich im Text+Berg-Korpus ist die Erkennung von Zeitausdrücken. Die vorliegende Arbeit widmet sich diesem Thema und untersucht die automatische Annotation von deutschen und französischen temporalen Ausdrücken im Text+Berg-Korpus.

Temporale Ausdrücke sind sprachliche Formen, „die auf Zeitpunkte oder -intervalle verweisen [...]“ (Stede 2007, 97). Die Beispiele in 1.1 und 1.2, Satz 343 und Satz 370 aus Artikel 3 des Jahrbuchs von 1900, beinhalten temporale Ausdrücke, welche in

¹<http://www.textberg.ch/>, Stand aller Online-Referenzen im Text: 05.04.2013.

²<http://www.sac-cas.ch/>

³In der vorliegenden Arbeit wurde das Release 147, Version 02 verwendet. Die Bände werden als *Jahrbücher* bezeichnet.

die zwei Hauptgruppen der Zeitpunkte und der Dauer eingeordnet werden können (Endriss et al. 1998). Die temporalen Ausdrücke sind in den Beispielsätzen jeweils unterstrichen.

(1.1) Am 30. Juni, früh um 2 Uhr 40 Min., brachen wir wieder von der Hütte auf.

(1.2) Trotz Nebel und Kälte hielten wir es drei Viertelstunden auf dem Gipfel aus.

Im ersten Beispielsatz finden wir zwei Ausdrücke, welche die Lage eines Zeitpunktes beschreiben. Es handelt sich um die Datumsangabe *30. Juni*, sowie die Uhrzeitangabe *2 Uhr 40 Min.*. Ebenfalls zu dieser Kategorie gehören Adverbien wie *heute*, *bald*, *jetzt* oder *nachmittags*. Der zweite Satz enthält einen temporalen Ausdruck der zeitlichen Dauer. Mit *drei Viertelstunden* wird auf eine Zeitdauer von 45 Minuten verwiesen. Neben den Gruppen der Zeitpunkte und der Dauer gibt es noch weitere Zeitausdrücke, wie der Satz 43 aus Artikel 161 des Jahrbuchs von 2004 in Beispiel 1.3 zeigt.

(1.3) Hans Berger freut sich jeden Frühling, wenn es wieder «obsi gaht» in die Hütte.

Phrasen wie *jeden Frühling* oder auch *zweimal im Monat* beschreiben regelmässig oder mehrmals stattfindende temporale Ausdrücke. In allen Kategorien der Zeitausdrücke finden wir eine Vielzahl an lexikalischer und morphologischer Varianten. Obwohl das Text+Berg-Korpus auch Texte in italienischer, englischer, rhätoromanischer und schweizerdeutscher Sprache umfasst, untersucht diese Arbeit nur die temporalen Ausdrücke in den deutschen und französischen Texten des Korpus.

1.1. Motivation

Viele der Artikel im Text+Berg-Korpus handeln von Bergexpeditionen oder Gipfelbesteigungen. Diese Textsorte umfasst zum grössten Teil die Beschreibungen der Expeditions- bzw. Wanderrouen. So stellt der Autor im Rahmen einer Gipfelbesteigung beispielsweise dar, welche Wege genommen wurden und welche Hütten, Routenpunkte oder geographische Orte die Expedition dabei passierte.

Diese Art von Text bewegte Piotrowski et al. (2010) zur Idee, die Routenbeschreibungen automatisch zu extrahieren, um sie anschliessend kartographisch abbilden zu können. Dafür müssen alle Toponyme, wie Bergnamen oder Hüttennamen, identifiziert und geographisch verortet werden. Die Eigennamenextraktion wurde für das Text+Berg-Korpus bereits zum grössten Teil erfolgreich durchgeführt. Piotrowski et al. (2010) sehen neben der Identifizierung von Toponymen auch die Erkennung

temporalen Ausdrücke als notwendig an, da Expeditionsrouten neben der geographischen auch eine zeitliche Dimension besitzen. Es wäre wünschenswert, in Zukunft durch automatische Extraktion aus den Texten des Korpus ableiten zu können, in welchem Jahr beispielsweise ein bestimmter Gipfel bestiegen wurde, um wie viel Uhr die Expedition dabei bestimmte Hütten passierte und wie lange die gesamte Besteigung dauerte.

Die Motivation dieser Arbeit besteht darin, mit der Annotation von temporalen Ausdrücken einen Beitrag zur automatischen Extraktion der im Text+Berg-Korpus beschriebenen alpinen Routen zu leisten.

1.2. Ziele der Arbeit

Diese Arbeit beschäftigt sich mit der automatischen Annotation von deutschen und französischen temporalen Ausdrücken im Text+Berg-Korpus. Folgende Ziele gelten als grundlegend für die gesamte Arbeit:

- Die Bereitstellung eines Moduls für Text+Berg mit Informationen über deutsche und französische temporale Ausdrücke im Korpus.
- Eine qualitativ hochwertige temporale Annotation, d.h. möglichst viele verschiedene Zeitausdrücke möglichst genau erkennen.

Um am Ende für jedes Jahrbuch separate Dateien mit Metadaten zur Information über Zeitausdrücke im Korpus zu erhalten, definieren wir die folgenden Punkte als Zwischenziele:

- Ein automatisches temporales Annotationssystem für das Deutsche untersuchen, anwenden sowie für das Text+Berg-Korpus anpassen und evaluieren.
- Ein automatisches temporales Annotationssystem für das Französische untersuchen, anwenden und evaluieren.
- Die Nutzung der Alignierungen zur Annotation von temporalen Ausdrücken im parallelen, deutsch-französischen Teil von Text+Berg untersuchen.

1.3. Struktureller Aufbau der Arbeit

In Kapitel 2 betrachten wir die theoretischen Grundlagen zur automatischen Annotation von temporalen Ausdrücken. Neben der Untersuchung der Annotationsrichtlinie *TimeML* werden wir den Nutzen der temporalen Annotation und die domänenspezifische temporale Annotation genauer beleuchten. Daran anschliessend beschäftigt sich Kapitel 3 mit der automatischen Annotation von deutschen temporalen Ausdrücken. Nach der Darstellung der bisherigen Forschung für das Deutsche auf diesem Gebiet untersuchen wir den temporalen Tagger *HeidelTime*, passen ihn für das Text+Berg-Korpus an und evaluieren seine Annotationen im Text+Berg-Korpus. Kapitel 4 widmet sich daraufhin der automatischen Annotation von französischen temporalen Ausdrücken. Dazu stellen wir bisherige Forschungsergebnisse vor, untersuchen ein System zur Annotation und nutzen es für französische Texte im Text+Berg-Korpus. Auch für dieses System präsentieren wir anschliessend entsprechende Evaluationsdaten. In Kapitel 5 nutzen wir die bestehenden Wortalignierungen des parallelen Teils in Text+Berg für die automatische temporale Annotation. Einerseits untersuchen wir, ob durch die Alignierungen temporale Ausdrücke überprüft werden können. Auf der anderen Seite soll ein System mithilfe der Alignierungen die Projektion von temporalen Ausdrücken ermöglichen. Anschliessend evaluieren wir die Resultate der Überprüfung und der Projektion. Im letzten Teil des Kapitels 5 betrachten wir die Kombination der beiden Verfahren. Kapitel 6 fasst schliesslich die Ergebnisse der Arbeit zusammen, beschreibt Grenzen und Probleme der Anwendung der automatischen temporalen Annotation und gibt einen Ausblick auf mögliche weitere Untersuchungen auf diesem Gebiet.

2. Stand der Forschung zur Annotation temporaler Ausdrücke

Die Extraktion und semantische Auswertung temporaler Ausdrücke ist bereits seit über einem Jahrzehnt ein Bereich der computerlinguistischen Forschung. In den letzten zehn Jahren wurden zahlreiche Forschungsansätze zu diesem Thema publiziert. Darunter finden sich Publikationen zu Annotationsstandards, zu Systemen der automatischen Extraktion sowie zu Korpora, welche für Trainings- und Evaluationszwecke entwickelt wurden.

Dieses Kapitel fasst die Entwicklungen der Annotation von temporalen Ausdrücken hin zu einem Annotationsstandard zusammen und erklärt anschliessend die Grundlagen des Standards *TimeML*, welcher in dieser Arbeit verwendet wurde. Im zweiten Teil des Kapitels untersuchen wir die Anwendungsmöglichkeiten temporaler Annotationen. Einige Überlegungen zur Relevanz der Textdomäne beschliessen dieses Kapitel. Weitere Angaben über die Entwicklung von sprachspezifischen Systemen zur automatischen Annotation befinden sich in den jeweiligen Kapiteln zur Annotation deutscher und französischer Texte.

2.1. Annotationsrichtlinien

2.1.1. Entwicklung

Die Annotation bestimmter Elemente in einem Text ist ein wichtiger Teil der Analyse von Korpora. Einheitlich annotierte Texte mit Metadaten lassen Vergleiche verschiedener Korpora zu, können als Trainingsdaten für Algorithmen dienen, sind zur objektiven Evaluation verschiedener Systeme geeignet und können Sprachformen und Ausdrücke auffinden, welche in der linguistischen Theorie nicht berücksichtigt wurden (Mani et al. 2005). Die Annotation temporaler Ausdrücke umfasst im ersten Schritt die Markierung von Wörtern, Wortsequenzen oder Phrasen, die sich auf die Zeit beziehen. Das können beispielsweise Ausdrücke für eine Uhrzeit oder eine Zeit-

periode sein. Die temporale Annotation ist besonders wichtig, um die Aktionen oder Handlungen eines Satzes (*events*) in einem kalendarischen Zeitrahmen zu situieren oder sie in Relation zu anderen Aktionen und Handlungen zu setzen (Mani et al. 2005).

Neben der Extraktion der temporalen Ausdrücke ist vor allem die semantische Interpretation wichtig, welche später in den Metadaten der Annotation festgehalten wird. Diese Interpretation wird im Folgenden als Normalisierung bezeichnet, da die Semantik jedes gefundenen temporalen Ausdrucks in eine einheitliche, normierte Form überführt wird. Für diesen Schritt ist die Unterscheidung der temporalen Ausdrücke ein wichtiges Kriterium. Dabei ist ihre klare Abgrenzung zu Aktionen (EVENT) wichtig, die ebenfalls Teil einer temporalen Annotation sein können. Im Rahmen dieser Arbeit werden wir Annotationen des Typs EVENT nicht berücksichtigen.

Nach Schilder und Habel (2001) lassen sich zeitliche Ausdrücke in drei Kategorien einordnen. Unter Ausdrücken mit „expliziter Referenz“ verstehen die Autoren Datums- oder Zeitangaben, welche sich explizit auf einen kalendarischen Zeitpunkt oder eine präzise Uhrzeit beziehen. Darunter fallen Extraktionen wie *25. August 1897* oder *5 Uhr*. Ihre Normalisierung lässt sich direkt aus der Extraktion ableiten. Ausdrücke mit „indexikalischer Referenz“ hingegen können nur mithilfe eines gegebenen Referenzzeitpunktes normalisiert werden. Extraktionen wie *im letzten Jahr* lassen sich ableiten, wenn eine Referenz zum jeweiligen aktuellen zeitlichen Ausdruck (in diesem Fall das aktuelle Jahr), gegeben ist. Bei Extraktionen mit „vager Referenz“ handelt es sich um zeitliche Ausdrücke, denen in der Normalisierungsphase kein präziser Zeitpunkt zugewiesen werden kann. Das können Extraktionen wie *nachmittags* oder *etwas vor 8 Uhr* sein. Im nächsten Kapitel, im Rahmen des Standards TimeML, erklären wir genauer, welche Metadaten für solche Ausdrücke generiert werden.

Die Annotation temporaler Ausdrücke war anfänglich ein Teilgebiet der Eigennamen-Erkennung (*Named Entity tagging*), welches auf der *DARPA Message Understanding Conference* (MUC) behandelt wurde (Mani et al. 2005). Im Rahmen der MUC-6 im Jahr 1995 wurden Systeme vorgestellt, die lediglich temporale Ausdrücke mit expliziter Referenz aus Zeitungsartikeln extrahierten. Die darauf folgende MUC-7 im Jahr 1998 machte es sich hingegen zur Aufgabe, zusätzlich auch Ausdrücke mit indexikalischer Referenz zu finden. Diese beiden Konferenzen widmeten sich jedoch nur der Extraktion von temporalen Ausdrücken. Um die temporale Annotation nutzbar zu machen, müssen die extrahierten Ausdrücke zwingend interpretiert werden, um vor allem jene mit indexikalischer Referenz korrekt zu disambiguieren. Die ersten Ansätze, den vollständigen kalendarischen Wert eines temporalen Ausdrucks als Attribut in einer normierten Form zu generieren, wurden mit den TIMEX2-Annotationsrichtlinien (Ferro et al. 2005) eingeführt. Dieser XML-basierte

Standard bildete wiederum die Basis für die **TIMEX3**-Annotation von temporalen Ausdrücken innerhalb der TimeML-Richtlinien (u.a. in Pustejovsky et al. 2005). Im folgenden Kapitel beschäftigen wir uns mit dem Aufbau, den Tags und Attributen sowie mit dem Nutzen von TimeML.

2.1.2. TimeML

TimeML ist eine Markup-Sprache, welche temporale Ausdrücke und Aktionen (**EVENT**) markiert und die Beziehungen zwischen diesen beiden Elementen darstellt. Das Ziel dieses XML-basierten Standards ist die Repräsentation von zeitlichen Relationen in einem Text (Pustejovsky et al. 2005). Die Grundlage für eine solche Repräsentation bildet dabei die Annotation von temporalen Ausdrücken. Sobald ein solcher Ausdruck manuell oder automatisch extrahiert wird, erhält er als direkte Kennzeichnung im Fliesstext ein **TIMEX3**-Tag. Dabei unterscheiden wir zwischen vier Typen von temporalen Ausdrücken, deren Bezeichnung jeweils im Attribut **type** festgehalten wird. Ein Ausdruck des Typs **TIME** ist eine Zeitangabe, die sich auf eine Tages- oder Uhrzeit bezieht. Der Typ **DATE** hingegen beschreibt temporale Ausdrücke, welche ein kalendarisches Datum darstellen. Um diese beiden Typen zu unterscheiden, betrachten wir die Granularität des Ausdrucks. Ist diese kleiner als ein Kalendertag (24 Stunden), so handelt es sich um den Typ **TIME**. Anderenfalls erhält der Ausdruck den Typ **DATE** (Pustejovsky et al. 2005). Neben diesen beiden gibt es zusätzlich die Typen **DURATION** und **SET**. Ein Ausdruck gilt als **DURATION**, wenn er eine Zeitperiode anzeigt. Als **SET** werden Wörter oder Wortsequenzen bezeichnet, die mehrmals oder regelmässig stattfindende zeitliche Ausdrücke darstellen.

Neben dem Attribut **type** erhält jedes **TIMEX3**-Tag das Attribut **value**, in welchem der Wert bzw. die semantische Interpretation des temporalen Ausdrucks gespeichert ist. Alle Werte erhalten dabei eine einheitliche Form anhand der Empfehlung ISO 8601¹ für Zeitangaben und Datumsformate. Für **DATE**-Typen ist dies das Format **JJJJ-MM-TT**, **TIME**-Typen schreiben wir im Format **JJJJ-MM-TTThh:mm**, wobei das **T** zwischen Datumsangabe und Uhrzeit auf eine Zeitangabe hinweist. Das Datum jeder Uhrzeitangabe muss bei der manuellen Annotation oder durch ein automatisches Annotationssystem anhand von vorhergehenden Referenzen disambiguiert werden. Kann für einen temporalen Ausdruck der komplette normierte Wert nicht durch ein Referenzdatum disambiguiert werden, so erhält der Wert an dieser Stelle den Platzhalter **X**. Beim Typ **DURATION** zeigt ein **P** an, dass es sich um eine Zeitperiode handelt. Wenn dieser ein **T** folgt, liegt eine Zeitspanne vor, die kleiner als ein Kalendertag (24 Stunden) ist. Dem **P** bzw. **PT** folgt jeweils die Angabe der Zeitdauer, z. B. **3H**

¹http://de.wikipedia.org/wiki/ISO_8601

für *3 Stunden*. Bei SET-Ausdrücken geben wir, soweit verfügbar, im `value`-Attribut ebenfalls die Zeitspanne an, in welcher der sich wiederholende temporale Ausdruck stattfindet. Zusätzlich benötigt ein SET-Ausdruck eines der beiden Attribute `quant` oder `freq`. Im Attribut `quant` speichern wir einen im temporalen Ausdruck enthaltenen Quantifikator. So vergeben wir beispielsweise für den Ausdruck *jede Woche* im Quantifikator-Attribut die Kennzeichnung `EVERY`. Zusätzlich finden wir im SET-Ausdruck eine Häufigkeitsangabe, welche durch das Attribut `freq` dargestellt wird. Für den Ausdruck *2 Tage in der Woche* würde das Attribut `freq` den Wert `2D` (`D` = days, Tage) erhalten.

Allen extrahierten temporalen Ausdrücken weisen wir zusätzlich eine automatische ID im Attribut `tid` zu. Damit kann der Ausdruck an anderer Stelle in der Annotation erwähnt werden, um beispielsweise ein `EVENT` zeitlich zu situieren (Pustejovsky et al. 2005). Die bis hierhin beschriebenen Attribute sind obligatorisch. Es gibt noch einige weitere, fakultative Attribute, die hauptsächlich von den `TIMEX2`-Richtlinien übernommen wurden. Diese Richtlinien bieten einen Standard zur Annotation von temporalen Ausdrücken, während `TimeML` diese Funktion um die Annotation von Aktionen (`EVENT`) und Relationen erweitert. Durch die Übernahme einiger Attribute von `TIMEX2` in `TIMEX3`-Tags sind sich die beiden Annotation sehr ähnlich und Übersetzungen von `TIMEX2` in `TIMEX3` lassen sich meist gut bewältigen (Strötgen und Gertz 2012b). Zu den wichtigsten fakultativen Attributen zählt `mod`, welches die Semantik eines im temporalen Ausdruck enthaltenen Modifikators angibt (Ferro et al. 2005). Bei Uhrzeitangaben können wir durch `BEFORE` oder `AFTER` Ausdrücke wie *vor acht Uhr* in `mod` genauer spezifizieren. Für Uhrzeitangaben und Zeitdauern kann in diesem Attribut auch ein `APPROX` gespeichert sein, wenn im Ausdruck eine gewisse Ungenauigkeit enthalten ist, wie im Beispiel *ungefähr zwei Stunden*. Zusätzlich können wir mit `START` und `END` Ausdrücke wie *Anfang des Jahres* genauer beschreiben. Tabelle 1 zeigt Beispiele für die verschiedenen Typen der `TIMEX3`-Annotation mit ihren Attributen. Alle vier Beispielsätze stammen aus dem Jahrbuch von 1934 des Text+Berg-Korpus.

Neben den `TIMEX3`-Tags können wir mit `TimeML` auch Tags des Typs `EVENT` vergeben, welche meist prädikative Ausdrücke der Handlung oder Aktion darstellen. Ausserdem ist es möglich, mit `SIGNAL`-Tags Funktionswörter zu kennzeichnen, um deren Rolle in der Beziehung zwischen einem `EVENT` und einem temporalen Ausdruck anzuzeigen (Pustejovsky et al. 2005). Den letztendlich wichtigsten Teil von `TimeML` stellen die Verbindungs-Tags `TLINK`, `SLINK` und `ALINK` dar, in welchen die Information gespeichert ist, wie Events zeitlich verankert sind. So können wir Events satz- wie auch diskursintern in Relation zueinander setzen. Wenn alle `TimeML`-Tags und -Attribute korrekt vergeben werden und der Text auf XML-konforme Weise darge-

Typ	Annotation
TIME	Mit der Tageshelle betreten wir <TIMEX3 mod="BEFORE" tid="t57283" type="TIME" value="1932-07-25T05:00» vor 5 Uhr </TIMEX3> den Lendarey-gletscher . (Artikel 60, Satz 103)
DATE	<TIMEX3 mod="START" tid="t58028" type="DATE" value="1934-06» Anfang Juni 1934 </TIMEX3> machten wir einen Versuch . (Artikel 96, Satz 8)
DURATION	Nach <TIMEX3 mod="APPROX" tid="t57202" type="DURATION" value="PT5H» etwa fünfstündigem </TIMEX3> Marsche sattelten wir die Tiere ab und schlugen unser Zelt auf . (Artikel 59, Satz 343)
SET	die neue Pflanze würde also ca. 8 mm über der alten sitzen , und wenn sich das <TIMEX3 quant="EVERY" tid="t56655" type="SET" value="P1Y» jedes Jahr </TIMEX3> wiederholt , so müsste die Pflanze schliesslich aus der Erde herauswachsen . (Artikel 42, Satz 134)

Tabelle 1.: Beispiele der TIMEX3-Annotation. Vier Sätze aus dem Jahrbuch 1934.

stellt wird², ist die Verarbeitung des Textes mit einem XML-Parser möglich. Obwohl der Aufbau der Annotation, die Funktionen der Tags und die jeweiligen Attribute im TimeML-Standard klar und übersichtlich dargestellt sind, fehlt der Richtlinie doch eine genaue Charakterisierung bzw. Typisierung der temporalen Ausdrücke. Dieses Fehlen an Kriterien für temporale Ausdrücke im Kontext wird von Ehrmann und Hagège (2009) klar kritisiert. Sie schlagen daraufhin drei Kriterien zur Charakterisierung von französischen temporalen Ausdrücken vor. Im Rahmen der vorliegenden Arbeit wurde sprachübergreifend vor allem das erste Kriterium als grundlegend zur Erkennung von temporalen Ausdrücken anerkannt. So muss ein Wort, eine Wortsequenz oder eine Phrase, welche ein TIMEX3-Tag erhält, auf die Fragen *Wann?*, *Wie lange?* oder *Wie häufig?* eine Antwort geben können. Einen weiteren Schwachpunkt von TimeML sehen Ehrmann und Hagège (2009) in den ungenauen Angaben zur Segmentierung von temporalen Ausdrücken. Während TimeML vorschreibt, Präpositionen vor temporalen Ausdrücken nicht als Teil eines TIMEX3-Tags zu annotieren, müssen sie nach den Kriterien von Ehrmann und Hagège (2009) ebenfalls annotiert werden. Die vorliegende Arbeit hält sich in diesem Fall mit

²Es muss zumindest ein XML-Header vorhanden sein.

wenigen Ausnahmen an den TimeML-Standard. Bei der Segmentierung von komplexen temporalen Ausdrücken versuchen wir in dieser Arbeit, Extraktionen wenn möglich zu einem `TIMEX3`-Tag zu kombinieren, sollten sie direkt aufeinander folgen. Wenn eine Uhrzeitangabe einem Datum folgt, annotieren wir diese jedoch einzeln. Dies geschieht ebenfalls mit Ausdrücken, die durch eine Koordination verbunden sind.

Diese Arbeit betrachtet nur die Annotation von Zeitausdrücken im Text+Berg-Korpus und beschäftigt sich demnach nur mit der Basis-Annotation aus TimeML. Alle generierten Annotationen wurden TimeML-konform durchgeführt, weshalb ein Hinzufügen von weiteren Tags wie `EVENT` oder `SIGNAL` möglich wäre.

2.2. Nutzen der temporalen Annotation

Nach der genauen Beschreibung der Annotation temporaler Ausdrücke anhand von TimeML beleuchten wir an dieser Stelle auch die mögliche Verwendung einer solchen Annotation. In erster Linie dient ein TimeML-konform annotiertes Korpus der weiteren Forschung auf diesem Gebiet in Form von Trainings- und Evaluationsdaten, welche vor allem für die Entwicklung neuer automatischer Annotationssysteme wichtig sind. Vielen Publikationen (u.a. Schilder und Habel 2001, Muller und Tanier 2004, Pustejovsky et al. 2005, Mani et al. 2005, Schilder et al. 2007, Parent et al. 2008, Bittar 2008, Strötgen und Gertz 2012b) nennen Question-Answering-Systeme als einen Bereich, in dem temporale Annotationen genutzt werden können. Zusätzlich können annotierte Zeitausdrücke in der allgemeinen Informationsextraktion oder zur Zusammenfassung von Texten verwendet werden.

In einem Question-Answering-System für das Text+Berg-Korpus könnten wir durch die Extraktion und korrekte Interpretation von temporalen Ausdrücken Fragen wie *Wie lange dauerte die Expedition im Mont-Blanc-Gebiet im Jahr 1900?* beantworten. Im Rahmen der Informationsextraktion ist die Annotation temporaler Ausdrücke wichtig, um Relationen zwischen anderen Eigennamen herzustellen. Im Korpus Text+Berg könnten wir so beispielsweise über das Jahr der Besteigung Assoziationen zwischen Personennamen und Bergname herstellen. Im Bereich der Textzusammenfassung sehen Mani et al. (2005) eine Chance, mehrere Dokumente mit redundanten Zeitangaben zusammenzufassen und chronologisch zu ordnen, was vor allem für Nachrichten-Publikationen, klinische Berichte oder Gerichtsverfahren genutzt werden kann. Im Fall des Text+Berg-Korpus wäre hier vor allem die Zusammenfassung der zeitlichen Abfolge innerhalb eines Artikels zu einer Bergexpedition interessant, da wir so beispielsweise Dauer von Wanderungen, Pausen oder einzel-

nen Wegabschnitten verschiedener Expeditionen vergleichen könnten. Ausserdem könnten wir alle in einem Jahrbuch beschriebenen Expeditionen chronologisch ordnen.

Vazov (2001) nennt als zusätzlichen Nutzen der temporalen Annotation die Verbesserung von Parsing-Technologien. Da temporale Ausdrücke häufig syntaktische Konstruktionen darstellten, könnte die Identifikation ihrer Grenzen die Aufgabe des Parsers vereinfachen.

Den bedeutsamsten Nutzen der temporalen Annotation im Text+Berg-Korpus bildet die Motivation dieser Arbeit, wie wir sie bereits in Kapitel 1.1 darstellten: die zeitliche Situierung von alpinen Routen der im Korpus beschriebenen Expeditionen (Piotrowski et al. 2010). Diese mögliche Verwendung und die oben beschriebenen weiteren Verwendungsmöglichkeiten zeigen, dass ein Korpus durch annotierte Zeitausdrücke bedeutend an Mehrwert gewinnt.

2.3. Domänenspezifische temporale Annotation

In Kapitel 2.1.1 wurde ein kurzer Einblick in die Entwicklung der Annotation von temporalen Ausdrücken gegeben. Es ist wichtig zu erwähnen, dass die ersten Korpora, welche zu Trainings- und Evaluationszwecken verwendet wurden, aus Zeitungsartikeln bestanden und wir sie somit der Domäne der *news* bzw. Presse-Publikationen zuordnen können. Strötgen und Gertz (2012a) verweisen darauf, dass die Domäne des Korpus, welches es zu annotieren gilt, ein wichtiger Punkt bei der Entwicklung automatischer Annotationssysteme ist. Die Qualität der Extraktion und semantischen Interpretation durch temporale Tagger ist stark abhängig von der Domäne des Korpus.

Je nach Domänen-Typ müssen wir eine andere Strategie zur Normalisierung bzw. zur Disambiguierung von ungenügend spezifizierten temporalen Ausdrücken, welche keine expliziten Angaben zur Normalisierung enthalten, anwenden. Wir müssen in solchen Fällen ein Referenzdatum und die korrekte Relation des temporalen Ausdrucks zum Referenzdatum bestimmen. In Texten der Domäne *news* ist die DCT (Document creation time) meist das Referenzdatum. Auch für Texte aus klinischen Berichten ist die DCT die Referenz für implizite Ausdrücke, wobei diese in der genannten Domäne sehr selten vorkommen. Für umgangssprachliche Texte, wie SMS-Nachrichten, kann das Sendedatum als Referenz verwendet werden, wie Strötgen und Gertz (2012a) vorschlagen. In narrativen Texten, in denen keine DCT verfügbar ist, muss nach Strötgen und Gertz (2012a) das Referenzdatum im Text gefunden werden. Sie betonen, dass die Disambiguierung impliziter temporaler Ausdrücke in

dieser Domäne die meisten Schwierigkeiten aufweist.

Diese Erkenntnisse zeigen uns, dass wir ein automatisches System zur temporalen Annotation auf Grund der verschiedenen Normalisierungsstrategien domänenspezifisch anpassen müssen. Wir können das Text+Berg-Korpus mit der Domäne der alpinen Texte nicht direkt in eine der beschriebenen Textdomänen einordnen. Da keine DCT verfügbar ist und die Texte in Text+Berg oftmals einen erzählenden Charakter haben, werden wir dieses Korpus in die Domäne der narrativen Texte einordnen. Kapitel 3.3 beschäftigt sich mit der Normalisierung in Text+Berg.

Strötgen und Gertz (2012a) nennen auch die unterschiedlichen Arten von temporalen Ausdrücken in verschiedenen Domänen als einen Grund, domänenspezifische temporale Tagger zu entwickeln. Sie fanden an eigens erstellten Korpora heraus, dass SET-Typen sehr viel häufiger in klinischen Berichten vorkommen, wohingegen sie Uhrzeitangaben sehr häufig in SMS finden. Auch die Extraktionsphase eines Taggers muss demnach auf die Textdomäne angepasst sein. Aus diesem Grund werden wir die Extraktionskomponenten des deutschen Taggers ebenfalls überarbeiten.

3. Annotation deutscher temporaler Ausdrücke

3.1. Stand der Forschung zur Annotation deutscher temporaler Ausdrücke

Die Entwicklung von automatischen temporalen Annotationssystemen konzentrierte sich in ihren Anfängen im Wesentlichen auf das Englische und einige wenige andere Sprachen. Für das Deutsche zählen Schilder und Habel (2001) zu den Ersten, die sich mit einem System zur automatischen Extraktion von temporalen Ausdrücken und ihrer Interpretation beschäftigten. Ihre Arbeit stellt die Grundlage für die Klassifikation von temporalen Ausdrücken im vorhergehenden Kapitel dar.

Schilder und Habel (2001) verwenden zum Training ihres Taggers ein Korpus aus deutschen Zeitungsartikeln. Ihr System zur Extraktion besteht aus einer Kaskade an endlichen Transduktoren (*Finite State Transducers*). Als Input für die endlichen Transduktoren dienen Informationen über das POS-Tag (Wortart) und das Lemma eines Wortes. Anhand dieser Information und mithilfe einer integrierten morphologischen Analyse kann der Transduktor mit 15 Zuständen und 61 Transitionen temporale Ausdrücke erkennen, zu welchen unter anderem Wochentage wie *Diens- tag* oder Zeitangaben wie *Monat* zählen. Zusätzlich extrahiert das System die DCT, Aktionen (**EVENT**) und nominale Beschreibungen von Aktionen.

Die gefundenen Ausdrücke werden noch nicht mithilfe einer Annotationsrichtlinie wie TimeML annotiert. Schilder und Habel (2001) fügen vielmehr einfache **CHUNK**-Tags in den Text ein, um die temporalen Ausdrücke zu markieren. Diese Tags enthalten bereits Attribute vergleichbar mit denen der **TIMEX3**-Tags. So wird beispielsweise im Attribut **time** das Datum immer in gleicher, abgekürzter Form festgehalten. Das Attribut **sem** erhält die semantische Interpretation des temporalen Ausdrucks, welche andere temporale Ausdrücke als Referenz zur Disambiguierung beinhalten kann. Die Disambiguierung der Semantik eines ungenügend spezifizierten temporalen Ausdrucks findet bei Schilder und Habel (2001) während des Prozesses der Generierung

der Output-Datei statt.

Zur Evaluation des Systems annotierten die Autoren das Korpus manuell, um eine korrekte Referenz bzw. einen Gold Standard zu erhalten. Im Bereich der Mustererkennung und -annotation werden zur Evaluation von Systemen die Masse Precision und Recall¹ berechnet. Unter allen vom automatischen Annotationssystem extrahierten Mustern wird ermittelt, wie viele davon tatsächlich relevante bzw. korrekte Muster sind, die auch im Gold Standard annotiert wurden. Dieser Wert wird als *Precision* bezeichnet. *Recall* hingegen definiert, wie viele aller relevanter Muster aus dem Gold Standard vom System gefunden werden.

Für die Extraktion der temporalen Ausdrücke, die später als TIMEX-Ausdrücke annotiert werden sollten und welche Schilder und Habel (2001) als „einfache temporale Ausdrücke“ bezeichnen, erreichen sie den Precision-Wert von 92,11 und den Recall-Wert von 94,09. Für die semantische Analyse der temporalen Ausdrücke bzw. die Generierung der temporalen Information, erreicht das System Precision- und Recall-Werte von 84,49.

Obwohl dieser erste Ansatz zur automatischen Erkennung temporaler Ausdrücke in deutschen Texten einen Tagger beschreibt, der noch keine TIMEX2/3-Tags generiert oder im Rahmen einer Annotationsrichtlinie steht, so versuchte er dennoch, temporale Ausdrücke einheitlich zu annotieren und deren Semantik anhand von temporalen Referenzen zu generieren. Vor allem die Klassifikation der temporalen Ausdrücke durch Schilder und Habel (2001) stellte eine der Grundlagen für den temporalen Tagger *HeidelTime* dar (Strötgen und Gertz 2012b), dessen Verwendung als Tagger für deutsche Texte im nächsten Kapitel beschrieben wird.

3.2. HeidelTime

Als ein Ziel dieser Arbeit legten wir fest, temporale Ausdrücke in den deutschen Texten des Korpus Text+Berg zu erkennen und zu normalisieren. Dazu verwenden wir den bereits bestehenden temporalen Tagger *HeidelTime* (Strötgen und Gertz 2012b). Im nun folgenden Kapitel stellen wir zuerst die Grundlagen und Funktionsweise dieses Werkzeugs für die temporale Annotation vor. Anschliessend präsentieren wir die Ergebnisse der Evaluation des Taggers für das Text+Berg-Korpus und beschreiben die Anpassung von *HeidelTime* für dieses Korpus. Nach einer Erläuterung zu den Schritten der automatischen temporalen Annotation von Text+Berg folgen die

¹Die Bezeichnung *Precision* beschreibt den deutschen Term *Präzision*, während das Mass *Recall* auf deutsch als *Ausbeute* bezeichnet wird. Da deutsche Literatur auf dem Gebiet der Computerlinguistik häufig die englischen Bezeichnungen verwendet, werden auch in dieser Arbeit häufig die Terme *Precision* und *Recall* benutzt.

Ergebnisse der abschliessenden Evaluation des angepassten Taggers für Text+Berg.

3.2.1. Grundlagen

HeidelTime ist ein regelbasiertes System zur temporalen Annotation von Texten. Es ist in der Version 1.2, welche wir für diese Arbeit verwenden, für die Sprachen Englisch, Deutsch und Holländisch verfügbar². Das System erkennt temporale Ausdrücke in einem Text und annotiert diesen Text mit den entsprechenden TIMEX3-Tags anhand der TimeML-Annotationsrichtlinien. Strötgen und Gertz (2012b) entschieden sich gezielt für einen regelbasierten Ansatz zur Konstruktion eines temporalen Taggers. Sie begründen diese Wahl mit dem Verweis auf die begrenzte Menge an temporalen Ausdrücken in einer Sprache sowie mit dem Argument der Normalisierung von temporalen Ausdrücken, welche zwingend Regeln benötigt. Ausserdem sehen sie in der regelbasierten Struktur die Möglichkeit, weitere Sprachen in das System zu integrieren ohne auf ein annotiertes Korpus zurückgreifen zu müssen. Letztendlich spielt die Erweiterbarkeit der zugrundeliegenden sprachspezifischen Ressourcen zur Erkennung und Normalisierung von temporalen Ausdrücken eine grosse Rolle bei der Entscheidung für ein regelbasiertes System und ist auch für die Anwendung von HeidelTime innerhalb dieser Arbeit ein essentieller Teil.

HeidelTime zeichnet sich durch Modularität und Multilingualität aus. Das Programm ist in einzelne Komponenten gegliedert: Neben dem Quellcode existieren für die zu extrahierenden Muster sowie für deren Normalisierung anhand der ISO-8601-Richtlinie eigene Dateien. Zusätzliche Dateien mit Regeln zur Art der Erkennung und Normalisierung von temporalen Ausdrücken vervollständigen das Programm. Dieses vom Quellcode separat angelegte Ressourcenpaket muss für jede zu annotierende Sprache definiert werden. Auch für die Disambiguierungsphase kann HeidelTime durch spezifische Module erweitert werden, um eine möglichst korrekte Disambiguierung von indirekten temporalen Ausdrücken in verschiedenen Domänen zu garantieren.

Der folgende Abschnitt zeigt den detaillierten Aufbau von HeidelTime und das Vorgehen des Systems bei der Annotation von temporalen Ausdrücken.

²In der folgenden Version 1.3 (April 2013) sind zusätzlich die Sprachen Arabisch, Vietnamesisch, Spanisch und Italienisch verfügbar.

3.2.2. Aufbau und Funktionsweise

Der wichtigste Bestandteil des Systems HeidelbergTime sind die sprachspezifischen Ressourcen. Sie setzen sich aus Dateien zur Musterdefinition, zur Normalisierung der jeweiligen Muster und aus Regeln zur Erkennung von temporalen Ausdrücken zusammen. Die Musterdefinition geschieht durch Auflisten von regulären Ausdrücken, die viele Varianten eines zu annotierenden temporalen Ausdrucks darstellen. In einer Musterdatei sind alle Zeitausdrücke einer bestimmten Kategorie zusammengefasst (z. B. Monate). Die Normalisierungsdateien enthalten für jede in den Mustern definierte Variante einen ISO-8601-konformen Wert, der später im TIMEX3-Tag `value` eingesetzt wird. Die Regeln legen letztendlich fest, welche Muster gefunden werden sollen und welche Normalisierung für die gefundenen temporalen Ausdrücke angewandt werden soll. Dort sind auch Beschränkungen der Muster auf beispielsweise eine bestimmte Wortart aufgelistet. Tabelle 2 fasst die einzelnen Systemkomponenten von HeidelbergTime und deren Funktionen zusammen.

Komponente	Funktion
Algorithmus	Programmcode Annotation in vier Phasen
Ressourcen	
Muster	Definition der regulären Ausdrücke <i>Beispiel:</i> [Dd]reiviertel Stunden?
Normalisierungen	Definition der ISO-8601-konformen Werte für jede Variante der Muster <i>Beispiel:</i> "Dreiviertel Stunde", "45M"
Regeln	Verknüpfung von Muster und Normalisierung Festlegung der Attribute der TIMEX3-Tags (<code>type</code> , <code>value</code> , <code>mod</code> etc.) Regeln zur Löschung von ungültigen temporalen Ausdrücken <i>Beispiele befinden sich in Kapitel 3.3.1 in den Darstellungen 3.1 und 3.2.</i>

Tabelle 2.: Die Systemkomponenten von HeidelbergTime mit dazugehörigen Funktionen (Strötgen und Gertz 2012b).

Der Algorithmus von HeidelbergTime definiert den Programmcode des Systems. Er erwartet als Eingabe einen Text, der bereits durch zusätzliche Komponenten vorbereitet wurde (siehe Kapitel 3.2.3). In der ersten Phase des algorithmischen Teils extrahiert das System die temporalen Ausdrücke anhand der Musterdateien. Anschließend weist das System den Ausdrücken Normalisierungen zu, die auch ungenügend spezifizierte Werte enthalten können. Wenn ein Wert im Text nicht genügend spezifiziert ist, wird versucht, den Bezug zu einem Referenz Ausdruck herzustellen. Hier ist die festgelegte Domäne für das Dokument entscheidend. Handelt es sich um ein

Dokument aus der Domäne *news*, so ist die Referenz der Zeitpunkt der Dokumenterzeugung (DCT). Im Fall des Text+Berg-Korpus wurde die Domäne auf *narratives* festgelegt. Für diese Domäne ist die DCT nicht relevant. Der Referenzausdruck ist hier immer der letzte, voll spezifizierte Ausdruck des betreffenden Typs (TIME, DATE, DURATION oder SET) im Text.

In der dritten Phase des Algorithmus disambiguiert das System die in der Normalisierungsphase als mehrdeutig oder zu wenig spezifiziert klassifizierten Ausdrücke. Wenn zwei extrahierte Ausdrücke mit überlappenden Token vorliegen, so wird bei unterschiedlicher Länge der längere Ausdruck favorisiert, während bei gleicher Länge der erste Ausdruck ein TIMEX3-Tag erhält. Im zweiten Schritt der Disambiguierung sucht das System für alle ungenügend spezifizierten Ausdrücke anhand der festgelegten Domäne eine entsprechende Referenz, damit die extrahierten Ausdrücke vollständige Normalisierungen erhalten können. Als letztes eliminiert der Algorithmus ungültige temporale Ausdrücke. Für diesen Schritt sind in den Ressourcen ebenfalls Regeln definiert.

3.2.3. Zusätzliche Komponenten und Installation

Für diese Arbeit verwenden wir das HeidelbergTime-Kit, welches neben dem Tagger HeidelbergTime noch weitere Komponenten benutzt und diese in UIMA zu einem System kombiniert. UIMA ist eine Architektur zur Verwaltung unstrukturierter Informationen³, welche im Rahmen von Programmierprojekten zur Wissensextraktion genutzt werden kann. Der Tagger HeidelbergTime ist ein Java-basiertes System, das als Input einen Text erwartet, der bereits in Sätze und Tokens aufgesplittet und mit POS-Tags versehen wurde. Daher kann HeidelbergTime nicht allein als Tagger eines Volltextes fungieren. Es müssen zusätzliche Komponenten vorgeschaltet werden⁴. Für diese Arbeit verwenden wir, in Anlehnung an Strötgen und Gertz (2012b), die UIMA-kompatiblen Komponenten des DKPro-Systems (Gurevych et al. 2007), welche aus einem Satz-Trenner und dem TreeTagger als POS-Tagger bestehen. Diese Komponenten werden in der UIMA-Pipeline vor dem HeidelbergTime-Algorithmus ausgeführt, um den Text zu präparieren. Anschliessend übersetzt eine weitere Komponente alle vorgenommenen Annotationen durch DKPro in ein HeidelbergTime-kompatibles Format. Erst dann kann der Tagger HeidelbergTime den Text mit TIMEX3-Tags annotieren. Das Text+Berg-Korpus liegt bereits tokenisiert und in Sätze aufgesplittet vor. Auch POS-Tag-Annotationen sind schon vorhanden. Aus Kompatibilitätsgründen wurden dennoch

³<http://de.wikipedia.org/wiki/UIMA>

⁴Es existiert eine Standalone-Version von HeidelbergTime, die ohne zusätzliche Komponenten auf einem Volltext als Tagger angewendet werden kann. Jedoch können in dieser Version die Ressourcen nicht verändert werden.

die devertikalisierten Texte aus dem Korpus durch den Satz-Trenner und den POS-Tagger vorverarbeitet. Die Komponente, welche die Satztrennungen und POS-Tag-Annotationen für HeidelTime vorbereitet, ist auf die UIMA-Programme abgestimmt und eine Abänderung für die Kompatibilität mit den Text+Berg-Annotationen sahen wir als zu aufwendig an. Ein Tokenisierer musste nicht mehr vorgeschaltet werden.

Alle beschriebenen Komponenten, einschliesslich des HeidelTime-Systems, gehören zur *Analysis Engine* von UIMA. Zuvor liest ein *Collection Reader* die zu annotierenden Texte aus einem festgelegten Ordner ein. Nach der Analyse muss mit einem *Consumer* festgelegt werden, wie die generierten TIMEX3-Tags ausgegeben werden. Für diese Arbeit benutzen wir hier ein Modul, welches ein XML-Dokument generiert und dabei die TIMEX3-Tags direkt in den Volltext integriert, ohne dessen Format zu verändern. Dieses Modul ist nicht Teil des HeidelTime-Kits 1.2, sondern wurde von Jannick Strötgen⁵ zur Verfügung gestellt.

Zur Ausführung des HeidelTime-Kits verwenden wir eine Linux-Plattform. Weitere Informationen zur Installation und Nutzung sind auf der Projekt-Seite von HeidelTime⁶ verfügbar.

3.2.4. Evaluation

HeidelTime wurde im Rahmen des TempEval-2-Wettbewerbs als bester temporaler Tagger für das Englische bewertet. TempEval-2 war Teil des Workshops für semantische Evaluation SemEval-2010 (Verhagen et al. 2010), der sich alle ein bis zwei Jahre mit der Evaluation von semantischen Analysesystemen der Computerlinguistik beschäftigt. Die Aufgabenstellung des Task A in TempEval-2 bestand darin, temporale Ausdrücke in einem Text anhand der TimeML-Richtlinien zu ermitteln und den Wert für `type` und `value` zu bestimmen. Für die Extraktion entwickelten Strötgen und Gertz (2010) in HeidelTime ein Precision- und ein Recall-optimiertes Set an Extraktionsregeln. Damit erreichten sie den Precision-Wert von 90,0 und den Recall-Wert von 91,0, dem eingesetzten Optimierungsmodell entsprechend. Auch die `value`-Werte konnten mit HeidelTime zu 85,0 korrekt generiert werden (Strötgen und Gertz 2010).

Die deutschen Ressourcen für HeidelTime entstanden im Rahmen der automatischen temporalen Annotation des Korpus *WikiWarsDE*, welches durch Strötgen und Gertz (2011) erstellt wurde. Dieses Korpus lehnt sich an das englische Korpus WikiWars an, welches aus Wikipedia-Artikeln über historische Kriege besteht. WikiWarsDE

⁵jannik.stroetgen@informatik.uni-heidelberg.de

⁶<https://code.google.com/p/heideltime/>

enthält diese entsprechend gleichen Artikel in deutscher Sprache. Da mit dem Korpus ein *Gold Standard* geschaffen werden sollte, dienten die temporalen Annotationen durch HeidelbergTime als Basis für eine manuelle Annotation bzw. Korrektur. Für dieses Korpus führten Strötgen und Gertz (2011) die neue Domäne *narratives* in HeidelbergTime ein, welche eine andere Disambiguierung vornimmt als in *news*-Dokumenten (siehe Kapitel 3.2.2). Bei der Evaluation der Annotationsergebnisse durch HeidelbergTime berechneten sie Precision- und Recall-Werte für verschiedene Kategorien, die zwischen Extraktion und Normalisierung der **TIMEX3**-Tags unterscheiden. Bei der Extraktion ermitteln Strötgen und Gertz (2011) Ergebnisse für Ausdrücke, die dem Gold Standard exakt entsprechen (*strikt*) sowie für Ausdrücke, bei denen zumindest ein Token des Ausdrucks auch in der entsprechenden Extraktion im Gold Standard vorkommt (*partiell*). Um die Normalisierung zu messen, untersuchten sie, ob das `value`-Attribut den richtigen, dem Gold Standard entsprechenden Wert besitzt. Die Ergebnisse von Strötgen und Gertz (2011) für das WikiWarsDE-Korpus sind in Tabelle 3 aufgelistet. In Anlehnung an die Evaluation von HeidelbergTime in Strötgen und Gertz (2011) evaluieren wir auch die Annotation von temporalen Ausdrücken im Text+Berg-Korpus mit HeidelbergTime, bevor wir Anpassungen des temporalen Taggers für dieses Korpus vornehmen.

Zur Evaluation annotierten wir den Artikel 3 aus dem Jahrbuch 1900 aus Text+Berg manuell mit dem Oxygen XML Editor⁷ mit **TIMEX3**-Tags, um einen Gold Standard zu erstellen. Dabei achteten wir darauf, die Annotation möglichst TimeML-konform durchzuführen und alle Zeitausdrücke zu erfassen. Durch unser bereits bestehende Wissen über die automatische Annotation von temporalen Ausdrücken kann die Objektivität dieser manuellen Annotation streng gesehen in Frage gestellt werden. Jedoch bemühten wir uns, die manuellen Annotationen unabhängig von temporalen Taggern durchzuführen. Der annotierte Artikel 3 besteht aus 409 Sätzen. Die Umwandlung der Jahrbücher zur Generierung eines Volltextes aus den XML-annotierten Artikeln beschreiben wir in Kapitel 3.3.2. Anschliessend annotierten wir den gleichen Artikel mit HeidelbergTime (Strötgen und Gertz 2012b) automatisch mit **TIMEX3**-Tags. Tabelle 3 zeigt die resultierenden Evaluationsergebnisse im Vergleich zu den Ergebnissen für das deutsche Korpus WikiWarsDE von Strötgen und Gertz (2011).

Allgemein erzielt der temporale Tagger für die Annotation der alpinen Texte aus Text+Berg deutlich schlechtere Ergebnisse als in Strötgen und Gertz (2011). Einzig die Präzision in der partiellen Extraktions-Kategorie ist vergleichbar mit dem Precision-Wert von WikiWarsDE. Dies zeigt klar die Stärke von HeidelbergTime, denn die Anzahl an Wörtern, die keine temporalen Ausdrücke darstellen und trotzdem ein **TIMEX3**-Tag erhielten, ist sehr gering. Trotzdem bewerten wir die Precision- und

⁷<http://www.oxygenxml.com/>, Demo-Version

	partiell			strikt			value		
	P	R	F	P	R	F	P	R	F
WikiWarsDE	98,5	85,0	91,3	92,6	79,9	85,8	87,0	87,0	87,0
Text+Berg	96,4	49,1	65,1	67,5	34,4	45,6	60,2	30,7	40,7

Tabelle 3.: Evaluation HeidelTime: WikiWarsDE und Text+Berg, Artikel 3 Jahrbuch 1900. P = Precision, R = Recall, F = F-Mass.

vor allem die Recall-Werte in allen anderen Kategorien als deutlich zu tief, um eine zufriedenstellende automatische Annotation des gesamten Korpus durchzuführen. Das F-Mass (*f-measure*), welches den harmonischen Durchschnitt zwischen Precision und Recall darstellt (Koehn 2010, 224), liegt für die strikte Extraktion temporaler Ausdrücke im Text+Berg-Korpus rund 40 Prozentpunkte tiefer als im WikiWarsDE-Korpus. Für die *value*-Extraktion gehen die Werte sogar noch weiter auseinander. Aus diesem Grund sehen wir ein deutliches Potenzial, die automatische Annotation mit HeidelTime für deutsche Texte im Text+Berg-Korpus zu verbessern und passen die deutschen Ressourcen von HeidelTime im Folgenden an. Wir wollen die Extraktion von Zeitausdrücken und ihre semantische Interpretation qualitativ verbessern. Im nächsten Kapitel beschreiben wir den Prozess des Anpassens, sowie das Ergebnis in Form einer zweiten Evaluation des angepassten HeidelTime-Systems.

3.3. HeidelTime als temporaler Tagger für Text+Berg

3.3.1. Anpassung

Um für das Text+Berg-Korpus bessere Ergebnisse des automatischen temporalen Taggers HeidelTime zu erzielen, passen wir dessen deutsche Ressourcen für dieses Korpus an. Der erste Schritt der Anpassung besteht darin, anhand des manuell annotierten Artikel 3 aus dem Jahrbuch 1900 alle temporalen Ausdrücke herauszufiltern, welche noch nicht durch HeidelTime extrahiert wurden. Diese temporalen Ausdrücke speichern wir einerseits in den Musterdateien als zusätzliche Extraktionsmuster ab. Andererseits definieren wir für diese temporalen Ausdrücke neue Normalisierungen. Tabelle 4 zeigt in der linken Spalte eine Reihe von Mustern aus der Datei *resources_repattern_reDurationWord.txt*. Diese ist in den HeidelTime-Ressourcen für die deut-

Muster	Normalisierung
[Dd]reiviertelstündige?[mnrs]?	"dreiviertelstündig", "45M" "dreiviertelstündige", "45M" "dreiviertelstündigem", "45M" "dreiviertelstündigen", "45M" "dreiviertelstündiger", "45M" "dreiviertelstündiges", "45M" "Dreiviertelstündig", "45M" "Dreiviertelstündige", "45M" "Dreiviertelstündigem", "45M" "Dreiviertelstündigen", "45M" "Dreiviertelstündiger", "45M" "Dreiviertelstündiges", "45M"
[Dd]reiviertel Stunden?	"Dreiviertel Stunde", "45M" "dreiviertel Stunde", "45M" "Dreiviertel Stunden", "45M" "dreiviertel Stunden", "45M"
[Dd]rei [Vv]iertel Stunden?	"Drei viertel Stunde", "45M" "Drei viertel Stunden", "45M" "Drei Viertel Stunde", "45M" "Drei Viertel Stunden", "45M" "drei Viertel Stunde", "45M" "drei Viertel Stunden", "45M" "drei viertel Stunde", "45M" "drei viertel Stunden", "45M"

Tabelle 4.: Auszug aus *reDurationWord* mit den dazugehörigen Normalisierungen aus *normDurationWord*.

sche Sprache abgelegt (*resources/german/repattern*) und beschreibt ein Muster für eine Regel, die im Anpassungsschritt hinzugefügt wurde. Es handelt sich um reguläre Ausdrücke zur Erkennung von Wörtern, Wortsequenzen oder Phrasen, die eine Zeitdauer anzeigen. Eine Musterdatei enthält immer möglichst viele temporale Ausdrücke der gleichen Art. Tabelle 4 zeigt Muster, die Sprecher des Deutschen benutzen, um eine Zeitdauer von 45 Minuten auszudrücken. Die regulären Ausdrücke beschreiben möglichst viele Varianten bzw. Token der zu extrahierenden Muster. Je nach Genus und Kasus des darauffolgenden Wortes können adjektivische Ausdrücke wie *dreiviertelstündig* eine unterschiedliche Endung aufweisen. Auch unter-

schiedliche Trennungen der Wortsequenzen wie in den Beispielen *dreiviertel Stunde* und *drei Viertel Stunden* sind in der deutschen Schriftsprache möglich. Für jeden Ausdruck und dessen Varianten ist in der Normalisierungsdatei ein ISO-konformer Wert festgelegt. Obwohl alle Ausdrücke aus Tabelle 4 den normalisierten Wert **45M** (45 Minuten) zugewiesen bekommen, müssen wir die möglichen, durch die regulären Ausdrücke definierten Varianten einzeln in der dazugehörigen Normalisierungsdatei *resources_normalization_normDurationWord.txt* auflisten. Nur so können wir allen Varianten für die Zeitdauer von 45 Minuten in der Annotationsphase einen konkreten normalisierten Wert zuweisen. Die Musterdatei mit Wörtern und Wortsequenzen für Zeitperioden wurde ergänzt durch andere Ausdrücke wie beispielsweise *halbstündig*, *Viertelstündchen* oder *fünf Viertel Stunden*.

Um die Musterdefinitionen und Normalisierungen zusammenzuführen, definieren wir in der Regeldatei des entsprechenden Typs eine Regel. Für jede Sprache sind die Regeln unterteilt in die vier Typen von temporalen Ausdrücken: **TIME**, **DATE**, **DURATION** und **SET**. Die folgenden Beispiele 3.1 und 3.2 zeigen die zwei Regeln, mit denen eine Zeitdauer, wie in den Beispielen aus Tabelle 4, extrahiert und normalisiert wird. Beide Regeln stammen aus der Regel-Datei *resources_rules_durationrules.txt*.

- ```
(3.1) RULENAME="duration_r3t",
 EXTRACTION="%reDurationWord",
 NORM_VALUE="PT%normDurationWord(group(1))"

(3.2) RULENAME="duration_r3u",
 EXTRACTION="%reApproximate %reDurationWord",
 NORM_VALUE="PT%normDurationWord(group(2))",
 NORM_MOD="APPROX"
```

Regel `duration_r3t` (3.1) beschreibt die einfachste Form einer Regel zur Annotation temporaler Ausdrücke. Im Extraktionsaufruf (`EXTRACTION`) wird definiert, welcher reguläre Ausdruck gefunden werden soll bzw. welcher Ausdruck in das `TIMEX3`-Tag eingeschlossen werden soll. Mit `%` kann auf eine Musterdatei verwiesen werden. Alle in dieser Datei aufgelisteten regulären Ausdrücke kommen bei der Annotation in Frage. Der Extraktionsaufruf kann neben Verweisen auf Musterdateien noch zusätzliche reguläre Ausdrücke enthalten.

Im Normalisierungsausdruck (`NORM_VALUE`) wird entschieden, welcher Teil der Extraktionsphase wie normalisiert wird. So legt Regel `duration_r3t` fest, dass die erste definierte Gruppe regulärer Ausdrücke normalisiert werden soll (`group(1)`). Der normalisierte Wert, der dem jeweiligen Ausdruck zugeschrieben wird, ist in der Normalisierungsdatei *normDurationWord* gespeichert, welche wie in der Extraktionsphase durch ein `%`-Zeichen aufgerufen werden kann. Da es sich bei diesem Beispiel

um eine Zeitspanne handelt, wird der Normalisierung nach ISO-8601-Richtlinie ein P für *period* vorausgestellt. Um anzuzeigen, dass es sich zusätzlich um eine Zeitspanne mit Uhrzeitanteil handelt, wird nach dem P ebenfalls ein T für *time* eingesetzt<sup>8</sup>.

Mit Regel `duration_r3u` (3.2) können die gleichen temporalen Ausdrücke wie in Regel `duration_r3t` gefunden werden. Zusätzlich steht hier vor dem temporalen Ausdruck ein weiterer Ausdruck, definiert durch reguläre Ausdrücke in der Musterdatei `%reApproximate`. Diese beschreiben Adjektive wie z. B. *ungefähr* oder Partikel wie z. B. *etwa*, welche den folgenden temporalen Ausdruck als ungenau kennzeichnen. Die Regel findet Ausdrücke wie *ungefähr halbstündig*. Der Ausdruck, welcher die Ungenauigkeit anzeigt, wird nicht durch einen separaten Dateiaufruf normalisiert. Vielmehr wird für diesen Ausdruck im `mod`-Attribut des `TIMEX3`-Tags ein Wert gespeichert. Dort wird durch `APPROX` angezeigt, dass es sich bei der entsprechenden Zeitdauer nur um eine ungefähre Angabe handelt. Da in der Extraktionsphase der Regel an der ersten Stelle nun der Verweis auf die Musterdatei mit Adjektiven und Partikeln für Ungenauigkeit steht, kommt erst an zweiter Stelle der Verweis auf die regulären Ausdrücke für eine Zeitdauer. Daher wird die Zeitdauer in der Normalisierungsphase als `group(2)` referenziert.

HeidelTime füllt mit diesen Regeln die Attribute der `TIMEX3`-Tags. Das `type`-Attribut wird automatisch festgelegt, da die Regel einer der Regeldateien `DATE`, `TIME`, `DURATION` oder `SET` entstammt. Der vom `TIMEX3`-Tag zu umschliessende Text wird in der Extraktionsphase festgelegt. Der Wert (`value`) wird, mithilfe der Normalisierungsdateien, in der `NORM.VALUE`-Phase definiert. Diese drei beschriebenen Elemente sind die obligatorischen Teile für ein `TIMEX3`-Tag. Fakultativ sieht HeidelTime noch die Attribute `mod` vor, sowie `freq` und `quant` für `SET`-Ausdrücke.

Mit den Anpassungen von HeidelTime verfolgen wir das Ziel, im Artikel 3 aus dem Jahrbuch 1900 alle temporalen Ausdrücke zu extrahieren und sie exakt zu normalisieren. Durch die Anpassung des Systems auf einen gewählten Artikel wollen wir auch die Annotationsqualität der anderen Artikel im Korpus verbessern. Im folgenden Abschnitt stellen wir eine Auswahl an wesentlichen Änderungen der deutschen Ressourcen in HeidelTime vor. Allgemein achten wir bei der Anpassung darauf, dass bei neuen sowie bei bestehenden definierten Mustern immer möglichst viele Schreibvarianten der temporalen Ausdrücke gefunden werden können. Ausserdem legen wir für einige bereits definierte temporale Ausdrücke zusätzlich separate Regeln zur Erkennung von Ungenauigkeitsangaben fest, um dies im Attribut `mod` zu speichern (*ungefähr 4 Uhr, etwa drei Stunden etc.*).

---

<sup>8</sup>[http://de.wikipedia.org/wiki/ISO\\_8601](http://de.wikipedia.org/wiki/ISO_8601)

**Feiertage** Wir erweiterten die Muster zur Erkennung von Feiertagen um Feiertage in der Schweiz<sup>9</sup>. Dabei wurden Feiertage mit festem Datum sowie Feiertage mit variablem Datum (z. B. dritter Montag im November) berücksichtigt. Diese Anpassung orientierte sich an den bereits bestehenden deutschen Ressourcen und ist unabhängig von der Analyse des Trainingsartikels.

**Jahreszahlen** Die Erkennung von Jahreszahlen war in den originalen Ressourcen von HeidelTime so definiert, dass alle Zahlen zwischen 1000 und 2999 als Jahreszahl erkannt wurden. Im Text+Berg-Korpus gibt es ausser Jahreszahlen jedoch noch viele weitere numerische vierstellige Zahlen in diesem Bereich, wie z. B. Höhenangaben oder Punkangaben auf Wanderrouen. Aus diesem Grund schränkten wir den Bereich an vierstelligen Zahlen für Jahre ein. Als untere Grenze wählten wir das Jahr 1100, da in den Artikeln gelegentlich geschichtliche Angaben oder Erzählungen über vergangene Ereignisse enthalten sind. Die obere Grenze definieren wir abhängig vom Jahr des Jahrbuchs, in dem der zu annotierende Artikel enthalten ist. So gilt z. B. für alle Jahrbücher zwischen 1864 und 1880 eine obere Jahresgrenze von 1889. Alle Zahlen darüber werden nicht als Jahreszahl erkannt. Zusätzlich erweiterten wir die negativen Regeln, welche festgelegte reguläre Ausdrücke explizit nicht als temporalen Ausdruck behandeln. Numerische vierstellige Angaben gefolgt von Einheiten wie *Franken* oder *Meter* werden nun als mögliche Jahreszahl ausgeschlossen.

**Uhrzeit: Format** Ein spezielles Uhrzeitformat, welches häufig im Text+Berg-Korpus auftaucht, jedoch nicht in HeidelTime berücksichtigt war, ist das Format wie im Beispiel *5 Uhr 15 Min./Minuten*. Zusätzlich passten wir die Uhrzeitangaben so an, dass sie auch aus Zahlwörtern bestehen können, wie z. B. *fünf Uhr*. Bei Ausdrücken wie *vor sechs Uhr* oder *nach 8 Uhr* wird nun neu im `mod`-Attribut der Wert `BEFORE` bzw. `AFTER` gespeichert.

**Uhrzeit: Tageszeit** Bei der Normalisierung von Uhrzeiten legten wir fest, dass diese im Normalfall immer als Uhrzeit vor 12 Uhr normalisiert werden. So wird die Angabe *5 Uhr* als `05:00` im `value`-Attribut gespeichert. Nur wenn vor oder nach der Uhrzeitangabe eine Tageszeitangabe wie *nachmittags* oder *morgens* steht, kann gezielt normalisiert werden als `17:00` oder `05:00`. Ausserdem legten wir spezielle Normalisierungen für Uhrzeitangaben wie *halb drei* fest, damit diese als `02:30` normalisiert werden können.

**Zeitdauer** Neben Ausdrücken für Zeitdauern wie in Tabelle 4, fügten wir Muster für Ausdrücke wie z. B. *eine halbe Stunde* oder für ungenaue Angaben

---

<sup>9</sup>Angaben zu den Quellen befinden sich im Anhang B.

wie z. B. *einige Tage* oder *etliche Zeit* hinzu. Zusätzlich definierten wir einen Wert für das `mod`-Attribut, welcher nicht den TimeML-Richtlinien entstammt. Ausdrücke wie *die letzten drei Monate* oder *die nächsten vier Tage* erhalten nun neu in `mod` den Wert `PAST_REF` oder `FUTURE_REF`. Diese zeigen an, dass es sich um eine Zeitspanne handelt, die bereits vergangen ist oder in der Zukunft liegt.

**Wochentage** Wir haben einige Änderungen an der Art des Umgangs mit Wochentagen durchgeführt. Wird ein gross geschriebener Wochentag gefunden (z. B. *Sonntags*), so wird dieser als Datum behandelt. Wenn dem Wochentag eine Tageszeitangabe folgt wie z. B. *Sonntag morgens*, so wird dieser temporale Ausdruck als Zeitangabe (`TIME`) behandelt. Ist der Wochentag klein geschrieben (*sonntags*), gilt er als `SET` ( mit der Bedeutung *jeden Sonntag*).

**Datum** Eine Datumsangabe, welche nur aus einem Kalendertag besteht, wie z. B. *am 24.*, wird nun extrahiert und zusammen mit dem letzten im Text gefundenen Monat und Jahr als vollständiges Datum normalisiert. Zusätzlich fügten wir eine Extraktion für Angaben wie *vor 12 Tagen* hinzu, welche nun erkannt und normalisiert werden können.

**Tageszeit** Alle Komposita, welche eine Tageszeitangabe enthalten (z. B. *Morgensessen*, *Abendspaziergang*), werden nun erkannt und es wird ihnen in der Normalisierungsphase das aktuelle Referenzdatum sowie die betreffende Angabe zur Tageszeit (`MO` = morning, `AF` = afternoon, `EV` = evening, `NI` = night) hinzugefügt. Für die Tageszeit *Mittag* wird immer die Uhrzeit 12:00 eingesetzt.

Diese Liste stellt die grundlegenden und aufwändigsten Anpassungen dar. Zusätzlich erweiterten wir, wenn nötig, die bestehenden deutschen Ressourcen in HeidelTime um reguläre Ausdrücke und Normalisierungen und generierten neue Regeln für die Hinzufügung von fakultativen Attributen.

Nach der Anpassung von HeidelTime auf Artikel 3 aus dem Jahrbuch 1900 präsentieren wir eine zweite Evaluation des Taggers, welche die Verbesserungen aufzeigt, die mit der Modifikation der deutschen Ressourcen erzielt wurde. Tabelle 5 zeigt die Precision- und Recall-Werte für die angepasste Version von HeidelTime für den Artikel 3 des Jahrbuchs von 1900. Obwohl das Evaluationskorpus gleichzeitig das Entwicklungskorpus zur Anpassung von HeidelTime darstellt, liegen die Werte für die strikte Extraktion nicht über 90 Prozentpunkten. Im Fall der Präzision stellen wir uns die Frage, warum nicht alle gefundenen `TIMEX3`-Tags relevante temporale Ausdrücke sind. Dies liegt am Auftreten von generischen Zeitausdrücken im Artikel. Vor allem Jahreszeitangaben wie *Sommer* sind oft nicht als zeitlich verortbarer temporaler Ausdruck gemeint, sondern beschreiben die Jahreszeit im Allgemeinen ohne



|                                 | partiell |      |      | strikt |      |      | value |      |      |
|---------------------------------|----------|------|------|--------|------|------|-------|------|------|
|                                 | P        | R    | F    | P      | R    | F    | P     | R    | F    |
| <b>WikiWarsDE</b>               | 98,5     | 85,0 | 91,3 | 92,6   | 79,9 | 85,8 | 87,0  | 87,0 | 87,0 |
| <b>Text+Berg vor Anpassung</b>  | 96,4     | 49,1 | 65,1 | 67,5   | 34,4 | 45,6 | 60,2  | 30,7 | 40,7 |
| <b>Text+Berg nach Anpassung</b> | 95,2     | 97,5 | 96,3 | 86,8   | 89,0 | 87,9 | 53,9  | 55,2 | 54,5 |

Tabelle 5.: Evaluation nach Anpassung von HeidelTime. Artikel 3 aus Jahrbuch 1900.

zeitliche Referenz. Wir annotierten diese Ausdrücke im Gold Standard nicht, da eine TimeML-konforme Annotation immer eine Referenz zu einer Jahreszahl vorsieht. Für *Sommer* wird beispielsweise der Wert 1895-SU generiert. Diese generischen Zeitangaben werden vom temporalen Tagger fälschlicherweise als TIMEX3-Tag erkannt, wobei eine Differenzierung zwischen Jahreszeitangaben mit zeitlicher Referenz und generischen Ausdrücken sehr schwierig umzusetzen ist.

Betrachten wir die Recall-Werte, so ist nicht sofort klar, warum trotz Anpassung nicht alle relevanten temporalen Ausdrücke gefunden wurden. Dies liegt zum grössten Teil an den OCR-Fehlern im Korpus. Diese betreffen z. B. Uhrzeiten, wie in der Angabe *5<sup>1</sup>/<sub>2</sub> Uhr*, welche manuell annotiert wurde mit dem Wert 05:30, jedoch vom temporalen Tagger als 02:00 erkannt wird. Das Problem der existierenden OCR-Fehler erklärt auch den Unterschied von etwa 10 Prozentpunkten zwischen den Werten für die partielle und für die strikte Extraktion. Ein temporaler Ausdruck mit OCR-Fehler kann meist zwar nicht korrekt, jedoch teilweise erkannt werden. Die allgemeine Qualität der automatischen temporalen Annotation im Trainingsartikel verbesserte sich jedoch deutlich, was durch das F-Mass repräsentiert wird. Dieses konnte im Bereich der partiellen Extraktion um ca. 30 Prozentpunkte, für die strikte Extraktion sogar um mehr als 40 Prozentpunkte angehoben werden. Wir konnten somit die Qualität der Extraktionen deutlich verbessern.

Die Anpassung von HeidelTime für den Artikel 3 aus dem Jahrbuch 1900 erzielt im Bereich der Recall-Werte bessere Ergebnisse als bei der Evaluation des Systems durch Strötgen und Gertz (2011). Ihre Recall-Werte werden für die strikte Extraktion durch das angepasste System um etwa 10 Prozentpunkte übertroffen, für die partielle Extraktion sogar um etwa 12 Prozentpunkte. Die Precision-Werte des angepassten Systems liegen für die Extraktion etwas unter den Werten für WikiWarsDE (Tabelle 3).

Der deutliche Schwachpunkt des angepassten Systems ist die Normalisierung, sichtbar im Wert des F-Masses für `value`, welchen wir nur um rund 14 Prozentpunkte

verbessern konnten. Im Text+Berg-Korpus sind verhältnismässig viele ungenügend spezifizierte temporale Ausdrücke enthalten, welche disambiguiert werden müssen. Die Verwendung des letzten vollwertigen temporalen Ausdrucks im Text als Referenz für Disambiguierungen stellt oft eine gute Wahl für alpine Texte dar. So handeln Texte in diesem Korpus oft von Bergbesteigungen, Wanderungen oder alpinen Exkursionen, die chronologisch beschrieben werden. Immer wieder kommen in den Artikeln jedoch Textpassagen vor, die eine Beschreibung der Gegebenheiten in der beobachteten Natur beinhalten können wie z. B. Beschreibungen von Berglandschaften oder geographische Angaben. Diese Textpassagen enthalten oftmals Erläuterungen über vergangene Ereignisse bzw. Analepsen, wie z. B. frühere Exkursionen am selben Ort oder Entstehungsgeschichten bestimmter Unterkünfte. Ein Beispiel dafür zeigen wir in 3.3. Der Kontext des Textbeispiels ist eine Exkursion im Jahr 1897. Im Textausschnitt wird die Hütte *de d’Orny* beschrieben. Diese Passage folgt innerhalb eines Exkursionsberichts kurz nach der Ankunft in dieser Hütte.

- (3.3) Die de d’ Orny , im Jahr 1893 neu erstellt von der Section des Diablerets , unter deren Obhut sie steht , ist ein behagliches Heim .  
 Nur erlebten wir gleich bei der Ankunft eine herbe Enttäuschung .  
 Nach dem Reglement sollte die Hütte mit Holz versorgt sein .  
 Aber wie so oft im Leben stimmte auch hier das Reglement mit der Thatsache nicht .  
 Vom letzten Jahr war nichts mehr übrig geblieben und dies Jahr noch nichts heraufgeschafft worden .  
*(Jahrbuch 1900, Artikel 3, Sätze 86-90. Anhang C zeigt die Textpassage in der Korpusdatei.)*

Im Textbeispiel 3.3 sind die temporalen Ausdrücke, welche von HeidelbergTime extrahiert werden, unterstrichen. Da die Exkursion im Jahr 1897 stattfindet, ist bei der Zuweisung der Normalisierungen klar, dass dem Ausdruck *letztes Jahr* der Wert 1896 und dem Ausdruck *dies Jahr* der Wert 1897 zugeschrieben werden muss. Der temporale Tagger nimmt zur Disambiguierung dieser ungenügend spezifizierten Jahresangaben des Typs DATE jedoch die letzte vollständig spezifizierte Referenz des gleichen Typs im Text. In diesem Fall ist das der Ausdruck *Jahr 1893*, und somit normalisiert HeidelbergTime den Ausdruck *letzten Jahr* mit 1892 und *dies Jahr* mit 1893. Diese Einschübe im Text sind zur Generierung der richtigen value-Werte sehr problematisch und ein Grund dafür, warum die Evaluationsergebnisse im Vergleich zu Strötgen und Gertz (2011) mehr als 30 Prozentpunkte niedriger ausfallen.

Um die value-Werte zu verbessern, entwickelten wir ein Python-Skript, welches die von HeidelbergTime generierten Normalisierungen überarbeitet und gegebenenfalls die temporalen Ausdrücke neu disambiguiert. Dieses Skript wurde nach der An-

notation durch HeidelbergTime auf die annotierten XML-Dateien angewendet. Dabei versuchten wir gezielt, die Datumsangaben von ungenügend spezifizierten temporalen Ausdrücken in DATE- und TIME-Tags erneut zu disambiguieren.

Durch das Parsen der XML-Struktur mit dem Python-Modul *lxml* greifen wir auf die TIMEX3-Tags des Typs DATE und TIME zu. Wenn das Attribut `value` der Tags der Form JJJJ-MM-TT entspricht bzw. eine vollständige Datumsangabe enthält, untersuchen wir diese Tags genauer. Zur Referenzierung von ungenügend spezifizierten Ausdrücken geht das Programm genauso vor wie HeidelbergTime und nimmt immer das letzte vollständige Datum im Text als Referenz. Jedoch wird nun am Anfang jedes Artikels ein neues Referenzdatum festgelegt. Dem Programm *disam.py* liegt die Annahme zugrunde, dass Erzählungen von Bergexpeditionen eine chronologische Abfolge haben. Sobald Angaben vorkommen, die in der Vergangenheit liegen (in Bezug zum vorhergehenden Datum), ziehen wir diese nicht als Referenz in Betracht. Dies gilt ebenfalls für Daten, die mehr als einen Kalendertag nach dem aktuellen Referenzdatum bzw. in der Zukunft liegen. Bei der Zuweisung von Referenzdaten lassen wir lediglich Schritte von +1 Kalendertag zu. Wenn im Text eines Tags weder die Jahreszahl noch die Monats- oder Tagesangabe des Wertes als Ziffer oder Zahlwort vorkommen, gilt der temporale Ausdruck als ungenügend spezifiziert und wir weisen ihm das neu berechnete Referenzdatum zu. Ist das aktuelle Referenzdatum beispielsweise der 30. Juni und die Disambiguierung für den Ausdruck *vorgestern* errechnet den 28. Juni, so gilt das neu errechnete Datum nicht als Referenz, sondern wir behalten die Referenz 30.06 bei. Wird jedoch der Ausdruck *am nächsten Tag* als 1. Juli errechnet, so gilt dieses Datum gleichzeitig als neue Referenz für folgende temporale Ausdrücke. Ein Datum wird ebenfalls als neue Referenz anerkannt, wenn Monats- und Tagesangabe explizit im Text vorkommen.

Auch für das Adverb *heute*, für welches HeidelbergTime lediglich den Wert `PRESENT_REF` vergibt, ermitteln wir ein Datum. Zusätzlich berechnen wir die Datumsangaben für alle Wochentage neu. HeidelbergTime geht davon aus, dass das Datum eines Wochentags immer vor dem aktuellen Referenzdatum liegen muss, bzw. in der Vergangenheit. Die Untersuchung des Artikels 3 aus dem Jahrbuch 1900 zeigte jedoch, dass in allen Fällen, in denen Wochentage vorkamen, das Datum der Wochentage in der Zukunft lag bzw. ein Wochentag in der Zukunft gemeint war. Das Skript korrigiert die Datumsangaben für Wochentage, sodass sie immer in der Zukunft liegen.

Nach der Überarbeitung der Werte (`value`) der temporalen Ausdrücke in Artikel 3 aus dem Jahrbuch 1900 präsentieren wir eine erneute Evaluation. Die Evaluationsergebnisse sind in Tabelle 6 aufgelistet. Sie stellt die Precision- und Recall-Werte sowie das F-Mass für das `value`-Attribut in den annotierten TIMEX3-Tags vor und nach der verbesserten Disambiguierung (vD) dar.

Sicherlich ist das Vorgehen des Skripts *disam.py* nicht für die Disambiguierung aller

temporalen Ausdrücke geeignet und in manchen Fällen kann es sogar zu falschen Ergebnissen führen. Die Ergebnisse der Evaluation nach der verbesserten Disambiguierung zeigen jedoch, dass die von HeidelbergTime generierten Werte im `value`-Attribut durch das Skript verbessert werden. Obwohl die Werte immer noch deutlich unter den Werten von Strötgen und Gertz (2011) liegen, haben sie sich doch im Vergleich zur Ausgabe von HeidelbergTime vor der verbesserten Disambiguierung um rund 11 Prozentpunkte verbessert. Ob das Skript korpusübergreifend auf die Annotation aller Jahrbücher angewendet werden kann bzw. auch für andere Artikel eine verbesserte Disambiguierung aufweist, untersuchen wir im Rahmen der Evaluation in Kapitel 3.3.3.

| VALUE   | Precision | Recall | F-Mass |
|---------|-----------|--------|--------|
| vor vD  | 53,9      | 55,2   | 54,5   |
| nach vD | 65,3      | 66,9   | 66,1   |

Tabelle 6.: Evaluation von `value` nach verbesserter Disambiguierung (vD). Artikel 3 aus Jahrbuch 1900.

### 3.3.2. Automatische Annotation

In diesem Kapitel werden die Schritte vorgestellt, welche nötig sind, um die deutschen Texte des Text+Berg-Korpus mit dem Tagger HeidelbergTime zu annotieren. Die Jahrbücher des Korpus liegen als XML-Dateien vor, in denen die Artikel, Sätze und Token annotiert wurden. Die Attribute der Tags enthalten Metadaten, wie beispielsweise die Sprache des Artikels oder Angaben über Lemma und POS-Tag eines Tokens. Da HeidelbergTime nur Dateien in Volltextform als Input entgegennimmt, müssen die Jahrbücher im ersten Schritt devertikalisiert werden. Das Durchlaufen der Dateien mit einem XML-Parser macht es uns möglich, jeden Satz eines deutschen Artikels auf eine neue Zeile in eine Textdatei zu schreiben. Die Token des Satzes trennen wir dabei durch ein Leerzeichen. Den Beginn jedes Artikels kennzeichnen wir mit einem Dummy-Artikel-Wort, um später der verbesserten Disambiguierung sowie der Generierung der Standoff-Dateien einen Hinweis auf die Artikelnummer sowie auf die Artikelgrenzen zu geben. Für jedes der Jahrbücher generieren wir eine solche Textdatei mit dem Inhalt aller deutscher Artikel. Ausschlaggebend für die Devertikalisierung ist die Sprache eines Artikels, welche deutsch sein muss.

Im zweiten Schritt annotieren wir die devertikalisierten Jahrbücher mit dem Tagger HeidelbergTime. Die Verarbeitung mit UIMA lässt es zu, dass alle Dateien eines angegebenen Input-Ordners mit einem Aufruf nacheinander annotiert werden. Durch

die unterschiedlichen Definitionen der Muster für Jahreszahlen (je nach Jahr des Jahrbuchs ist die obere Grenze für Jahreszahlen anders, siehe Kapitel 3.3.1) lassen wir immer etwa 20 Jahrbücher aufeinanderfolgender Jahre gemeinsam annotieren. Die Annotation mit HeidelbergTime für ein solches Jahrbuch-Paket dauert mit unserer Linux-Umgebung ca. 30-40 Minuten. In der Vorverarbeitung durch Komponenten in UIMA kann die Tokenisierung ausgeschaltet werden, da die devertikalisierten Textdateien der Jahrbücher bereits tokenisiert sind. Lediglich die Satzgrenzenerkennung und der POS-Tagger (TreeTagger) müssen vor die Annotation geschaltet werden. Als Ausgabe für jede Jahrbuch-Textdatei generiert HeidelbergTime eine XML-Datei des Typs *.tml*, welche die TIMEX3-Tags enthält, jedoch keine Formatänderungen des Texts aufweist. Dies ist wichtig, um die Standoff-Dateien zu generieren.

Eine Standoff-Annotation „beschreibt die logische Trennung von Primärdaten und Annotation“ (Burghardt und Wolff 2009). Sie enthält Metadaten zu einem Text, welche nicht in der originalen Korpusdatei, sondern in einer separaten Datei abgespeichert werden. Im Fall von Text+Berg gelten die XML-Dateien der Jahrbücher als Primärdaten. Für die Jahrbücher des Text+Berg-Korpus gibt es bereits Standoff-Dateien mit Informationen über Eigennamen von Bergen, Personen und Hütten. Jedes TIMEX3-Tag in der Standoff-Datei hat das Attribut *span*, in welchem angegeben ist, welche Token aus der originalen XML-Datei des Jahrbuchs das TIMEX3-Tag in der Standoff-Datei umfasst. Im Fall der TIMEX3-Tags geben wir zusätzlich die Attribute *type* und *value*, sowie gegebenenfalls die Attribute *mod*, *freq* oder *quant* an. Ausserdem erhält jedes TIMEX3-Tag eine ID. Wir ordnen alle TIMEX3-Tags in den Standoff-Dateien in der Reihenfolge der Typen: TIME, DATE, DURATION, SET. Die Angabe der Tokenspanne hat die Form *Artikel-Satz-Token*. Mit einem Skript berechnen wir diese Spanne anhand der Artikel-Dummies im Text (Artikel), der Zeilennummer im Artikel (Satz) sowie der Tokenposition im Satz (Token). In seltenen Fällen wurden durch HeidelbergTime TIMEX3-Tags generiert, die über Satzgrenzen hinausgingen. Wir vermuten, dass dies auf fehlerhafte Analysen des Annotationssystems zurückzuführen ist. In diesen Fällen geben wir die Tokenspanne in der Standoff-Datei nur bis zum letzten Token im Satz an. Beispiel 3.4 zeigt die Zeile für den Ausdruck *jeden Tag* aus der Standoff-Datei für deutsche temporale Ausdrücke des Jahrbuchs 1957.

```
(3.4) <t type="SET" span="2-278-9, 2-278-10" id="t_2415" value="P1D"
 quant="EVERY"/>
```

Die Standoff-Dateien für temporale Ausdrücke kennzeichnen wir durch *-TIMEX3* im Datei-Namen. Sie liegen im XML-Format vor und können zur weiteren Nutzung der temporalen Information (Kapitel 2.2) in Text+Berg verwendet werden.

### 3.3.3. Ergebnisse und Evaluation für Text+Berg

Mit dem angepassten automatischen Annotationssystem HeidelTime konnten wir in den deutschen Artikeln von Text+Berg mehr als 368 000 temporale Ausdrücke identifizieren. Rund 67% davon sind Ausdrücke des Typs DATE, welcher neben Datumsangaben auch Zeitwörter wie *heute*, *bald* oder *früher* umfasst. Die Anzahl an Ausdrücken des Typs TIME für Uhrzeit- und Tageszeitangaben fällt mit ca. 17% deutlich kleiner aus. Etwa 13% der gefundenen temporalen Ausdrücke beschreiben Zeitdauern wie *2 Stunden*. Die Zeitausdrücke für Regelmässigkeit und Häufigkeit des Typs SET kommen verhältnismässig selten vor und machen nur etwa 3% aller temporaler Ausdrücke aus. Tabelle 7 zeigt die genaue Anzahl extrahierter temporaler Ausdrücke, geordnet nach Typ.

| Typ          | Anzahl TIMEX3  |
|--------------|----------------|
| TIME         | 61'145         |
| DATE         | 246'634        |
| DURATION     | 49'889         |
| SET          | 10'855         |
| <b>TOTAL</b> | <b>368'523</b> |

Tabelle 7.: Anzahl temporaler Ausdrücke im gesamten Text+Berg-Korpus.

Im Anhang A stellen wir die temporalen Ausdrücke nach Typ geordnet in Diagrammen dar. Dabei berechneten wir für jeden Zeitausdrucks-Typ den prozentualen Anteil an allen gefundenen temporalen Ausdrücken im jeweiligen Jahrbuch, das Verhältnis von annotierten Zeitausdrücken des jeweiligen Typs zur Gesamtanzahl an Token im Jahrbuch und das Verhältnis aller annotierter Token eines Typs zu allen Token im betreffenden Jahrbuch (3 Diagramme pro Typ).

Bei allen Diagrammen des Typs TIME sehen wir, dass diese temporalen Ausdrücke im Laufe der Jahre immer seltener in den Jahrbüchern erkannt wurden. Das Verhältnis von TIME-Ausdrücken zu allen TIMEX3-Annotationen zeigt: Während im Jahrbuch von 1868 fast 31% der temporalen Ausdrücke dem Typ TIME angehören, macht dieser Typ im Jahr 1988 nur etwa 2% aus. Da die meisten Uhrzeit- und Tageszeitausdrücke in den Jahrbüchern vor 1900 extrahiert wurden, kann der Häufigkeitsverlauf auch auf die Anpassung auf einen Artikel aus dem Jahr 1900 zurückzuführen sein. So könnte die Art von Ausdrücken, die in diesen Jahren häufig verwendet wurden, rund 100 Jahre später weniger gebräuchlich sein. Wir stellten jedoch fest, dass es vor allem beim Typ TIME, welcher meist Uhrzeiten beschreibt, am wenigsten Variabilität gibt, da Uhrzeiten meist zwingend Ziffern oder Zahlwörter beinhalten müssen. Daher betrachten wir das Argument des Trainingseffekts nicht als stichhaltig. Auch das

Verhältnis der **TIME**-Ausdrücke zu allen Token und das Verhältnis aller Token der **TIME**-Annotationen zu der Gesamtanzahl der Token im Jahrbuch zeigen ein solches Bild, bei dem Uhrzeitangaben in jüngeren Jahrbüchern seltener vorkommen.

Diese These wird durch die Verteilung der Zeitausdrücke des Typs **DATE** unterstützt. In jüngeren Jahrbüchern kommen tendenziell mehr Datumsangaben vor. Das Verhältnis von **DATE**-Ausdrücken zu allen **TIMEX3**-Annotationen zeigt hier: Während im Jahr 1868 nur etwa 51% aller temporaler Ausdrücke dem Typ **DATE** angehörten, waren es im Jahr 1988 rund 88%. Setzen wir die annotierten **DATE**-Ausdrücke und ihre Token in Relation zur Gesamtanzahl an Token in den betreffenden Jahrbüchern, erhalten wir ein sehr ähnliches Bild. Durch die Analyse sehen wir, dass sich die Verwendung von Zeitausdrücken in alpinen Texten im Laufe der Jahre deutlich verändert hat. Wir vermuten, dass früher auf die genau Beschreibung eines Tagesablaufs mit Uhrzeitangaben mehr Wert gelegt wurde, während in den neueren Jahren auf diese Art der Beschreibung meist verzichtet wurde. Dafür nehmen die Datumsangaben in jüngeren Jahrbüchern zu.

Für temporale Ausdrücke des Typs **DURATION** konnte anhand des Verhältnisses von Annotationen dieses Typs zu allen **TIMEX3**-Annotationen keine solche Tendenz entdeckt werden. Der Anteil an Zeitdauern in diesem Diagramm verläuft über die Jahre relativ konstant. Interessanterweise liegen der tiefste und der höchste Anteil an Zeitdauern mit etwa 7% im Jahr 1988 und über 21% im Jahr 1992 beide in Jahrbüchern der jüngeren Zeit. Setzen wir die **DURATION**-Annotationen und ihre Token in Relation zur Gesamtanzahl an Token in den jeweiligen Jahrbüchern, so erhalten wir eine andere Analyse. Das zweite und dritte Diagramm für **DURATION** zeigen, dass in den Jahrbüchern zwischen 1980 und 2000 mehr Token als Zeitdauern annotiert wurden als in den meisten anderen Jahren. Ausserdem werden tendenziell mehr Zeitdauern in den Jahren um 1900 erkannt, was wir auf einen Trainingseffekt zurückführen.

Auch bei den temporalen Ausdrücken des Typs **SET** sehen wir in allen Diagrammen einen leichten Trainingseffekt. So ist der Anteil an **SET**-Ausdrücken in diesem Diagramm in den Jahrbüchern um 1900 tendenziell etwas höher als in den Jahren zwischen 1930 und 1980. Erst ab 1980 kommen mehr Zeitausdrücke des Typs **SET** vor, vergleichbar mit der Analyse für **DURATION**. Das Verhältnis der **SET**-Ausdrücke zu allen **TIMEX3**-Annotationen zeigt: Mit fast 7% wird im Jahr 1987 der höchste Anteil an **SET**-Ausdrücken erreicht. Der niedrigste Wert liegt mit etwa 1% im Jahrbuch von 1897. Auch die Diagramme mit den Verhältnissen der Annotationen und ihrer Token zu allen Token in den Jahrbüchern zeigen ähnliche Analysen.

Für die Evaluation des angepassten Systems HeidelbergTime als temporaler Tagger für Text+Berg annotierten wir jeweils 200 Sätze aus 4 Jahrbüchern manuell, um ein Evaluationskorpus von insgesamt 800 Sätzen zu erhalten. Um die Jahrbücher in ihrer Breite möglichst gut zu repräsentieren, stammen die Evaluationsdaten aus

Jahrbüchern im Abstand von 30 Jahren: 1869<sup>10</sup>, 1930, 1960 und 1990. Die jeweils 200 Sätze jedes Jahrbuchs sind zusammenhängende Sätze aus einem oder aufeinander folgenden Artikeln. Bei der manuellen Annotation vergaben wir die `TIMEX3`-Tags anhand der TimeML Richtlinien. Wir definierten nur Attribute, welche auch HeidelbergTime generiert<sup>11</sup>. Wir achteten darauf, alle Zeitausdrücke zu annotieren, unabhängig von unserer Kenntnis über die in HeidelbergTime enthaltenen Extraktionsmuster. Un genügend spezifizierte temporale Ausdrücke disambiguierten wir mithilfe des Kontextes. Zur Berechnung der Evaluationsergebnisse erstellten wir ein Python-Skript, welches durch XML-Parsing alle `TIMEX3`-Tags eines Satzes des manuell annotierten Evaluationskorpus mit den Tags aus dem gleichen Satz des durch HeidelbergTime getaggtten Textes vergleicht. Dafür wandelten wir die getaggtten Texte in ein spezielles Evaluationsformat um, in dem jeder Satz durch ein `<s>`-Tag gekennzeichnet ist und eine Satz-ID erhält, um sicherzustellen, dass die gleichen Sätze verglichen werden. In Tabelle 8 sind die Precision- und Recall-Werte sowie das F-Mass aufgelistet. Die Werte werden zuerst nach Jahren geordnet dargestellt. Danach folgt die Auflistung der Werte für das gesamte Evaluationskorpus (*Total*), sowie zum Zweck des Vergleichs die Werte für das Entwicklungskorpus (Artikel 3 aus Jahrbuch 1900) und die Werte für das WikiWarsDE-Korpus von Strötgen und Gertz (2011).

Die Evaluationsergebnisse zeigen bessere Werte für Precision als für Recall. Betrachten wir die Recall-Werte, so erkennt die partielle Extraktion rund 24% der relevanten temporalen Ausdrücke nicht. Wenn von einer strikten Extraktion ausgegangen wird, werden etwa 38% der relevanten Zeitausdrücke nicht gefunden. Dies deutet darauf hin, dass viele temporale Ausdrücke im Entwicklungskorpus (Artikel 3 aus Jahrbuch 1900) nicht vorkamen und deshalb nicht Teil der HeidelbergTime-Ressourcen sind. In Anbetracht der Recall-Werte für die Sätze der unterschiedlichen Jahre stellen wir fest, dass in den Jahrbüchern 1869 und 1930, welche dem Entwicklungskorpus zeitlich am nächsten liegen, nicht mehr relevante temporale Ausdrücke extrahiert wurden als in den Jahrbüchern der restlichen Jahre. In den Texten späterer Jahre messen wir in etwa gleich gute Ergebnisse. Die Recall-Werte für die strikte Extraktion sind in den neueren Jahren sogar deutlich besser als in den älteren Jahrbüchern. Bei den Werten für `value` hat die Anpassung für das Jahr 1900 einen Einfluss, denn die Werte für die Jahrbücher 1869 und 1930 sind etwas besser als für die Jahre 1960 und 1990. Generell liegen die Recall-Werte für `value` jedoch etwas unter den Werten für Artikel 3 aus Jahrbuch 1900. Betrachten wir die Precision-Werte der partiellen Extraktion, so sind etwa 91% der von HeidelbergTime identifizierten temporalen Ausdrücke relevant, bei der strikten Extraktion sind es nur rund 73%. Auch diese Werte liegen

---

<sup>10</sup>Das Jahrbuch 1870 ist nicht Teil des Korpus Text+Berg.

<sup>11</sup>HeidelbergTime generiert nicht alle möglichen, in den TimeML vorgeschlagenen Attribute.



|                     | partiell    |             |             | strikt      |             |             | value       |             |             |
|---------------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
|                     | P           | R           | F           | P           | R           | F           | P           | R           | F           |
| <b>1869</b>         | 98,7        | 79,6        | 88,1        | 69,6        | 56,1        | 62,1        | 59,5        | 48,0        | 53,1        |
| <b>1930</b>         | 88,9        | 70,6        | 78,7        | 72,8        | 57,8        | 64,4        | 70,4        | 55,9        | 62,3        |
| <b>1960</b>         | 87,7        | 79,2        | 83,2        | 72,3        | 65,3        | 68,6        | 47,7        | 43,1        | 45,3        |
| <b>1990</b>         | 87,5        | 76,7        | 81,7        | 78,1        | 68,5        | 73,0        | 50,0        | 43,8        | 46,7        |
| <b>Total</b>        | <b>90,7</b> | <b>76,5</b> | <b>82,9</b> | <b>73,2</b> | <b>61,9</b> | <b>67,0</b> | <b>56,9</b> | <b>47,7</b> | <b>51,9</b> |
| <b>Art. 3, 1900</b> | 95,2        | 97,5        | 96,3        | 86,8        | 89,0        | 87,9        | 53,9        | 55,2        | 54,5        |
| <b>WikiWarsDE</b>   | 98,5        | 85,0        | 91,3        | 92,6        | 79,9        | 85,8        | 87,0        | 87,0        | 87,0        |

Tabelle 8.: Evaluation für 800 Sätze aus Text+Berg nach Anpassung von HeidelTime.

etwas unter den Werten für das Entwicklungskorpus aus dem Jahr 1900. Wie bei den Recall-Werten erzielt auch hier die strikte Extraktion in neueren Jahrbüchern bessere Ergebnisse als in älteren Jahrbüchern. Die Precision-Werte für `value` liegen mit ca. 57% für alle Evaluationsdaten etwas über den Werten für das Entwicklungskorpus. Jedoch zeigt sich zwischen den einzelnen Jahrbüchern der Evaluationsdaten eine grosse Varianz in den Precision-Werten für `value`. Während der Wert für die 200 Sätze aus Jahrbuch 1930 bei 70,4 liegt, erzielt die Auswertung der `value`-Werte aus Jahrbuch 1960 einen Wert von nur 47,7.

Diese Evaluationsergebnisse zeigen, wie schon die Ergebnisse aus der Evaluation der Extraktion für den Artikel 3 aus Jahrbuch 1900, dass der `value`-Wert die schlechtesten Ergebnisse erzielt. Die richtige Normalisierung der temporalen Ausdrücke bzw. die korrekte Disambiguierung von ungenügend spezifizierten temporalen Ausdrücken stellt die schwierigste Aufgabe für den temporalen Tagger dar. Hier werden die meisten Fehler gemacht. Die normalisierten Werte im ISO-8601-Format werden vom Tagger nur zu etwas mehr als 50% richtig generiert. Da das Skript zur verbesserten Disambiguierung von temporalen Ausdrücken im Entwicklungskorpus die Evaluationsergebnisse um rund 11 Prozentpunkte (Precision und Recall) anheben konnte, wenden wir es auch auf die Evaluationsdaten an. Wir wollen herausfinden, ob das für Artikel 3 aus Jahrbuch 1900 entwickelte Skript auch für Artikel anderer Jahrbücher Verbesserungen in der `value`-Generierung erzielen kann. Tabelle 9 zeigt die Precision- und Recall-Werte sowie das F-Mass für `value` vor und nach der verbesserten Disambiguierung (vD), geordnet nach Jahren.

Das Skript zur Verbesserung der Disambiguierung erzielt für die Evaluationsdaten sehr unterschiedliche Ergebnisse. Für die jeweils 200 Sätze aus den Jahrbüchern

1869 und 1930 bleiben die Recall- und Precision-Werte für `value` unverändert. Wir überprüften daraufhin einzelne Textpassagen aus diesen Jahrbüchern manuell. Die Prüfung ergab, dass das Skript zwar neue Werte für das `value`-Attribut errechnen konnte, jedoch stimmen diese anschliessend immer noch nicht mit den korrekten Werten der betreffenden `TIMEX3`-Tags im Gold Standard überein. Die Ergebnisse für Jahrbuch 1960 zeigen, dass das Skript durch die Neuberechnung der Datumsangaben die `value`-Werte auch verschlechtern kann. Nur in den 200 Sätzen aus Jahrbuch 1990 konnte das Skript die Disambiguierung um rund 3 Prozentpunkte verbessern.

Allgemein zeigen die Evaluationsergebnisse, dass die Anpassung eines temporalen Taggers für das zu annotierende Korpus deutliche Verbesserungen der Extraktion erbringt. Die richtige Interpretation der temporalen Ausdrücke ist hingegen um einiges schwieriger und auch der Versuch, die Werte für `value` in einem zweiten Schritt neu zu berechnen, bringt nur geringe Verbesserungen. Da das Skript zur verbesserten Disambiguierung jedoch nur in 200 von 800 Sätzen eine Verschlechterung der `value`-Werte erzeugte und in 400 Sätzen der Evaluationsdaten keine Veränderungen zeigte, wendeten wir es nach der Annotation durch HeidelTime auf alle Dateien des Korpus an.

|              | <b>value</b> |             |             | <b>value vD</b> |             |             |
|--------------|--------------|-------------|-------------|-----------------|-------------|-------------|
|              | <b>P</b>     | <b>R</b>    | <b>F</b>    | <b>P</b>        | <b>R</b>    | <b>F</b>    |
| <b>1869</b>  | 59,5         | 48,0        | 53,1        | 59,5            | 48,0        | 53,1        |
| <b>1930</b>  | 70,4         | 55,9        | 62,3        | 70,4            | 55,9        | 62,3        |
| <b>1960</b>  | 47,7         | 43,1        | 45,3        | 43,1            | 38,9        | 40,9        |
| <b>1990</b>  | 50,0         | 43,8        | 46,7        | 53,1            | 46,6        | 49,6        |
| <b>Total</b> | <b>56,9</b>  | <b>47,7</b> | <b>51,9</b> | <b>56,5</b>     | <b>47,4</b> | <b>51,5</b> |

Tabelle 9.: Vergleich Evaluation von `value` vor und nach verbesserter Disambiguierung (vD) in 800 Sätzen, nach Jahr geordnet.

## 4. Annotation französischer temporaler Ausdrücke

Das Text+Berg-Korpus enthält neben dem grössten deutschsprachigen Teil auch Artikel in französischer Sprache. Die Jahrbücher ab dem Jahr 1957 wurden in Deutsch und Französisch publiziert und bilden ein paralleles Korpus. Zudem sind digitalisierte Ausgaben der französischsprachigen Zeitschrift *Echo des Alpes* verfügbar. Neben weiteren fremdsprachigen Artikeln in italienisch, englisch und rätoromanisch machen französische Texte den zweitgrössten Teil des Korpus aus.

Die vorliegende Arbeit beschäftigt sich neben der Annotation von deutschen temporalen Ausdrücken auch mit der automatischen temporalen Annotation im Französischen, welche wir in diesem Kapitel untersuchen. Nach der Analyse des aktuellen Stands auf diesem Forschungsgebiet, beschreiben wir den Tagger von Bittar (2009) genauer und verwenden ihn für die Annotation in Text+Berg. Die Ergebnisse der Evaluation dieser Annotation beschliessen das Kapitel.

### 4.1. Stand der Forschung zur Annotation französischer temporaler Ausdrücke

Unter den ersten Forschungen zur automatischen Extraktion von temporalen Ausdrücken in französischen Texten befindet sich die Arbeit von Vazov (2001), welcher mit seinem Ansatz temporale Ausdrücke mithilfe von syntaktischen und semantischen Einschränkungen erkennt. Das System extrahiert französische Adverben, adverbiale Nominalphrasen und adverbiale Ausdrücke, um Antwort geben zu können auf die Frage, *wann* etwas passiert ist. Als Testkorpus dienen dem System etwa 500 Sätze aus amtlichen Berichten des kanadischen Parlaments (*Hansard Canadianen 1986-1993*). Das regelbasierte System setzt sich zusammen aus der Suche von regulären Ausdrücken und aus „left-to-right and right-to-left chart parsing“ (Vazov 2001). Dies ist eine Parsingstrategie, welche bei langen temporalen Ausdrücken den Kontext links und rechts eines bestimmten Markers untersucht. Beide Suchtechniken

werden durch semantische und syntaktische Faktoren eingeschränkt. Die extrahierten temporalen Ausdrücke wurden bei diesem Ansatz noch nicht anhand eines Standards annotiert. Zur Evaluation des Systems wurden in den Sätzen des Testkorpus alle temporalen Ausdrücke mit den dazugehörigen Markern manuell identifiziert. Vazov (2001) erzielt mit seinem System für die Extraktion einen Precision-Wert von 85,0 und einen Recall-Wert von 95,0.

Im Rahmen der Forschung von Muller und Tannier (2004) zur Annotation und Evaluation von temporalen Relationen in französischen Texten wurde ebenfalls eine Extraktion von temporalen Ausdrücken durchgeführt. Das Entwicklungskorpus für diese Arbeit bestand aus 8 Texten aus der Domäne *news* der AFP (Agence France Presse). Um die temporalen Ausdrücke in diesem Korpus zu erkennen, wurden Kaskaden mit regulären Ausdrücken verwendet. Diese Technik konzentrierte sich allerdings nur auf das Erkennen von Datumsangaben und Zeitdauern. Muller und Tannier (2004) versuchten ebenfalls, ungenaue temporale Angaben zu disambiguieren, um Aktionen (EVENT) zeitlich präzise lokalisieren zu können. Sie konzentrieren sich in ihrer Arbeit zwar stärker auf die korrekte Annotation von Relationen zwischen Aktionen, bezeichnen die Detektion von temporalen Ausdrücken jedoch als eine der wichtigsten Informationsquellen für diesen Prozess.

Einer der ersten Ansätze im Rahmen der TimeML-Annotationsrichtlinien zur automatischen Erkennung von temporalen Ausdrücken und Aktionen für das Französische kommt von Parent et al. (2008). Die Autoren bezeichnen ihre Forschung auf dem Gebiet der temporalen Annotation zu diesem Zeitpunkt als die einzige für das Französische, die auf diese Weise versucht, temporale Information aus einem Text zu extrahieren (Parent et al. 2008: „Pour le français, à notre connaissance, nous sommes les seuls à avoir abordé ce problème en suivant une méthodologie comparable.,,“). In ihrer Arbeit stellen sie ein System vor, dass zuerst temporale Ausdrücke mit einer regelbasierten Kaskade an Automaten erkennt und diese anhand der TimeML-Annotationsrichtlinien klassifiziert. Das System zur Erkennung temporaler Ausdrücke basiert auf einer Kaskade mit 20 Niveaus und etwa 90 Regeln. Es ist auf die Domäne der *news* ausgelegt, denn als Entwicklungskorpus verwenden Parent et al. (2008) Agenturmeldungen der AFP. Im zweiten Schritt wird versucht, Aktionen korrekt zu erkennen und zu annotieren. Mit ihrem System erzielten Parent et al. (2008) Evaluationsergebnisse von 93,0 für Precision und Recall für die strikte Erkennung temporaler Ausdrücke im Entwicklungskorpus. Für das Testkorpus, welches aus Artikeln der Webseite der École Polytechnique de Montréal besteht, erreicht das System Precision-Werte von 83,0 und Recall-Werte von 79,0 für die strikte Erkennung. Diese Ergebnisse zeigen, dass die temporale Annotation vor allem dann gute Ergebnisse erbringt, wenn das System auf das zu annotierende Korpus angepasst wurde. Dies stützt die These dieser Arbeit, dass eine qualitativ hochwertige auto-

matische Annotation von temporalen Ausdrücken zwingend abhängig ist von der Domäne der zu annotierenden Texte und eine Anpassung der betreffenden Systeme an das zu annotierende Korpus unumgänglich ist.

Auf der gleichen Konferenz JEP / TALN / RECITAL 2008<sup>1</sup>, auf der auch Parent et al. (2008) ihr Projekt vorstellten, präsentierte Bittar (2008) seine ersten Ansätze zur Elaboration einer kompletten Ressource für die Annotation von französischen Texten nach den TimeML-Richtlinien. Diese Ressource sollte neben sprachspezifischen Annotationsrichtlinien und einem Referenzkorpus auch ein Modul zur automatischen temporalen Textannotation umfassen, welches wir im folgenden Kapitel beschreiben und anschliessend als temporalen Tagger für die französischen Texte im Text+Berg-Korpus verwenden.

## 4.2. Unitex-Graphen und Normalisierung nach Bittar (2009)

Die Arbeit zu TimeML-Ressourcen für das Französische nach Bittar (2008) umfasst einen temporalen Tagger zur Annotation von TIMEX3-Tags in französischen Texten. Für dieses System wurde ein regelbasierter Ansatz gewählt, da ein bereits annotiertes Korpus für das Training möglicher statistischer Systeme zu diesem Zeitpunkt nicht verfügbar war. Die Regeln zur Extraktion der temporalen Ausdrücke basieren auf dem Paket *Time\_French* von Gross (2002). Dieses Paket enthält eine umfassende lokale Grammatik zur Auffindung von temporalen Ausdrücken in französischen Texten. Dies umfasst Datumsangaben, Zeitangaben, Zeitdauerangaben und Ausdrücke der Häufigkeit. Die Graphen stellen endliche Automaten dar, die im Verarbeitungsprogramm für Korpora *Unitex*<sup>2</sup> auf Texte angewendet werden können. Mit Unitex können elektronische Ressourcen wie Wörterbücher oder Grammatiken zur Korpusanalyse verwendet werden. Mit diesem Programm können wir in Texten Muster lokalisieren.

Nach Gross (2002) sind Graphen von endlichen Automaten besonders geeignet, um eine lokale Grammatik zur Detektion von temporalen Ausdrücke im Französischen zu generieren. Dabei repräsentiert jeder Zustand des Automaten ein Wort aus einer Auswahl an Begriffen und eine Sequenz von Zuständen beschreibt eine Abfolge von Wörtern bzw. einen Ausdruck der Grammatik. Die Graphen erkennen so z. B. Zeitangaben wie *au matin* (am Morgen), *le soir de la présente journée* (am Abend des gegenwärtigen Tages) oder numerische Zeitangaben des 12-Stunden- wie auch

---

<sup>1</sup><http://www.atala.org/TALN-RECITAL-JEP-2008-15eme>

<sup>2</sup><http://www-igm.univ-mlv.fr/unitex/>

des 24-Stunden-Zyklus. Gross (2002) schlägt zur Auflösung der Ambiguität des 12-Stunden-Zeitzyklus die Nutzung der Tageszeitangaben vor, was wir auch bei der Anpassung des deutschen Taggers HeidelbergTime berücksichtigt haben. In diesem Zusammenhang weist er auf das Problem der generischen Zeitangaben hin, welche nicht als temporale Ausdrücke annotiert werden sollten, wie z. B. die Angabe *du jour in plat du jour* (Tagesgericht).

Die lexikalischen Ressourcen in Form von Graphen wurden für die Annotation anhand der TimeML-Richtlinien angepasst und neu organisiert, um die damit extrahierten Muster den Typen TIME, DATE, DURATION und SET zuordnen zu können (Bittar 2010). Das Hinzufügen der Generierung von TIMEX3-Tags<sup>3</sup> durch die Graphen macht sie zu endlichen Transduktoren. Zusätzlich wurde ein Graph entwickelt, der Muster annotiert, welche keine temporalen Ausdrücke darstellen und diese somit für die temporale Annotation sperrt. Darunter fallen beispielsweise Telefonnummern oder Altersangaben.

Unitex annotiert mithilfe dieser Graphen einen Text mit TIMEX3 und SIGNAL-Tags. Anschliessend müssen ISO-8610-konforme Werte für jeden erkannten temporalen Ausdruck generiert werden. Dies geschieht extern mit einem Normalisierungs-Skript in Perl, welches die indirekt angegebenen Werte disambiguiert. Wenn für die Generierung eines Wertes ein Referenzdatum nötig ist, so nimmt das Skript standardmässig das erste gefundene, voll spezifizierte Datum im Text. Alternativ können wir dem Skript ein Referenzdatum als Argument angeben. Hier fällt auf, dass sich der Disambiguierungsprozess vom System HeidelbergTime unterscheidet, wo in der Domäne *narratives* immer das letzte voll spezifizierte Datum im Text als Referenz zur Disambiguierung verwendet wird. Während des Prozesses der Normalisierung im System von Bittar (2009) wird für jeden TIMEX3-Typ eine separate Funktion zur Normalisierung aufgerufen, welche den durch die Graphen extrahierten String als Argument nimmt und mithilfe der Information im String und den Informationen im Referenzdatum versucht, das *value*-Argument zu generieren. Kann ein Element wie z. B. der Monat nicht disambiguiert werden, so erhält das *value*-Argument an dieser Stelle die Symbole *XX*.

Als Testkorpus für diesen temporalen Tagger wird in Bittar (2009) das Entwicklungskorpus (Agenturmeldungen AFP) aus Parent et al. (2008) verwendet. Die Evaluationsergebnisse liegen für die strikte Extraktion hier bei einem Precision-Wert von 84,2 und einem Recall-Wert von 81,8, was leicht über den Werten von Parent et al. (2008) liegt, welche sie für ihr Testkorpus, das nicht zum Training verwendet wurde, erhielten.

Bittar beschreibt seinen temporalen Tagger in verschiedenen Publikationen, jedoch

---

<sup>3</sup>Zusätzlich generieren die Graphen auch SIGNAL-Tags, die wir hier nicht genauer betrachten.

erhielten wir die Angaben über den Download des frei verfügbaren Systems, bestehend aus den Graphen für Unitex und einem Normalisierungsskript in Perl, durch die Publikation Bittar (2009).

## 4.3. Unitex-Graphen als temporaler Tagger für Text+Berg

### 4.3.1. Automatische Annotation

Im Rahmen dieser Arbeit sahen wir in der ersten Planungsphase vor, für die Annotationen der französischsprachigen Texte im Text+Berg-Korpus ebenfalls den temporalen Tagger HeidelbergTime zu verwenden. Da HeidelbergTime in der Version 1.2 nur für das Deutsche, Englische und Niederländische verfügbar ist, wollten wir neue Sprachressourcen für das Französische hinzufügen. Wie die Anpassung der deutschen Ressourcen in Kapitel 3.3 zeigten, ist die Generierung von sprachspezifischen Musterdateien, Regeln und Normalisierungen sehr aufwendig und zeitintensiv, sollte der Tagger, wie im Fall dieser Arbeit, für ein grosses Korpus möglichst gute Ergebnisse erzielen. Da es für das Französische, anders als für das Deutsche, noch keine Ressourcen in HeidelbergTime gibt, hätte eine Erstellung der umfangreichen Sprachressourcen den verfügbaren Zeitrahmen für diese Arbeit überschritten. Um für die Domäne der alpinen Texte ein funktionierendes Annotationssystem zu erstellen, müssten wir das Korpus intensiv untersuchen. Wir sollten allgemeine sowie für diese Textsorte spezifische temporale Ausdrücke identifizieren, um sie anschliessend den Ressourcen in HeidelbergTime hinzuzufügen. Die Ressourcen müssten möglichst viele sprachliche Varianten der temporalen Ausdrücke umfassen, weshalb wir auch die Morphologie und Syntax der temporalen Ausdrücke genau betrachten müssten. Um diese umfangreichen lexikalischen Ressourcen und die dazugehörigen Normalisierungen und Regeln in HeidelbergTime zu generieren, sie zu implementieren, zu testen, zu evaluieren und gegebenenfalls zu verbessern, wären nach unserer Auffassung allein mindestens 6 Monate nötig.

In Anbetracht dieser Gründe verwendeten wir für die Annotation von temporalen Ausdrücken in den französischen Texten das System von Bittar (2009), jedoch wurden an diesem System keine Anpassungen für das zu annotierende Korpus vorgenommen.

Die Schritte der automatischen Annotation für das Französische sind die gleichen wie für die deutschen Texte im Text+Berg-Korpus. Zuerst devertikalisieren wir die Jahrbücher, d. h. wir bringen sie in Volltext-Form. Anschliessend extrahieren und

annotieren wir mit den Graphen in Unitex alle temporalen Ausdrücke. Mit dem oben beschriebenen Perl-Skript generieren wir für alle **TIMEX3**-Tags einen normalisierten bzw. disambiguierten Wert für das **value**-Tag. Für jedes Jahrbuch erstellen wir eine XML-Datei, in der alle französischen Artikel in Volltextform mit annotierten **TIMEX3**-Tags enthalten sind. Anschliessend generieren wir mit einem eigenen Skript für jedes Jahrbuch eine Standoff-Datei, in der die Spanne für jeden temporalen Ausdruck in der originalen XML-Datei des Jahrbuchs angegeben ist. Hier kann nicht das gleiche Programm wie für das Deutsche verwendet werden. Da die Unitex-Graphen durch die Annotation das Textformat in manchen Fällen leicht verändern und Sätze aus zwei Zeilen auf eine einzige Zeile zusammenfügen, überprüfen wir bei der Generierung der Spanne für einen temporalen Ausdruck, ob die ID der Spanne tatsächlich eine korrekte ID aus dem originalen Jahrbuch darstellt.

### 4.3.2. Ergebnisse und Evaluation für Text+Berg

Um das Ergebnis der Unitex-Graphen als temporalen Tagger für französische Texte des Text+Berg-Korpus zu evaluieren, annotierten wir jeweils 100 zusammenhängende Sätze aus 5 verschiedenen Jahrbüchern manuell, sodass wir Evaluationsdaten von 500 Sätzen nutzen konnten. Wie auch schon für die Evaluation der deutschen Annotationen, wählten wir Jahrbücher im Abstand von etwa 30 Jahren aus, um eine möglichst repräsentative Stichprobe des gesamten Korpus zu erhalten. Die jeweils 100 Sätze aus den Jahren 1872 und 1900 entstammen dem Subkorpus *Echo des Alpes*. Für die Jahre 1960 und 1990 wählten wir jeweils 100 zu den deutschen Evaluationsdaten parallele Sätze aus. Die Tabelle 10 zeigt die Precision- und Recall-Werte sowie das F-Mass für 500 Sätze, nach Jahren geordnet und die Ergebnisse für die gesamten Evaluationsdaten (*Total*). Zum Vergleich stellen wir auch die Werte aus der Evaluation des angepassten Taggers HeidelbergTime für deutsche Zeitausdrücke dar (*DE HT*).

Die Relevanz bzw. Präzision der extrahierten temporalen Ausdrücke variiert sehr stark. So messen wir für die partielle Extraktion in den Sätzen der Jahre 1872, 1929 und 1960 Precision-Werte, die nur wenig schlechter sind als die vergleichbaren Werte des angepassten HeidelbergTime-Systems für die deutschen Texte. Für die Sätze des Jahres 1990 schneidet das System der Unitex-Graphen hingegen schlechter ab und annotiert viele Ausdrücke, die keine Zeitausdrücke darstellen. Betrachtet man die strikte Extraktion, so sind die Werte in manchen Jahren ebenfalls vergleichbar mit den deutschen Evaluationsergebnissen (1929 und 1960), in anderen Jahren fällt der Wert jedoch bis zu 67 Prozentpunkte schlechter aus. Auch die **value**-Werte sind deutlich schlechter als für das System HeidelbergTime.



|              | partiell    |             |             | strikt      |             |             | value       |            |             |
|--------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|------------|-------------|
|              | P           | R           | F           | P           | R           | F           | P           | R          | F           |
| <b>1872</b>  | 80,0        | 9,8         | 17,5        | 40,0        | 4,9         | 8,7         | 40,0        | 4,9        | 8,7         |
| <b>1900</b>  | 62,5        | 27,8        | 38,5        | 18,8        | 8,3         | 11,5        | 6,2         | 2,8        | 3,9         |
| <b>1929</b>  | 88,9        | 20,5        | 33,2        | 66,7        | 15,4        | 25,0        | 44,4        | 10,3       | 16,7        |
| <b>1960</b>  | 80,0        | 21,1        | 33,4        | 60,0        | 15,8        | 25,0        | 40,0        | 10,5       | 16,6        |
| <b>1990</b>  | 44,4        | 16,0        | 23,5        | 22,2        | 8,0         | 11,8        | 22,2        | 8,0        | 11,8        |
| <b>Total</b> | <b>71,2</b> | <b>19,0</b> | <b>29,2</b> | <b>41,5</b> | <b>10,5</b> | <b>16,4</b> | <b>30,6</b> | <b>7,3</b> | <b>11,5</b> |
| <b>DE HT</b> | 90,7        | 76,5        | 82,9        | 73,2        | 61,9        | 67,0        | 56,9        | 47,7       | 51,9        |

Tabelle 10.: Evaluation des Taggers nach Bittar (2009). Ergebnisse für 500 Sätze aus Text+Berg, nach Jahr geordnet. P = Precision, R = Recall, F = F-Mass. *Total* gibt die Werte für das gesamte Evaluationskorpus an. *DE HT* zeigt die Werte von HeidelTime (angepasst) für das deutsche Evaluationskorpus.

Die Anzahl an relevanten temporalen Ausdrücken aus dem Gold Standard, welche durch die Unitex-Graphen extrahiert werden konnten, ist extrem gering, wie die Recall-Werte in Tabelle 10 aufzeigen. Mit einem Durchschnittswert von 19% für die partielle Extraktion und 10,5% für die strikte Extraktion beweisen sie, dass das System nur einen sehr geringen Anteil relevanter Ausdrücke der manuell annotierten Evaluationsdaten identifizieren kann. Werte von unter 10% in den einzelnen Jahrbüchern für die strikte Extraktion und die *value*-Werte zeigen im Vergleich zur deutschen Annotation, dass die Unitex-Graphen nur bedingt für die temporale Annotation französischer Texte im Text+Berg-Korpus geeignet sind. Dies beweist auch das F-Mass, welches für die partielle Extraktion französischer Zeitausdrücke über 50 Prozentpunkte schlechter ausfällt als bei der Evaluation des angepassten Taggers HeidelTime für die Extraktion deutscher temporalen Ausdrücke. Diese Ergebnisse zeigen, dass die Anpassung eines automatischen Annotationssystems für ein spezielles Korpus einen wichtigen Teil zur Qualität der automatischen temporalen Annotation beiträgt.

Die Evaluation des Systems von Bittar (2009) lassen uns erkennen, dass wir die automatische Annotation für französische Texte in Text+Berg bis zu diesem Punkt nicht zufriedenstellend durchführen konnten und diese temporalen Annotationen nur sehr schlecht zur weiteren Verwendung genutzt werden können.

## 5. Nutzung der Alignierungen zur Annotation temporaler Ausdrücke

Das Korpus Text+Berg umfasst Jahrbücher von 1864 bis 2011. Der Schweizer Alpenclub publizierte alle Jahrbücher ab dem Jahr 1957 parallel in deutscher und französischer Sprache. Das bedeutet, dass die Inhalte jeweils in beide Sprachen übersetzt wurden. Ist die Originalsprache für einen bestimmten Artikel bekannt, so ist dies in den Metadaten des Korpus vermerkt. Insgesamt sind durch die Übersetzungen 55 parallele Jahrbücher verfügbar. Das parallele Korpus umfasst jeweils mehr als 4 Millionen Token in deutscher und französischer Sprache (Göhring und Volk 2011).

Parallele Korpora sind sehr wertvoll in der Computerlinguistik. Die statistische maschinelle Übersetzung braucht parallele Korpora zum Training der Systeme. Parallele Texte können zum Vergleich verschiedener linguistischer Untersuchungen verwendet werden, so z. B. zur Vergleichbarkeit von semantischen Rahmen in zwei Sprachen. Auch können aus parallelen Korpora Suchsysteme wie *bilingwis*<sup>1</sup> gebaut werden, bei denen es möglich ist, die Übersetzung eines Wortes der Ausgangssprache im Kontext der Zielsprache zu sehen. Alle diese Anwendungen basieren auf der Alignierung der parallelen Texte. *Alignierung* bedeutet, dass Informationen darüber vorliegen, welches Element der Ausgangssprache welchem Element der Zielsprache entspricht. Aligierte Elemente können Artikel, Paragraphen, Sätze, Phrasen oder Wörter bzw. Token sein. So kann z. B. das deutsche Wort *Woche* mit dem französischen Ausdruck *semaine* aligniert sein. Da bei der Grösse der meisten Textkorpora der Prozess des manuellen Alignierens sehr zeitintensiv wäre, werden parallele Korpora in der Regel durch automatische Systeme aligniert.

In diesem Kapitel beschreiben wir zuerst das automatische Alignierungsverfahren, welches zur Alignierung des parallelen Teils in Text+Berg entwickelt wurde (5.1). Anschliessend untersuchen wir, wie die Alignierungsinformationen zur automatischen temporalen Annotation genutzt werden können. Neben dem Vergleich von temporalen Ausdrücken (5.2) testen wir die Projektion temporaler Annotationen

---

<sup>1</sup><http://kitt.cl.uzh.ch/kitt/bilingwis/>

im parallelen Korpus (5.3). Zum Schluss versuchen wir, die temporale Annotation im Französischen durch die Kombination der beiden Verfahren zu verbessern.

## 5.1. Alignierungen in Text+Berg

In diesem Kapitel stellen wir das System von Sennrich und Volk (2010) vor, das zur automatischen Satzalignierung von OCR-generierten parallelen Korpora wie dem Text+Berg-Korpus entwickelt wurde. Bei der Technik des OCR (*Optical Character Recognition*) werden gedruckte Texte durch Einscannen und automatische Texterkennung digitalisiert.

Da OCR-generierte Texte meist mehr Fehler enthalten als herkömmliche digitale Texte, erzeugen automatische Alignierungsalgorithmen wie z. B. Gale und Church (1991) für diese Art von Korpora schlechte Ergebnisse. Das System *bleualign* (Sennrich und Volk 2010) verwendet die maschinelle Übersetzung und BLEU-Punkte, um geeignete Satzalignierungen für einen Text zu finden. In den Beispielen 5.1 und 5.2 zeigen wir zwei Sätze aus dem Text+Berg-Korpus, welche automatisch aligniert wurden. Beispiel 5.1 ist der Satz 257 aus dem Artikel 4 im deutschen Jahrbuch 1991, der dazu alignierte französische Satz in 5.2 ist der Satz 253 aus Artikel 4 des parallelen französischen Jahrbuchs von 1991.

(5.1) Die 51 Gipfel wurden in 24 Tagen bestiegen , nicht eingerechnet die Zeit für die Aufstiege zu den Hütten und für die Strecken zwischen den Bergen .

(5.2) - Les 51 sommets ont été atteints en 24 journées d' ascension proprement dite ( à savoir , sans compter les montées en cabane et les parcours entre les montagnes ) .

Die grundlegende Idee von *bleualign* ist, den Text einer der beiden Sprachen, die es zu alignieren gilt, automatisch zu übersetzen, um anschliessend die Ähnlichkeit der Übersetzung mit dem Text der zweiten Sprache zu vergleichen. Die Satzalignierungen, welche für das Training eines statistischen maschinellen Übersetzungssystems nötig sind, wurden für dieses Verfahren manuell für 1000 Sätze des parallelen Korpus generiert. Im Text+Berg-Korpus gelten die Artikelgrenzen als die einzigen verlässlichen Festpunkte in den Jahrbüchern. Im ersten Schritt des Systems werden die Angaben der Inhaltsverzeichnisse der parallelen Jahrbücher verglichen und bei Übereinstimmung von Wörtern, wie beispielsweise dem Name des Autors, werden die Artikel aligniert. Die Jahrbücher mit Artikelalignierungen werden, zusammen mit ihrer maschinellen Übersetzung, als Input für den Alignierungsalgorithmus verwendet. Nachdem das Programm mithilfe von BLEU-Punkten im Text Festpunkte generiert hat, werden

die Sätze zwischen den Festpunkten aligniert. Dafür verwenden Sennrich und Volk 2010 entweder der Algorithmus von Gale und Church (1991) oder Heuristiken von BLEU. Mit dem Algorithmus von *bleualign* können Sennrich und Volk (2010) die Qualität der Satzalignierung im Vergleich zur Alignierung mit dem Algorithmus von Gale und Church (1991) verbessern.

Darauf aufbauend versuchen Sennrich und Volk (2011) das automatische Alignierungssystem durch einen iterativen Ansatz zu verbessern. Um den Algorithmus *bleualign* anwenden zu können, muss eine möglichst gute maschinelle Übersetzung der Ausgangssprache verfügbar sein, um diese Übersetzung anschliessend mit dem Text der Zielsprache vergleichen zu können. Für ein statistisches maschinelles Übersetzungssystem werden bereits Satzalignierungen benötigt. Im ersten Schritt verwenden Sennrich und Volk (2011) dazu den Alignierungsalgorithmus von Gale und Church (1991), der auf der Basis des Längenvergleichs der Sätze aligniert. *Bleualign* kann anschliessend mit dieser Übersetzung arbeiten, um eine neue Satzalignierung zu generieren. Diese wird dann in das maschinelle Übersetzungssystem eingebunden und eine neue Übersetzung wird erzeugt. Diese Schritte können mehrmals bzw. iterativ durchgeführt werden. Sennrich und Volk (2011) erreichen damit ähnlich gute Ergebnisse wie mit dem reinen Algorithmus *bleualign*. Der Unterschied des iterativen Ansatzes und des alleinstehenden *bleualign*-Algorithmus ist, dass ersterer keine sprachspezifischen Ressourcen braucht bzw. keine manuellen Satzalignierungen im Voraus gemacht werden müssen.

Die so entstandenen Satzalignierungen dienen als Input für ein statistisches maschinelles Übersetzungssystem, in welchem Algorithmen wie GIZA++<sup>2</sup> Wortalignierungen berechnen. Diese Wortalignierungen können wir für verschiedene Untersuchungen im parallelen Teil des Text+Berg-Korpus verwenden. In den folgenden Kapiteln nutzen wir die Alignierungsinformationen für den Vergleich und die Projektion von temporalen Annotationen.

## 5.2. Vergleich der deutschen und französischen Annotationen

Durch die Alignierung des parallelen Korpus in Text+Berg auf der Wortebene bietet sich uns die Möglichkeit, die automatischen Annotationen von temporalen Ausdrücken in den deutschen und französischen Texten zu vergleichen. Die Alignierungsinformationen geben an, welche deutsche Tokensequenz mit welcher französischen Tokensequenz aligniert ist. Das Beispiel 5.3 zeigt eine Zeile aus der Alignierungsdatei

---

<sup>2</sup><http://code.google.com/p/giza-pp/>

zur Version 147v02 des Text+Berg-Korpus.

(5.3) 1960\_de-15-55-16:16 1960\_fr-15-59-14:17

Die Zeile in 5.3 beschreibt die Alignierung des deutschen Tokens 16 des Satzes 55 in Artikel 15 aus dem deutschen Jahrbuch 1960. Dieses Token ist aligniert mit den französischen Token 14 bis 17 des Satzes 59 in Artikel 15 aus dem zum deutschen parallelen französischen Jahrbuch 1960. Hier ist das deutsche Wort *Kartoffeln* mit dem französischen Ausdruck *de pommes de terre* aligniert.

Durch diese Information können wir für jeden deutschen temporalen Ausdruck testen, mit was die entsprechenden Token im französischen aligniert sind. Mit einem Python-Skript führen wir hierzu eine Untersuchung durch. Für alle deutschen temporalen Ausdrücke aus den gesamten Jahrbüchern 1960 und 1990 prüfen wir mit der Alignierungsinformation, ob die alignierten französischen Token ebenfalls als Zeitausdrücke annotiert wurden. Tabelle 11 zeigt die Ergebnisse der Überprüfung der Alignierung von deutschen temporalen Ausdrücken, generiert durch das angepasste System HeidelTime, mit französischen Zeitausdrücken, annotiert durch das Unitex-System von Bittar (2009).

|              | Total TIMEX3-DE | TIMEX3-DE<br>mit Alignierungsinformation | Korrespondenz<br>TIMEX3 DE-FR |
|--------------|-----------------|------------------------------------------|-------------------------------|
| <b>1960</b>  | 2054            | 1239 (60%)                               | 355 (17%)                     |
| <b>1990</b>  | 1355            | 972 (72%)                                | 270 (20%)                     |
| <b>Total</b> | 3409            | 2211 (65%)                               | 625 (18%)                     |

Tabelle 11.: Untersuchung der Korrespondenz von deutschen und französischen temporalen Ausdrücken via Alignierung in den gesamten Jahrbüchern 1960 und 1990.

Für 60% der temporalen Ausdrücke im deutschen Jahrbuch 1960 liegt eine Alignierungsinformation für eines oder mehrere Token des Ausdrucks vor. Für diese temporalen Ausdrücke mit Alignierungsinformation wurde getestet, ob mindestens eines der alignierten französischen Token auch Teil eines annotierten temporalen Ausdrucks ist. Im Jahrbuch 1960 war das für 17% aller Zeitausdrücke der Fall. Bei den temporalen Ausdrücken im Jahrbuch 1990 besitzen 72% der temporalen Ausdrücke ein oder mehrere Token, für die eine Alignierung gespeichert ist. Für 20% aller deutschen Zeitausdrücke im Jahrbuch 1990 gibt es alignierte französische Token oder Tokensequenzen, die ebenfalls Teil eines annotierten temporalen Ausdrucks sind.

Die Übereinstimmung von deutschen und französischen temporalen Ausdrücken im parallelen Teil von Text+Berg, getestet über die Alignierungen, liegt demnach bei

rund 18%. Wir begründen diesen Wert damit, dass der temporale Tagger für die französischen Texte nicht angepasst wurde für das Korpus, während der deutsche Tagger dank Anpassungsschritten korpuspezifischere Zeitausdrücke findet. Diese temporalen Ausdrücke wurden vom französischen System womöglich nicht erkannt, weshalb es weniger Entsprechungen von deutschen und französischen Annotationen gibt. Dazu kommt die Tatsache, dass auch die Präzision des deutschen Taggers für die partielle Extraktion temporaler Ausdrücke in den Jahrbüchern 1960 und 1990 nur bei rund 88% liegt. Demnach gibt es im Deutschen einige Annotationen, die keine Zeitausdrücke darstellen bzw. fälschlicherweise generiert wurden und es deshalb für diese temporalen Ausdrücke im französischen keine entsprechende Annotation gibt. Die Werte für die Übereinstimmungen werden zudem vom automatischen Alignierungssystem beeinflusst, das nicht durchgängig perfekte Alignierungen erbringt. Dennoch zeigen die Ergebnisse des Vergleichs der automatisch generierten Annotationen, dass es über die Nutzung der Alignierung Entsprechungen der deutschen und französischen temporalen Annotationen gibt. Diese Entsprechungen nutzten wir in einer zweiten Untersuchung, um die Präzision der automatischen Annotationen zu verbessern. Die Idee bei diesem Verfahren ist die Annahme, dass alle annotierten Ausdrücke in einer Sprache mit grosser Wahrscheinlichkeit einen relevanten Zeitausdruck darstellen, wenn sie eine Alignierung besitzen, die im parallelen Text der zweiten Sprache auch als Teil eines temporalen Ausdrucks annotiert wurde. Dieses Verfahren soll die Präzision der Annotationen verbessern. Die Evaluation der Annotationen in den französischen Texten weisen für die partielle Extraktion einen etwa 20 Prozentpunkte schlechteren Wert auf im Vergleich zur Präzision des deutschen Taggers. Für die strikte Präzision messen wir einen über 30 Prozentpunkte schlechteren Wert. Für die temporalen Ausdrücke im französischen Jahrbuch 1990 fällt die Präzision mit Werten von rund 44% (partiell) und 22% (strikt) für die Extraktion sogar sehr schlecht aus im Vergleich zu den Ergebnissen für die deutsche Annotation. Wir gehen davon aus, dass der französische Tagger viele Token mit einem `TIMEX3`-Tag versehen hat, welche keine Zeitausdrücke darstellen. Beispiel 5.4 zeigt eine falsche französische Annotation im Satz 83 aus Artikel 9 des französischen Jahrbuchs von 1991. Der Tagger annotierte die Höhenangaben *1448 m* und *1495 m* als Jahreszahlen.

(5.4) Elle va passer entre deux chaînons de moyennes montagnes , Igloo Mountain ( `<TIMEX3 tid="t168" type="DATE" value="1448">1448</TIMEX3>` m ) au nord et Cathedral Mountain ( `<TIMEX3 tid="t169" type="DATE" value="1495">1495</TIMEX3>` m ) au sud .

Auf Grund der Unterschiede der Präzision der deutschen und französischen Annotationen zielen wir mit der zweiten Untersuchung darauf ab, diese Werte durch die Nutzung der Alignierung zu verbessern. Mit einem Python-Skript vergleichen wir die Standoff-Dateien der temporalen Annotationen für die beiden parallelen Jahrbücher 1960 und 1990. Im ersten Durchlauf definieren wir das Deutsche als Ausgangssprache und das Französische als Zielsprache. Durch das Parsen der XML-Dateien können wir alle Tokensequenzen der deutschen und französischen temporalen Ausdrücke in entsprechenden Listen speichern. Anschliessend sucht das Programm für jedes Token in der Tokensequenz eines deutschen temporalen Ausdrucks nach einer Alignierungsinformation in der Alignierungsdatei. Wird für eines der Token des deutschen Zeitausdrucks eine Alignierung gefunden, so testet das Skript, ob eines der Token in der alignierten französischen Tokensequenz in der Liste der Spans für französische temporale Ausdrücke vorkommt. Wenn dieser Fall vorliegt, fügen wir die ID des betreffenden französischen temporalen Ausdrucks zur Liste der verifizierten französischen temporalen Ausdrücke hinzu. Ein temporaler Ausdruck gilt als verifiziert, wenn mindestens ein Token der deutschen Tokensequenz eine Alignierung zu einem französischen Token besitzt, welches ebenfalls Teil eines annotierten Zeitausdrucks ist. Am Ende erstellt das Programm eine neue Standoff-Datei für das französische Jahrbuch, welche lediglich die verifizierten temporalen Ausdrücke enthält. Da wir für die französischen Jahrbücher 1960 und 1990 bereits für vorhergehende Evaluationen jeweils 100 Sätze manuell annotiert haben, konnten wir diese Daten auch für die Evaluation der neu generierten Standoff-Dateien nutzen. Tabelle 12 zeigt die Evaluationsergebnisse für die verifizierten französischen temporalen Ausdrücke (Verifizierte Annotation) im Vergleich zu den originalen Werten der Annotation des französischen Taggers (Temporaler Tagger).

Das Ziel des Verfahrens war es, die Präzision zu verbessern und falsche französische Annotationen auszuschliessen. In den Evaluationsdaten der 100 Sätzen aus dem Jahrbuch 1960 wurden durch manuelle Arbeit 19 temporale Ausdrücke annotiert. Der französische Tagger fand im gleichen Text nur fünf Zeitausdrücke. Von diesen fünf Annotationen blieben nach der Verifizierung nur zwei temporale Ausdrücke übrig. Diese Annotationen sind jedoch in ihrer Tokensequenz sowie im `value`-Wert zu 100% exakt übereinstimmend mit der manuellen Annotation. In den 100 Sätzen des Jahrbuch 1990 konnten durch manuelle Annotation 25 Zeitausdrücke identifiziert werden. Der Tagger von Bittar (2009) kann nur neun temporale Ausdrücke in diesem Text finden. Nach dem Verifizierungsprozess bleiben fünf Annotationen übrig. Diese verifizierten Annotationen steigern die Präzision der Annotation für die 100 Sätze des Jahres 1990 um bis zu 18 Prozentpunkte.

Die Verifizierung der französischen temporalen Annotationen optimiert nur den Precision-Wert. Die Recall-Werte verschlechtern sich durch dieses Verfahren oder

| FRANZÖSISCH                    | partiell |      |      | strikt |      |      | value |      |      |
|--------------------------------|----------|------|------|--------|------|------|-------|------|------|
|                                | P        | R    | F    | P      | R    | F    | P     | R    | F    |
| <b>1960</b>                    |          |      |      |        |      |      |       |      |      |
| <b>Temporaler Tagger</b>       | 80,0     | 21,1 | 33,4 | 60,0   | 15,8 | 25,0 | 40,0  | 10,5 | 16,6 |
| <b>Verifizierte Annotation</b> | 100,0    | 10,5 | 19,0 | 100,0  | 10,5 | 19,0 | 100,0 | 10,5 | 19,0 |
| <b>1990</b>                    |          |      |      |        |      |      |       |      |      |
| <b>Temporaler Tagger</b>       | 44,4     | 16,0 | 23,5 | 22,2   | 8,0  | 11,8 | 22,2  | 8,0  | 11,8 |
| <b>Verifizierte Annotation</b> | 60,0     | 12,0 | 20,0 | 40,0   | 8,0  | 13,4 | 40,0  | 8,0  | 13,4 |
| <b>Total (1960+1990)</b>       |          |      |      |        |      |      |       |      |      |
| <b>Temporaler Tagger</b>       | 62,2     | 18,6 | 28,6 | 41,1   | 11,9 | 18,5 | 31,1  | 9,3  | 14,3 |
| <b>Verifizierte Annotation</b> | 80,0     | 11,3 | 19,8 | 70,0   | 9,3  | 16,4 | 70,0  | 9,3  | 16,4 |

Tabelle 12.: Überprüfung der französischen Annotationen über Alignierung mit deutschen Annotationen.

bleiben unverändert. Das F-Mass zeigt, dass der ausgewogene Durchschnitt zwischen Precision und Recall durch das Verfahren des Vergleichs und der Verifizierung meist verringert wird. Für die Evaluationsdaten aus dem Jahrbuch 1960 lag das F-Mass nach der Annotation durch den französischen Tagger bei 33,4. Nach der Verifizierung beträgt der Wert nur noch 19,0. Obwohl die französische Annotation nun deutlich präziser ist, werden weniger relevante Zeitausdrücke gefunden und die allgemeine Qualität der Annotation verschlechterte sich. Nur bei *value* zeigt das Vergleichsverfahren keinen Einfluss auf die Recall-Werte und die Verbesserung der Präzision steigert das F-Mass um einige Prozentpunkte.

Durch den Vergleich der deutschen temporalen Annotationen mit den französischen annotierten Zeitausdrücken können wir im Französischen viele falsche Annotationen ausschliessen. Im Rahmen der Untersuchung vergleichen wir auch die Französischen Annotationen mit den deutschen und führen einen Vergleich in die andere Richtung durch. Obwohl die Präzision des Taggers HeidelTime mit rund 88% für die partielle Extraktion deutlich höher liegt als für den französischen Tagger, überprüfen wir dennoch, ob wir durch das Vergleichen auch falsche deutsche temporale Annotationen ausschliessen können. Tabelle 13 zeigt die Ergebnisse für die verifizierten deutschen temporalen Ausdrücke. Für die Evaluation generieren wir neue Standoff-Dateien für die insgesamt 400 Sätze der Evaluationsdaten aus den Jahrbüchern 1960 und 1990. In die neuen Standoff-Dateien nehmen wir von allen für diese Texte durch HeidelTime generierten temporalen Ausdrücke nur diejenigen auf, welche durch Alignierung mit einem annotierten französischen temporalen Ausdruck ver-



bunden sind. Dabei muss nur mindestens ein Token des deutschen temporalen Ausdrucks mit mindestens einem Token eines französischen Zeitausdrucks aligniert sein. Den Typ des temporalen Ausdrucks und den Wert (`value`) betrachten wir dabei nicht. Die Ergebnisse aus Tabelle 13 zeigen, im Gegensatz zum Vergleichsverfahren von Deutsch nach Französisch, generell keine Verbesserung der Präzision der deutschen annotierten temporalen Ausdrücke. Nur die partielle Extraktion im Jahrbuch 1960 konnte um etwa 5 Prozentpunkte gesteigert werden. In allen anderen Fällen sinkt die Präzision. Die allgemeine Qualität der Annotationen, repräsentiert durch das F-Mass, sinkt enorm. Wir vermuten, dass ein solches Verfahren nur dann Verbesserungen der Präzision bringt, wenn die Annotationen in der Ausgangssprache bereits hohe Werte in diesem Bereich aufweisen und die Werte für die Zielsprache im Vergleich dazu tiefer ausfallen. Da die automatische Annotation im Französischen zum Teil Präzisionswerte von nur 22,2% erbringt (strikte Extraktion 1990), ist sie für einen Vergleich über Alignierungen und die Verifikation von temporalen Annotationen der Parallelsprache nicht anwendbar.

| DEUTSCH                        | partiell |      |      | strikt |      |      | value |      |      |
|--------------------------------|----------|------|------|--------|------|------|-------|------|------|
|                                | P        | R    | F    | P      | R    | F    | P     | R    | F    |
| <b>1960</b>                    |          |      |      |        |      |      |       |      |      |
| <b>Temporaler Tagger</b>       | 87,7     | 79,2 | 83,2 | 72,3   | 65,3 | 68,6 | 47,7  | 43,1 | 45,3 |
| <b>Verifizierte Annotation</b> | 92,9     | 18,3 | 30,6 | 64,3   | 12,7 | 21,1 | 42,9  | 8,5  | 14,2 |
| <b>1990</b>                    |          |      |      |        |      |      |       |      |      |
| <b>Temporaler Tagger</b>       | 87,5     | 76,7 | 81,7 | 78,1   | 68,5 | 73,0 | 50,0  | 43,8 | 46,7 |
| <b>Verifizierte Annotation</b> | 62,5     | 6,8  | 12,3 | 50,0   | 5,5  | 9,9  | 37,5  | 4,1  | 7,4  |
| <b>Total (1960+1990)</b>       |          |      |      |        |      |      |       |      |      |
| <b>Temporaler Tagger</b>       | 87,6     | 78,0 | 82,5 | 75,2   | 66,9 | 70,8 | 48,9  | 43,5 | 46,0 |
| <b>Verifizierte Annotation</b> | 77,7     | 12,6 | 21,7 | 57,2   | 9,1  | 15,7 | 40,2  | 6,3  | 10,9 |

Tabelle 13.: Überprüfung der deutschen Annotationen über Alignierung mit französischen Annotationen.

Für die Annotationen im Französischen bringt die Verifikation durch Vergleich deutliche Verbesserungen der Präzision. Betrachten wir jedoch die Precision-Werte für das Jahr 1990 in Tabelle 12, fällt auf, dass die Werte für `value` immer noch 20 Prozentpunkte tiefer liegen als für die partielle Extraktion. Das liegt vermutlich unter anderem daran, dass der französische Tagger für Uhrzeitangaben (`TIME`) keinen TimeML-konformen Wert im ISO-8601-Format `JJJJ-MM-TTThh:mm` generiert. Es wird jeweils nur die Uhrzeit im `value`-Attribut angegeben, nicht jedoch die Datumsangabe. Der deutsche Tagger HeidelbergTime hingegen generiert für Uhrzeiten immer einen vollständig TimeML-konformen Wert mit Datumsangabe. Die Uhrzeitangaben

der alignierten temporalen Ausdrücke stimmen im Deutschen und Französischen häufig überein. Das Beispiel in Tabelle 14 zeigt die Standoff-Annotation für den deutschen temporalen Ausdruck *Mittagsrast*<sup>3</sup>, welcher mit dem französischen Zeitausdruck *midi* aligniert ist. Die Uhrzeitangabe stimmt in den `value`-Attributen der beiden Annotationen überein.

Auf Grund von Beispielen wie dem aus Tabelle 14 modifizieren wir das Skript zur Verifizierung der französischen temporalen Annotationen in einem zweiten Schritt. Im Fall einer Alignierung eines deutschen und eines französischen temporalen Ausdrucks, die nicht den gleichen `value`-Wert aufweisen, übernehmen wir für den französischen temporalen Ausdruck den Wert des alignierten deutschen Zeitausdrucks. Das Ziel dabei ist die Verbesserung der Qualität der Angaben im französischen `value`-Attribut.

| Sprache | Annotation                                                                           |
|---------|--------------------------------------------------------------------------------------|
| DE      | <code>&lt;t type="TIME" span="6-98-3" id="t_15" value="1988-10-02T12:00"/&gt;</code> |
| FR      | <code>&lt;t type="TIME" span="6-100-5" id="t_2" value="T12:00"/&gt;</code>           |

Tabelle 14.: Beispiel-Annotation für die alignierten temporalen Ausdrücke *Mittagsrast* und *midi*.

| PRECISION   | Französisch             | partiell | strikt | value       |
|-------------|-------------------------|----------|--------|-------------|
| <b>1990</b> | Annotation              | 44,4     | 22,2   | <b>22,2</b> |
|             | Verifizierte Annotation | 60,0     | 40,0   | <b>20,0</b> |

Tabelle 15.: Überprüfung der französischen Annotationen über Alignierung mit deutschen Annotationen und Übernahme des deutschen `value` Attributs in den Evaluationssätzen aus Jahrbuch 1990.

Tabelle 15 zeigt die Veränderung der Precision-Werte für die Evaluationsdaten aus dem Jahrbuch von 1990, wenn wir bei entsprechender Alignierung mit einem deutschen temporalen Ausdruck den deutschen Wert für die französische Annotation übernehmen. Für die Sätze aus dem Jahrbuch von 1960 machen wir keine weiteren Untersuchung, da der Annotationsvergleich schon optimale Ergebnisse für `value` erzielen konnte. Wir vermuten, dass auf Grund der Unvollständigkeit vieler französischer Werte dieses Verfahren eine Verbesserung der Präzision von `value` in der Annotation der Sätze aus Jahrbuch 1990 erzielen kann. Die Werte für das

---

<sup>3</sup>Die Tatsache, dass der Ausdruck *Mittagsrast* neben einer Tageszeitangabe (*Mittags-*) auch die Zeitdauer *-rast* enthält, die als einzelnes Wort vom Tagger als `DURATION` erkannt wird, und den Zeitausdruck somit auf zwei verschiedene Arten interpretieren lässt, wird hier nicht näher betrachtet.

Jahrbuch 1990 zeigen jedoch, dass sich der Precision-Wert für `value` verschlechtert und sogar niedriger liegt als bei der ersten Annotation durch den französischen Tagger.

Allgemein führt das Filtern der durch das System von Bittar (2009) annotierten temporalen Ausdrücke zu deutlichen Verbesserungen der Präzision. Wir schliessen damit falsch annotierte Ausdrücke, die keinen alignierten deutschen Zeitausdruck besitzen, von der Annotation aus. Im nächsten Kapitel führen wir eine weitere Untersuchung durch, bei der wir die Alignierungen der parallelen Texte in Text+Berg zur Projektion von temporalen Annotationen nutzen.

### 5.3. Projektion der temporalen Annotationen

Im vorhergehenden Kapitel beschrieben wir einen Ansatz, der die Präzision eines mit temporalen Ausdrücken automatisch annotierten Korpus verbessern kann. Die Voraussetzung dafür war, dass ein paralleler Text in einer zweiten Sprache existiert, welcher ebenfalls automatisch annotiert wurde und insgesamt bessere Evaluationsergebnisse erzielte als der zu verbessernde Text. In diesem Kapitel untersuchen wir Methoden, wie die Ausbeute bzw. der Recall-Wert eines automatischen Annotationssystems im Nachhinein verbessert werden kann, wenn ein fremdsprachiger Paralleltext mit temporalen Annotationen vorliegt. Grundlage hierfür bilden ebenfalls die Wortalignierungen der beiden Texte.

Ein interessanter Ansatz zur Generierung eines temporal annotierten Korpus unter Verwendung von Alignierungen kommt von Costa und Branco (2010). Dieser Ansatz arbeitet nicht mit zwei parallelen Texten, sondern passt einen mit temporalen Ausdrücken annotierten Text der Ausgangssprache halb-automatisch für die Zielsprache an. Im Fall von Costa und Branco (2010) soll ein TimeML-konform annotiertes englisches Korpus für das Portugiesische adaptiert werden. Das Ziel dieses Verfahrens ist es, möglichst schnell ein semantisch annotiertes Korpus für eine Sprache zu erhalten, für die noch keine Trainings- oder Evaluationsdaten dieser Art verfügbar sind. Costa und Branco (2010) verwenden dazu die englischen Daten aus TempEval-1, welche ein bereits TimeML-konform annotiertes Korpus darstellen. Im ersten Schritt entfernen sie alle TimeML-Annotationen aus dem englischen Text. Der Rohtext wird mit dem Google Translator Toolkit ins Portugiesische übersetzt und von einem Übersetzer manuell korrigiert. Durch Beibehalten der Zeilenumbrüche alignieren sie jeden parallelen Paragraphen des annotierten englischen Ausgangstext, des englischen Rohtextes sowie des übersetzten portugiesischen Rohtextes. So können die temporalen Annotationen bzw. das TimeML-Markup aus dem englischen Original-

text für jeden Paragraph bestimmt und anschliessend automatisch ans Ende jedes portugiesischen Paragraphen angehängt werden. Im letzten Schritt verteilen Costa und Branco (2010) die Markup-Annotationen für Zeitausdrücke am Ende jedes Paragraphen manuell auf die portugiesischen Token. Dabei werden einige Attribute manuell verändert. Diese Methode ist für kleine Korpora schnell, braucht keine existierenden Trainingsdaten oder Systeme, es werden nur wenige humane Annotatoren benötigt und man erhält am Ende ein zum Ausgangstext paralleles Korpus (Costa und Branco 2010). Wenn die Annotationen im Ausgangstext von guter Qualität sind, ist dies eine Möglichkeit, ohne einen angepassten Tagger möglichst viele temporale Ausdrücke in einem Paralleltext zu bestimmen. Wir sind jedoch der Meinung, dass diese Methode durch den Zeitaufwand für die manuellen Arbeiten bei grossen Korpora wie dem Text+Berg-Korpus nicht anwendbar ist.

Spreyer und Frank (2008) liefern einen Ansatz zur Generierung von temporalen Annotationen in einem Paralleltext durch Verwendung von Alignierungsinformationen. Das Ziel ihres Verfahrens ist es, die automatisch annotierten TimeML-Tags eines englischen Texts über die Wortalignierung auf die parallele deutsche Übersetzung zu übertragen. Die Basis eines solchen Projektionsverfahrens, so Spreyer und Frank (2008), bilden qualitativ-hochwertige Annotationen in der Ausgangssprache. Da sie die Annotationen in der Ausgangssprache sowie die Wortalignierungen im parallelen Korpus automatisch generieren, rechnen sie bei der Methode der Annotationsübertragung mit Fehlern. Um diese Probleme zu minimieren, setzen sie eine Reihe von Modulen mit zusätzlichem Wissen ein, welches das System der Projektion unterstützt, die richtigen Token für den temporalen Ausdruck in der Zielsprache zu identifizieren.

Als eine linguistische Beschränkung setzen sie den 'non-content'-Filter ein. Dieser bestimmt auf der Basis von POS-Tags, dass die Alignierung eines Zeitausdrucks der Ausgangssprache mit einer inhaltslosen Wortart in der Zielsprache als ungültig betrachtet werden kann. Als inhaltslose Wortarten definieren Spreyer und Frank (2008) Konjunktionen, Präpositionen und Satzzeichen. Zusätzlich sehen sie Alignierungen mit Token in der Zielsprache, welche kein Lemma besitzen, als ebenfalls ungültig an<sup>4</sup>.

Die Alignierungsinformationen können für einen temporalen Ausdruck in der Ausgangssprache auf eine Sequenz von Token in der Zielsprache verweisen, die nicht zusammenhängend ist. Spreyer und Frank (2008) nehmen an, dass eine unzusammenhängende Tokensequenz für einen Zeitausdruck sehr unwahrscheinlich ist. Deshalb führen sie für die Projektion der temporalen Ausdrücke eine Beschränkung ein, welche die alignierten Tokensequenzen modifiziert, sodass in der Zielsprache nur

---

<sup>4</sup>Wenn die Einheiten in Ausgangs- und Zielsprache beide kein Lemma besitzen, so gilt die Alignierung als gültig.

zusammenhängende Token annotiert werden. Bei dieser Beschränkung gibt es zwei Möglichkeiten: Wenn eine nicht zusammenhängende Menge an Token vorliegt, wird die längste zusammenhängende Tokenkette ausgewählt und somit andere einzelne Token oder kürzere Tokensequenzen von der Annotation ausgeschlossen (Option Ausschluss). Die zweite Möglichkeit, die gegensätzlich zur ersten arbeitet, füllt die Lücken zwischen den unzusammenhängenden Tokensequenzen und betrachtet alle Token, welche zwischen den unzusammenhängenden Tokensequenzen liegen, als Teil der Annotation des temporalen Ausdrucks in der Zielsprache (Option Auffüllen). Neben dem linguistischen Filter und den Filtern Ausschluss und Auffüllen zur Verbesserung der Projektion bestimmen Spreyer und Frank (2008) ebenfalls eine Obergrenze für die Anzahl an Token der Zielsprache, zu welchen ein einzelnes Token der Ausgangssprache aligniert sein darf. Sie trainieren und evaluieren ihr System anhand eines parallelen englisch-deutschen Subkorpus des Europarl-Korpus. Die automatischen temporalen Annotationen im englischen Teil generieren sie mit dem TARSQI-Toolkit<sup>5</sup>. Als Evaluationsdaten verwenden sie 236 manuell annotierte parallele Sätze. Für die automatischen englischen Annotationen der temporalen Ausdrücke erreichen sie einen Wert von 80.56% für das F-Mass. Sie untersuchen die Projektion von TimeML-konformen Annotationen für `EVENT`, `SLINK` und `TIMEX3`-Tags, wobei sie verschiedene Kombinationen der Filtermodule anwenden. Generell stellen sie fest, dass bei der Projektion vom Englischen ins Deutsche die `TIMEX3`-Tags am besten abschneiden. Dabei erreicht die Projektion von temporalen Ausdrücken den Precision-Wert von 71,01 und den Recall-Wert von 52,76. Diese Werte wurden erreicht, wenn als Filter die Option Ausschluss gewählt wurde, zusammen mit einer Obergrenze von maximal zwei alignierten, zielsprachlichen Token pro einzeltem Ausgangstoken. Bei der Evaluation der automatisch durch Projektion generierten temporalen Ausdrücke im deutschen Teil des Korpus werden keine Angaben gemacht, ob sich die Daten auf eine strikte oder partielle Extraktion beziehen oder inwiefern der `value`-Wert betrachtet wird.

Die Methode der Projektion von Spreyer und Frank (2008) setzt Korpusdaten voraus, wie sie auch im Text+Berg-Korpus vorhanden sind. Wir verfügen über ein paralleles deutsch-französisches Korpus, das für jedes Token Metadaten über Lemma und POS-Tag gespeichert hat. Zudem existieren bereits optimierte Alignierungsinformationen sowie automatisch generierte temporale Annotationen im deutschen Teil des Korpus. Aus diesem Grund und speziell zur Verbesserung des Recall-Werts der automatischen Annotation in den französischen Texten, wenden wir die Projektionsmethode von Spreyer und Frank (2008) auch auf den parallelen Teil von Text+Berg an.

---

<sup>5</sup><http://www.timeml.org/site/tarsqi/>

Die Basis für diese Untersuchung bilden die deutschen Standoff-Dateien der temporalen Annotation durch HeidelTime für die Jahrbücher 1960 und 1990. Diese Annotation im Deutschen erreichte für die partielle Extraktion im Durchschnitt Precision-Werte von rund 88,0 und Recall-Werte von etwa 78,0 für diese beiden Jahrbücher. Zusätzlich nutzen wir für die Projektion die Alignierungsinformationen, welche für das Release 147v02 von Text+Berg zur Verfügung stehen. Für jedes Jahr erstellen wir eine Datei mit allen für dieses Jahrbuch relevanten Alignierungsinformationen. Zum Einsetzen des linguistischen Filters während der Projektionsselektion verwenden wir zudem die parallelen originalen französischen Korpusdateien der betreffenden Jahrbücher.

Im ersten Schritt erstellen wir ein Python-Skript, welches für alle deutschen temporalen Ausdrücke die alignierten französischen Token oder Tokensequenzen ausgibt, falls für ein oder mehrere Token des betreffenden Zeitausdrucks eine Alignierung vorliegt. Um die Liste der alignierten französischen Token später den temporalen Ausdrücken zuordnen zu können, verwenden wir die ID der deutschen Zeitausdrücke aus der Standoff-Datei auch für die alignierten französischen Token. Um die alignierten französischen Token für einen deutschen Zeitausdruck zu identifizieren, testen wir für jedes deutsche Token, ob eine 1:1 oder eine 1:m Alignierung vorliegt. Wenn der deutsche temporale Ausdruck aus mehreren Token besteht, testen wir anschließend, ob für eine Subsequenz der gesamten Tokensequenz des Zeitausdrucks eine Alignierungsinformation vorliegt. So können wir auch m:1 oder m:m Alignierung identifizieren. Die so generierten Listen mit den IDs der temporalen Ausdrücke als Schlüssel und den alignierten französischen Token als Wert verwenden wir im Anschluss zum Filtern der relevanten temporalen Ausdrücke. Die Beispiele 5.5, 5.6 und 5.7 zeigen drei Zeilen aus der generierten Liste mit den Alignierungsinformationen für drei deutsche temporalen Ausdrücke. Die deutschen temporalen Ausdrücke sowie die alignierten französischen Token sind in den Beispielsätzen unterstrichen.

(5.5) t\_1526 65-145-3:4

DE: Ob auch der Inhalt des Rucksackes durchnässt ist , wird sich später erweisen .

FR: On verra plus tard si l' intérieur du sac est trempé .

(5.6) t\_1702 11-3-2:2 11-3-11:12

DE: Nach den letzten , etwas anstrengenden Tagen , geniessen wir alle die zwei Ruhetage im Basislager .

FR: Après deux journées assez fatigantes , nous apprécions un jour de repos au camp de base .

(5.7)  $\tau_{1729}$  16-14-14:14 16-14-16:16

DE: Wir wollten nicht riskieren , die mageren 10 Tage , die uns zum eigentlichen Bergsteigen verblieben , im Zelt mit Ins-Schneetreiben-Hinausstarren zu verbringen .

FR: Nous ne voulons pas courir le risque de passer sous la tente les dix pauvres jours disponibles pour les ascensions , à nous morfondre en contemplant les bourrasques de neige .

Der Zeitausdruck  $\tau_{1526}$  in 5.5 ist mit den französischen Token 3 und 4 aus Satz 145 in Artikel 65 aligniert, d.h. der Ausdruck *später* entspricht dem französischen *plus tard*. Der deutsche annotierte temporale Ausdruck  $\tau_{1702}$  in 5.6 ist mit den drei französischen Token 2, 11 und 12 aus Satz 3 in Artikel 11 aligniert. Diese Tokensequenz ist nicht kontinuierlich und wird im Anschluss durch die Filtermethoden weiterverarbeitet. Interessant ist jedoch, dass die falsche Alignierung, welche hier vorliegt, wahrscheinlich durch einen humanen Übersetzungsfehler entstanden ist, denn *zwei Ruhetage* wurde im französischen mit *un jour de repos* übersetzt. Das Beispiel 5.7 zeigt den deutschen Zeitausdruck  $\tau_{1729}$ , welcher mit den Token 14 und 16 aus dem Satz 14 des französischen Artikels 16 aligniert ist. Auch hier finden wir eine diskontinuierliche Tokensequenz. Allerdings liegt hier keine falsche Alignierung oder ein Übersetzungsfehler vor, wie die Sätze aus dem Korpus zeigen. Die französische Syntax unterscheidet sich von der deutschen und stellt das Adjektiv im französischen zwischen das Zahlwort und das Nomen. Auch diese Sequenz wird anschliessend durch die Filtermethoden weiterverarbeitet. Diese syntaktischen Konstruktionen erklären möglicherweise, warum wir den Filter „Auffüllen“ für die Projektion vom Deutschen in das Französische besonders erfolgreich einsetzen konnten (siehe folgende Kapitel), obwohl diese Methode für Alignierungen wie in 5.6 falsche Projektionen erzeugen kann.

Als erste Option zum Filtern von relevanten französischen Alignierungen setzten wir die von Spreyer und Frank (2008) als „konservativer Ansatz“ benannte Methode ein. Dabei schliessen wir alle zu einem deutschen Zeitausdruck alignierten französischen Tokensequenzen aus, welche keine kontinuierliche Sequenz darstellen. Dazu betrachten wir alle französischen Alignierungen der im ersten Schritt generierten Liste, um für jeden deutschen temporalen Ausdruck eine aufsteigend sortierte Liste ohne Duplikate mit französischen Token zu erstellen. Anschliessend fügen wir jeden temporalen Ausdruck mit kontinuierlicher Tokensequenz zu einer neuen französischen Standoff-Datei für das betreffende Jahrbuch hinzu.

Als weitere Optionen zum Filtern wendeten wir die zwei Kriterien „Ausschluss“ und „Auffüllen“ aus Spreyer und Frank (2008) an. In beiden Fällen betrachten wir die diskontinuierlichen Tokensequenzen. Der Filter „Ausschluss“ sucht in diesen ali-

gnierten französischen Sequenzen die längste zusammenhängende Subsequenz. Diese wird als französischer Zeitausdruck definiert, während wir alle anderen alignierten Token, die nicht Teil dieser Subsequenz sind, ausschliessen und diese somit weglassen. Werden mehrere gleichlange Subsequenzen gefunden, definiert das Programm die erste gefundene Tokensequenz als französischen temporalen Ausdruck.

Der zweite topologische Filter „Auffüllen“ arbeitet auf entgegengesetzte Weise. Er betrachtet ebenfalls alle alignierten französischen Tokensequenzen, die nicht kontinuierlich sind. Zur Herstellung der Kontinuität in den Sequenzen füllt der Filter die Lücken in der Tokensequenz und fügt Token hinzu. Im betreffenden Programm betrachten wir in den sortierten Listen mit den alignierten französischen Token das erste Token und das letzte Token der Sequenz. Sie gelten als Start- und Endpunkt der Tokensequenz. Alle dazwischen liegenden Token fügen wir automatisch hinzu und definieren die gesamte Sequenz als kontinuierlichen französischen Zeitausdruck. Für beide Filter (Ausschluss und Auffüllen) generieren wir für jedes Jahrbuch eine neue Standoff-Datei mit allen französischen temporalen Ausdrücken und deren Tokenspannen.

Als vierten Filter testen wir den linguistischen Ansatz nach Spreyer und Frank (2008). Hier betrachten wir Alignierungen zu Konjunktionen, Präpositionen, Satzzeichen oder zu Token mit unbekanntem Lemma als nicht relevant. Im Fall der Annotationsprojektion in Text+Berg holen wir für jedes zu einem deutschen Zeitausdruck alignierte französische Token Informationen aus der originalen französischen Korpusdatei ein. Mit den Annotationen im Korpus und über die Angaben der Tokenposition können wir die Wortart und das Lemma für ein Token auf einfache Weise identifizieren. Handelt es sich dabei um ein POS-Tag der oben erwähnten Kategorien oder um ein Lemma mit der Kennzeichnung „unk“, schliessen wir das Token als relevante Alignierung aus. Bei der anschliessenden Generierung der Standoff-Dateien fügen wir für jeden temporalen Ausdruck nur relevante Token zu den Spannen hinzu.

Mit den vier Filterschritten „Konservativer Ansatz“ „Ausschluss“ „Auffüllen“ und „Linguistik“ erzeugen wir vier unterschiedliche neue Standoff-Dateien für jedes Jahrbuch. Für alle alignierten und gefilterten französischen temporalen Ausdrücke übernehmen wir das `value`-Attribut des deutschen Zeitausdrucks. Aus den neuen Standoff-Dateien nehmen wir zur Evaluation alle Annotationen für die Sätze der betreffenden Evaluationsdaten heraus. Die Evaluationsdaten umfassen für die Jahrbücher 1960 und 1990 jeweils 100 zusammenhängende, manuell annotierte französische Sätze. Tabelle 16 zeigt alle Precision- und Recall-Werte und das F-Mass für die partielle und strikte Extraktion sowie für das `value` Attribut, geordnet nach Jahren.

In Spreyer und Frank (2008) wird zusätzlich der Filter der Alignierungsselektion vorgestellt. Dabei werden schon im ersten Schritt bei der Identifizierung aller Ali-



gnierungen für einen temporalen Ausdruck der Ausgangssprache nur bestimmte Alignierungen zugelassen. Es wird eine Obergrenze für die Anzahl an Token in der Zielsprache bestimmt, zu denen ein einzelnes Token aus der Ausgangssprache aligniert sein darf. Spreyer und Frank (2008) schreiben, dass sie mit der Beschränkung 1:2 und anschließendem Filtern besonders gute Ergebnisse für temporale Ausdrücke erzielen. Für die Annotationsprojektion in Text+Berg wendeten wir diese Beschränkung ebenfalls vor jedem Filterschritt an. Diese Beschränkung veränderte die Evaluationsergebnisse nur in einem Fall leicht, in allen anderen Fällen gar nicht. Daher nahmen wir diesen Schritt nicht in die Annotationsprojektion auf. Wir vermuten, dass im Text+Berg-Korpus nur wenige Alignierungen für temporale Ausdrücke zu finden sind, die die Grenze von 1:2 überschreiten. Deshalb hat die Beschränkung wenig bis keine Auswirkung.

In Tabelle 16 sind die Evaluationsergebnisse der Projektion aufgelistet. Die Werte sind nach Jahren geordnet und es werden die einzelnen Filter unterschieden. Zum Vergleich präsentiert die letzte Zeile die Precision-, Recall und F-Mass-Werte von Spreyer und Frank (2008) für ihre Projektion von Englisch nach Deutsch.

Für das Jahr 1990 erzeugt die Projektion deutlich bessere Ergebnisse als das in Kapitel 4 verwendete automatische temporale Annotationssystem von Bittar (2009), in der Tabelle als „Temporaler Tagger“ benannt. Da es sich um ein relativ kleines Evaluationskorpus mit nur jeweils 100 manuell annotierten Sätzen pro Jahrbuch handelt, gleichen sich die Ergebnisse für die verschiedenen Filter in vielen Fällen. Die Projektion erzielt allgemein bessere Precision- als Recall-Werte. Dennoch finden wir durch das Projektionsverfahren für das Jahr 1990 mehr als doppelt so viele relevante Zeitausdrücke als durch den temporalen Tagger. Für die strikte Extraktion steigen die Recall-Werte um 20 Prozentpunkte, für die partielle Extraktion im besten Fall sogar um 24 Prozentpunkte an. Den richtigen Wert für das `value`-Attribut zu definieren stellt ein grösseres Problem dar. Hier liegen die Recall-Werte nach den Projektionsschritten für das Jahr 1990 nur wenig oberhalb der Werte des temporalen Taggers oder können, wie im Fall des konservativen Ansatzes, durch die Projektion nicht verbessert werden.

Im Jahrbuch 1960 hingegen verbessern sich die Recall-Werte für `value` durch das Projektionsverfahren um rund die Hälfte. Auch die Relevanz der gefundenen temporalen Ausdrücke kann durch Projektion deutlich verbessert werden. Die Recall-Werte liegen hier für die strikte Extraktion bei rund 29,0 und für die partielle Extraktion bei etwa 66,0. Die Projektion kann hier die Recall-Werte des temporalen Taggers fast verdreifachen. Im Bereich der Präzision unterscheiden sich die Ergebnisse für 1960 stark von den Ergebnissen für 1990. Der temporale Tagger erreichte bereits hohe Precision-Werte für das Jahrbuch 1960 und das Projektionsverfahren konnte diese Werte für die partielle Extraktion nur leicht verbessern. Für die strikte Extraktion

fällt die Präzision durch Projektion deutlich schlechter aus als die des temporalen Taggers.

| FRANZÖSISCH                     | partiell |      |      | strikt |       |       | value |      |      |
|---------------------------------|----------|------|------|--------|-------|-------|-------|------|------|
|                                 | P        | R    | F    | P      | R     | F     | P     | R    | F    |
| <b>1960</b>                     |          |      |      |        |       |       |       |      |      |
| <b>Temporaler Tagger</b>        | 80,0     | 21,1 | 33,4 | 60,0   | 15,8  | 25,0  | 40,0  | 10,5 | 16,6 |
| <b>Projektion</b>               |          |      |      |        |       |       |       |      |      |
| Konservativer Ansatz            | 85,7     | 63,2 | 72,6 | 35,7   | 26,3  | 30,3  | 28,6  | 21,1 | 24,3 |
| Filter Ausschluss               | 86,7     | 68,4 | 76,5 | 33,3   | 26,3  | 29,4  | 26,7  | 21,1 | 23,6 |
| Filter Auffüllen                | 81,3     | 68,4 | 74,3 | 37,5   | 31,6  | 34,3  | 25,0  | 21,1 | 22,9 |
| Filter Linguistik               | 80,0     | 63,2 | 70,6 | 40,0   | 31,6  | 35,3  | 26,7  | 21,1 | 23,6 |
| <b>1990</b>                     |          |      |      |        |       |       |       |      |      |
| <b>Temporaler Tagger</b>        | 44,4     | 16,0 | 23,5 | 22,2   | 8,0   | 11,8  | 22,2  | 8,0  | 11,8 |
| <b>Projektion</b>               |          |      |      |        |       |       |       |      |      |
| Konservativer Ansatz            | 100,0    | 32,0 | 48,5 | 87,5   | 28,0  | 42,4  | 25,0  | 8,0  | 12,1 |
| Filter Ausschluss               | 90,0     | 36,0 | 51,4 | 70,0   | 28,0  | 40,0  | 30,0  | 12,0 | 17,1 |
| Filter Auffüllen                | 100,0    | 40,0 | 57,1 | 70,0   | 28,0  | 40,0  | 30,0  | 12,0 | 17,1 |
| Filter Linguistik               | 100,0    | 40,0 | 57,1 | 70,0   | 28,0  | 40,0  | 30,0  | 12,0 | 17,1 |
| <b>Total (1960+1990)</b>        |          |      |      |        |       |       |       |      |      |
| <b>Temporaler Tagger</b>        | 62,2     | 18,6 | 28,6 | 41,1   | 11,9  | 18,5  | 31,1  | 9,3  | 14,3 |
| <b>Projektion</b>               |          |      |      |        |       |       |       |      |      |
| Konservativer Ansatz            | 92,9     | 47,6 | 62,9 | 61,6   | 27,2  | 37,7  | 26,8  | 14,6 | 18,9 |
| Filter Ausschluss               | 88,4     | 52,2 | 65,6 | 51,7   | 27,2  | 35,6  | 28,4  | 16,6 | 21,0 |
| Filter Auffüllen                | 90,7     | 54,2 | 67,9 | 53,8   | 29,8  | 38,4  | 27,5  | 16,6 | 20,7 |
| Filter Linguistik               | 90,0     | 51,6 | 65,6 | 55,0   | 29,8  | 38,7  | 28,4  | 16,6 | 21,0 |
| <b>Spreyer und Frank (2008)</b> |          |      |      |        |       |       |       |      |      |
| <b>Projektion EN nach DE</b>    |          |      |      | 71,01  | 52,76 | 60,54 |       |      |      |

Tabelle 16.: Annotation französischer temporaler Ausdrücke durch Projektion. Evaluation (P = Precision, R = Recall, F = F-Mass) für 200 Sätze aus den Jahrbüchern 1960 und 1990. Als Vergleich sind die für Zeitausdrücke optimierten Werte von Spreyer und Frank (2008) angegeben.

In Spreyer und Frank (2008) geben die Autoren nicht an, auf welche Art der Extraktion sich ihre Evaluationsergebnisse beziehen. Wir gehen davon aus, dass die Werte die strikte Extraktion betreffen, weshalb diese mit den entsprechenden Werten für das Text+Berg-Korpus verglichen werden. Insgesamt liegt der Precision-Wert der besten Ergebnisse aus den Jahren 1960 und 1990 für das Projektionsverfahren im parallelen Teil von Text+Berg bei 63,8. Der Durchschnitt für den Recall-Wert der besten Werte liegt bei 29,8. Spreyer und Frank (2008) erzielen mit einem optimierten

System für TIMEX3-Annotationen den Precision-Wert von 71,01 und den Recall-Wert von 52,76. Die Ergebnisse im Text+Berg-Korpus fallen schlechter aus, zeigen jedoch im Vergleich mit den Ergebnissen des französischen temporalen Taggers Verbesserungen bezüglich der Qualität der Annotation auf.

Das Verfahren der Projektion von temporalen Ausdrücken in einem parallelen Korpus durch die Nutzung von Alignierungsinformationen setzt eine bestehende solide temporale Annotation in der Ausgangssprache voraus. Da wir die automatische Annotation im französischen Teil des Korpus als nicht ausreichend ansehen, um die Projektion von französischen temporalen Ausdrücken in das Deutsche zu testen, führen wir das Projektionsverfahren für die Sprachrichtung Französisch-Deutsch nicht durch.

## **5.4. Kombiniertes Verfahren zur Verbesserung der französischen Annotation**

Mit dem Vergleich der deutschen und französischen temporalen Annotationen aus Kapitel 5.2 konnten wir zeigen, dass die Qualität eines temporalen Taggers, welcher nicht für das betreffende Korpus angepasst oder trainiert wurde, durch den Ausschluss möglicher falscher Annotationen verbessert werden kann. Die Projektion von automatisch generierten temporalen Annotationen in einem alignierten parallelen Korpus aus Kapitel 5.3 zeigte, dass mit einer soliden automatischen Annotation einer Sprache im Korpus auch für die parallele zweite Sprache Annotationen generiert werden können. Während das erste Verfahren des Vergleichs hauptsächlich die Präzision verbessern konnte, wurden mit der zweiten Methode der Projektion deutliche Verbesserungen der Ausbeute bzw. der Recall-Werte erzielt. Im letzten Teil dieses Kapitels testen wir nun, ob die Kombination der beiden Verfahren dazu führen kann, die besten Evaluationsergebnisse für die automatische temporale Annotation im Französischen zu erreichen.

Dazu kombinieren wir die Standoff-Dateien der französischen Annotationen des Vergleichsverfahrens und der Projektionsmethode. Als Filter für die Projektion wählen wir die Option „Auffüllen“ da diese für die Jahrbücher 1960 und 1990 die besten Recall-Werte erzeugen konnte. Mit einem Python-Skript durchlaufen wir die Standoff-Datei dieser Option der Projektionsmethode für beide Jahrbücher. Für jede Annotation eines temporalen Ausdrucks überprüft das Programm, ob eines der Token der Spanne in der Standoff-Datei des Vergleichsverfahrens vorkommt. Wenn dies der Fall ist, nehmen wir die Annotation für diesen temporalen Ausdruck aus der Datei des Vergleichsverfahrens, um sie in eine neue Datei zu speichern. Die Idee

dahinter ist, dass die Annotationen in den Standoff-Dateien des Vergleichsverfahrens vom französischen temporalen Tagger stammen. Wir sehen diese Zeitausdrücke als präziser an. Finden wir in der Datei des Vergleichsverfahrens keinen temporalen Ausdruck mit diesem Token, übernehmen wir für den temporalen Ausdruck die Annotation aus der Projektion. Anschliessend fügen wir alle Annotationen aus der Datei des Vergleichsverfahrens, welche noch nicht in der neuen Standoff-Datei enthalten sind, ebenfalls dort hinzu. So generiert das Programm eine neue, kombinierte Standoff-Datei.

Tabelle 17 zeigt die Evaluationsergebnisse für das kombinierte Verfahren, im Vergleich zu den Ergebnissen des automatischen französischen temporalen Taggers, der verifizierten Annotation durch Vergleich mit den deutschen Annotationen und den Ergebnissen für die temporalen französischen Annotationen als Ergebnis der Projektion von deutschen Annotationen ins Französische.

| FRANZÖSISCH              | partiell    |             |             | strikt      |             |             | value       |             |             |
|--------------------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
|                          | P           | R           | F           | P           | R           | F           | P           | R           | F           |
| <b>1960</b>              |             |             |             |             |             |             |             |             |             |
| Temporaler Tagger        | 80,0        | 21,1        | 33,4        | 60,0        | 15,8        | 25,0        | 40,0        | 10,5        | 16,6        |
| Verifizierte Annotation  | 100,0       | 10,5        | 19,0        | 100,0       | 10,5        | 19,0        | 100,0       | 10,5        | 19,0        |
| Projektion (Auffüllen)   | 81,3        | 68,4        | 74,3        | 37,5        | 31,6        | 34,3        | 25,0        | 21,1        | 22,9        |
| <b>Kombination</b>       | <b>81,3</b> | <b>68,4</b> | <b>74,3</b> | <b>37,5</b> | <b>31,6</b> | <b>34,3</b> | <b>25,0</b> | <b>21,1</b> | <b>22,9</b> |
| <b>1990</b>              |             |             |             |             |             |             |             |             |             |
| Temporaler Tagger        | 44,4        | 16,0        | 23,5        | 22,2        | 8,0         | 11,8        | 22,2        | 8,0         | 11,8        |
| Verifizierte Annotation  | 60,0        | 12,0        | 20,0        | 40,0        | 8,0         | 13,4        | 40,0        | 8,0         | 13,4        |
| Projektion (Auffüllen)   | 100,0       | 40,0        | 57,1        | 70,0        | 28,0        | 40,0        | 30,0        | 12,0        | 17,1        |
| <b>Kombination</b>       | <b>85,7</b> | <b>48,0</b> | <b>61,5</b> | <b>57,1</b> | <b>32,0</b> | <b>41,0</b> | <b>28,6</b> | <b>16,0</b> | <b>20,5</b> |
| <b>Total (1960+1990)</b> |             |             |             |             |             |             |             |             |             |
| Temporaler Tagger        | 62,2        | 18,6        | 28,6        | 41,1        | 11,9        | 18,5        | 31,1        | 9,3         | 14,3        |
| Verifizierte Annotation  | 80,0        | 11,3        | 19,8        | 70,0        | 9,3         | 16,4        | 70,0        | 9,3         | 16,4        |
| Projektion (Auffüllen)   | 90,7        | 54,2        | 67,9        | 53,8        | 29,8        | 38,4        | 27,5        | 16,6        | 20,7        |
| <b>Kombination</b>       | <b>83,5</b> | <b>58,2</b> | <b>68,6</b> | <b>47,3</b> | <b>31,8</b> | <b>38,0</b> | <b>26,8</b> | <b>18,6</b> | <b>22,0</b> |

Tabelle 17.: Kombination der Verfahren Vergleich und Projektion zur Annotation von französischen temporalen Ausdrücken.

Wir stellen fest, dass sich die Ergebnisse für das Kombinationsverfahren für das Jahr 1960 nicht von den Projektionsergebnissen unterscheiden. Das Hinzufügen der verifizierten Annotationen zu den durch Projektion generierten Annotationen hat keinen Einfluss auf die Qualität der 100 Sätze aus dem Jahrbuch 1960. Dies zeigt uns, dass die Alignierungen für temporale Ausdrücke von guter Qualität sind. Für

das Jahr 1990 messen wir bei der Kombination eine Verbesserung. Das Kombinationsverfahren erzielt hier die höchsten Recall-Werte für die automatische temporale Annotation im Französischen für die partielle und strikte Extraktion, sowie für den *value*-Wert. Wir konnten den anfänglichen, durch den temporalen Tagger erbrachten Wert von 8,0 für die strikte Extraktion vervierfachen. Die Precision-Werte für die Extraktion sind bei der Kombination besser als für die verifizierte Annotation, jedoch etwas schlechter als bei der Projektion. Der Precision-Wert für *value*, den wir durch das verifizierte Verfahren fast verdoppeln konnten, liegt nach der Kombinationsmethode mit 28,6 wieder in etwa im Bereich des temporalen Taggers.

Die Ergebnisse zeigen, dass wir die Qualität der Annotationen durch die Kombination verbessern konnten. Das F-Mass der strikten Extraktion zeigt mit 38,0 nun einen rund doppelt so hohen Wert wie der französische temporale Tagger. Bei der partiellen Extraktion wurde mit 68,6 ein Wert erreicht, der 40 Prozentpunkte höher liegt als zuvor beim temporalen Tagger.

Die Werte liegen insgesamt unter den Ergebnissen, welche der angepasste temporale Tagger HeidelbergTime für deutsche Texte im Text+Berg-Korpus erzielte. Jedoch weisen die vorgestellten Verfahren der automatischen Annotation in einem parallelen Korpus, welche alle auf die Verfügbarkeit von Alignierungen zurückgreifen, eine deutliche Verbesserung der Annotation des französischen Taggers nach. Dabei mussten wir keine Änderungen am Tagger selbst vornehmen.

# 6. Konklusion

## 6.1. Zusammenfassung und Ergebnisse

In dieser Arbeit beschäftigten wir uns mit der automatischen Annotation von deutschen und französischen temporalen Ausdrücken im Text+Berg-Korpus. Unser Ziel dabei war es, ein Modul bereitzustellen, das Informationen über deutsche und französische Zeitausdrücke im Korpus enthält.

Als Annotationsrichtlinie verwendeten wir den TimeML-Standard. In Kapitel 2 beschrieben wir diesen und betrachteten ihn kritisch. Wir erörterten mehrere Verwendungsmöglichkeiten für standardisierte temporale Annotationen im Text+Berg-Korpus. Vor allem im Rahmen der Extraktion von alpinen Exkursionsbeschreibungen können wir mit der temporalen Annotation zur zeitlichen Situierung der alpinen Routen beitragen. Mit der Darstellung der Rolle der Textdomäne bei der temporalen Annotation konnten wir zeigen, dass das Text+Berg-Korpus mit seinen alpinen Texten nicht den typischen Trainingskorpora für automatische temporale Annotationssysteme entspricht. Wir stellten fest, dass die Extraktion und Auswertung von Zeitausdrücken abhängig ist von der Domäne des Texts.

Um deutsche Zeitausdrücke im Text+Berg-Korpus zu annotieren, verwendeten wir in Kapitel 3 den temporalen Tagger HeidelbergTime (Strötgen und Gertz 2012b). Dabei handelt es sich um einen regelbasierten, sprachunabhängigen Algorithmus, welcher zur Annotation auf sprachspezifische Ressourcen zurückgreift. Er extrahiert und normalisiert temporale Ausdrücke anhand der TimeML-Richtlinien. Der Tagger berücksichtigt gezielt die Domäne des Textes. Für Text+Berg wählten wir die Domäne *narratives*. Als Input nimmt das System Volltexte. Diese gibt HeidelbergTime als XML-Dateien mit TimeML-Annotationen aus. Nach einer ersten Evaluation des Systems als Tagger für Text+Berg, passten wir HeidelbergTime für das Text+Berg-Korpus an. Der Artikel 3 aus dem Jahrbuch 1900 diente dazu als Entwicklungskorpus. Wir überarbeiteten die Muster- und Normalisierungsdateien der deutschen Ressourcen anhand der im Entwicklungskorpus vorhandenen Zeitausdrücke. Durch diesen Anpassungsschritt konnten wir für das Entwicklungskorpus mit dem Precision-Wert von 95,2 und dem Recall-Wert von 97,5 (partielle Extraktion) deutlich bessere

Ergebnisse erzielen als bei der ersten Evaluation des Taggers. Für das Evaluationskorpus, welches aus 800 manuell annotierten Sätzen der Jahrbücher 1869, 1930, 1960 und 1990 bestand, fielen die Ergebnisse mit Precision-Werten von rund 90,0 und Recall-Werten von etwa 75,0 (partielle Extraktion) etwas tiefer aus. Mit einem Skript zur Verbesserung der Normalisierung konnten wir die Generierung des `value`-Werts im Entwicklungskorpus um ca. 10 Prozentpunkte verbessern, für das Evaluationskorpus zeigten sich hier jedoch nur bedingt Verbesserungen. Generell zeigen die Erhöhungen der Evaluationsergebnisse nach der Anpassung von Heidelberg, wie wichtig die Analyse der korpuspezifischen temporalen Ausdrücke ist. Nur durch das Training des Systems auf den zu annotierenden Korpus kann ein automatisches Annotationssystem gute Ergebnisse erzielen. Auch die Auswahl einer Domäne, die der Textdomäne im Text+Berg-Korpus sehr nahe kam, spielte eine wichtige Rolle. Die Annotationen des angepassten Systems Heidelberg speicherten wir für jedes Jahrbuch in einer Standoff-Datei.

Für die Annotation französischer Zeitausdrücke im Text+Berg-Korpus verwendeten wir in Kapitel 4 das automatische Annotationssystem von Bittar (2009). Dieses wurde für die Annotation von Zeitausdrücken der *news*-Domäne entwickelt und arbeitet mit dem Werkzeug Unitex zur Korpusanalyse. Anders als bei Heidelberg passten wir das französische System nicht an. Als Testkorpus für das Französische benutzten wir 500 manuell annotierte Sätze aus den Jahrbüchern 1872, 1900, 1929, 1960 und 1990. Die Evaluationsergebnisse zeigen für die automatische temporale Annotation des Französischen in Text+Berg deutlich schlechtere Werte als für das Deutsche. Vor allem die sehr tiefen Recall-Werte von 10,0 bis rund 28,0 (partielle Extraktion) replizieren die Beobachtungen, die wir schon während der deutschen Annotation machten: Das Training des Taggers für ein bestimmtes Korpus ist ein wichtiger Teil der automatischen temporalen Annotation. Da das französische Annotationssystem auf eine Domäne trainiert ist, welche sich stark von der Textdomäne des Text+Berg-Korpus unterscheidet und das System keine Anpassung zu korpuspezifischen temporalen Ausdrücken erfuhr, zeigte sich das Annotationssystem von Bittar (2009) für Text+Berg als nur bedingt geeignet.

In Kapitel 5 untersuchten wir die Nutzung der Alignierungen im parallelen deutsch-französischen Teil von Text+Berg. Über die Alignierungsinformationen konnten wir die in den vorhergehenden Kapiteln gemachten temporalen Annotationen vergleichen. Wir untersuchten mit einem Skript, welche deutschen temporalen Ausdrücke im Französischen ebenfalls mit einem annotierten Zeitausdruck aligniert sind. Auf Grund der deutlich besseren Qualität der Annotation im deutschen Teil gingen wir davon aus, dass eine negative Übereinstimmung von temporalen Ausdrücken auf einen falsch annotierten temporalen Ausdruck im Französischen hinweist. In den französischen Jahrbüchern konnten wir auf diese Weise falsche Annotationen elimi-

niert und die Präzision der automatisch generierten Annotationen verbessern.

In einer zweiten Untersuchung beschäftigten wir uns mit der Übertragung von deutschen temporalen Annotationen in das Französische. Auch hier nutzten wir die Alignierungsinformationen. Wir fanden heraus, dass die Projektion bessere Ergebnisse erzielte als die automatische Annotation mit dem unangepassten französischen Tagger von Bittar (2009). Vor allem die Recall-Werte konnten wir mit der Projektionsmethode steigern. In einem kombinierten Verfahren testeten wir anschliessend, ob sich der Vergleich von bestehenden Annotationen und die Projektion der temporalen Annotation so kombinieren lassen, dass für das Französische optimale Ergebnisse erzielt werden. Dieses Verfahren erwies sich als grundsätzlich geeignet, um damit für das Französische die besten Precision- und Recall-Werte aller vorgestellten Annotationsmethoden zu erzielen. Die Evaluationsergebnisse erreichten durch dieses Kombinationsverfahren bessere Werte als der temporale Tagger.

Diese Ergebnisse zeigten uns die Wichtigkeit der Alignierungen in einem parallelen Korpus: Ist eine qualitativ solide automatische temporale Annotation in einer Sprache vorhanden, so erzielt ein Projektionsverfahren bessere Ergebnisse für die Annotation der zweiten Sprache als die Anwendung eines bereits bestehenden, untrainierten Taggers für diese Sprache. Um die durch die Projektion generierten Annotationen noch zu verbessern, kann ein beliebiger temporaler Tagger für diese Sprache angewendet werden und die damit erzeugten Annotationen nach der Verifizierung im Vergleichsverfahren ebenfalls in die Annotationen der Projektion eingebunden werden. Die Ergebnisse sind zwar schlechter als für die Annotation durch einen angepassten Tagger, jedoch deutlich besser als für einen unangepassten Tagger.

Das Ziel der Arbeit, die Bereitstellung eines Moduls mit Informationen über deutsche und französische temporalen Ausdrücken für das Text+Berg-Korpus, konnten wir erreichen: Für das Text+Berg-Korpus stehen nun für jedes Jahrbuch Standoff-Dateien mit Annotationen zu deutschen und französischen Zeitausdrücken bereit. Im Französischen generierten wir die Annotationen für das *Echo des Alpes* und für die Jahrbücher des SAC bis 1956 durch den temporalen Tagger von Bittar (2009). Diese Annotationen erfüllen den Anspruch an hochwertige Qualität nur mässig. Die Annotationen für die französischen Jahrbücher 1957 bis 2011 stammen aus dem kombinierten Verfahren von Projektion und Vergleich und sind von besserer Qualität. Die deutschen Annotationen erzeugten wir mit dem angepassten System HeidelTIme (Strötgen und Gertz 2012b). Wir verbesserten die Disambiguierung anschliessend mit einem eigenen Programm. Dadurch erzeugten wir Annotativen, die wir als qualitativ gute Annotationen beurteilen. Inwiefern die automatisch generierten Annotationen wirklich zufriedenstellend sind, könnten wir durch eine mögliche zukünftige Weiterverwendung der annotierten Zeitausdrücke im Korpus überprüfen.



## 6.2. Grenzen und Probleme

Bei der automatischen temporalen Annotation fanden wir heraus, dass die Normalisierung von ungenügend spezifizierten Zeitausdrücken sehr problematisch ist. Wir vermuten, dass im Text+Berg-Korpus im Vergleich zu anderen Korpora eine hohe Anzahl an ungenügend spezifizierten temporalen Ausdrücken besteht. Die Technik der Nutzung des letzten voll spezifizierten Zeitausdrucks des gleichen Typs als Referenzdatum zur Normalisierung, wie sie von HeidelbergTime angewandt wird, führt im Text+Berg-Korpus nur bedingt zu guten Ergebnissen. Problematisch ist oftmals auch der nicht chronologische Erzählstil in Text+Berg mit den damit einhergehenden Analepsen. Für unsere Anwendungen war es nicht möglich, einen Textabschnitt mit retrospektiven Ereignissen temporal von der aktuellen Chronik der Erzählung abzugrenzen. Dadurch ermittelten wir bei der Normalisierung immer wieder falsche Referenzdaten. Nur die Erwähnung von expliziten Datumsangaben kann dem System wieder eine korrekte Referenz geben. Fehlen explizite Zeitausdrücke überdurchschnittlich häufig, fallen auch die Werte für `value` schlecht aus.

Ein weiteres Problem bei der Extraktion von temporalen Ausdrücken ist die Annotation von generischen temporalen Ausdrücken. Ist in einem Text allgemein die Rede vom Morgen als Tageszeit oder vom Sommer als Jahreszeit und stehen diese Ausdrücke nicht in Relation zur momentanen Erzählchronik, so sollten sie nicht als temporale Ausdrücke markiert werden. Wir sahen im Rahmen dieser Arbeit keine Möglichkeit, dies in die automatischen Annotationssysteme mit einzubinden.

Ein korpuspezifisches Problem bei der Extraktion von Zeitausdrücken stellen auch die OCR-Fehler dar. Dies liegt, den Beobachtungen beim Training des deutschen Taggers zufolge, vor allem daran, dass Ziffern für das OCR-System häufig Probleme bereiteten. So wurde die Ziffer 0 gelegentlich als Buchstabe O erkannt oder Bruchzahlen in Uhrzeitangaben nicht korrekt dargestellt. Da Zeitausdrücke oft Ziffern enthalten, sind sie generell anfällig für OCR-Fehler und können in der Folge von einem temporalen Tagger nicht erkannt werden.

Zusätzlich wollen wir an dieser Stelle erwähnen, dass die Anpassung des Taggers HeidelbergTime vereinfacht werden könnte, würde das Programm mit Lemmata arbeiten. In den Musterdateien müssten dann nicht verschiedene reguläre Ausdrücke für einen Zeitausdruck aufgelistet werden, sondern es könnte lediglich auf das Lemma verwiesen werden. So könnten wir zusätzliche Sprachressourcen und Ressourcen für neue Sprachen schneller integrieren.

### 6.3. Offene Punkte und Ausblick

Als mögliche weitere Untersuchungen auf dem Gebiet der temporalen Annotation im Text+Berg-Korpus sehen wir in erster Linie die Verbesserungen der Probleme aus Kapitel 6.2. Durch eine genaue Korpusanalyse könnten wir die für Analepsen typischen Muster herausarbeiten. Diese Muster könnten wir im Skript zur Verbesserung der Normalisierungen einbinden, um retrospektive Zeitausdrücke zu kennzeichnen und sie anschliessend als mögliches Referenzdatum für ungenügend spezifizierte temporale Ausdrücke zu sperren. Um die Annotation von generischen Zeitausdrücken zu vermeiden, könnten wir für das Deutsche die Ressourcen entsprechend modifizieren. Wir müssten testen, ob dies generell zu einer verbesserten Präzision führt oder ob der Recall-Wert damit sinkt. Auch die Berücksichtigung von OCR-Fehlern könnten wir in die Ressourcen des Taggers mit einbinden.

Wir vermuten, dass wir mit der Generierung von französischen Ressourcen für HeidelbergTime die Annotation von Zeitausdrücken in dieser Sprache verbessern könnten. Dazu müssten wir lexikalische Ressourcen für französische Zeitausdrücke zusammenstellen sowie die französischen Texte im Korpus intensiv auf spezifische temporale Ausdrücke hin untersuchen.

Um die temporale Annotation im Text+Berg-Korpus zu komplettieren, müssten wir auch die Texte in italienischer, englischer, schweizerdeutscher und rätoromanischer Sprache annotieren. Gerade bei Sprachen mit wenig computerlinguistischen Ressourcen, wie dem schweizerdeutschen und dem rätoromanischen, steht der Aufwand zur Generierung eines Taggers jedoch nicht in Relation zur Textmenge der betreffenden Sprachen im Korpus. Für englische und italienische Texte könnten wir bestehende Systeme für die jeweiligen Sprachen testen, analog zur Untersuchung für das Deutsche und Französische in dieser Arbeit.

Durch die Weiterverwendung der extrahierten Zeitausdrücke in computerlinguistischen Anwendungen oder Systemen könnten wir in der Zukunft testen, ob die Qualität der Annotationen ausreichend ist. Wenn die Interpretationen der gefundenen temporalen Ausdrücke in zu vielen Fällen unzuverlässig ist, sollten wir die Annotationen noch weiter optimieren. Um die Annotationen von TimeML in vollem Umfang nutzen zu können, müssen wir auch Aktionen und Relationen zwischen Zeitausdrücken und Aktionen im Text markieren. Diese Annotationen stützen sich auf die extrahierten Zeitausdrücke und wären ein möglicher nächster Schritt in der Aufbereitung des Text+Berg-Korpus.

# Bibliographie

- A. Bittar. Annotation des Informations Temporelles dans des Textes en Français. In *RECITAL 2008*, 2008.
- A. Bittar. Annotation of Events and Temporal Expressions in French Texts. In *Linguistic Annotation Workshop*, pages 48–51, 2009.
- A. Bittar. *Building a TimeBank for French: A Reference Corpus Annotated According to the ISO-TimeML Standard*. PhD thesis, Université Paris Diderot (Paris 7), 2010.
- N. Bubenhofer, M. Volk, A. Althaus, M. Jitca, D. Klaper, und R. Sennrich. Text+Berg-Korpus (Release 147 v02). XML-Format, 2012. Digitale Edition des Jahrbuch des SAC 1864-1923, Echo des Alpes 1872-1924 und Die Alpen 1925-2011.
- M. Burghardt und C. Wolff. Stand off-Annotation für Textdokumente: Vom Konzept zur Implementierung (zur Standardisierung?). In *From Form to Meaning: Processing Texts Automatically: Proceedings of the Biennial GSCL Conference 2009*, pages 53–59. 2009.
- F. Costa und A. Branco. Temporal Information Processing of a New Language: Fast Porting with Minimal Resources. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 671–677, 2010.
- M. Ehrmann und C. Hagège. Proposition de Caractérisation et de Typage des Expressions Temporelles en Contexte. *Actes de TALN*, 2009.
- U. Endriss, U. Küssner, und M. Stede. Repräsentation Zeitlicher Ausdrücke: Die Temporal Expression Language. Verbmobil Memo 133, Technical University Berlin, Department of Computer Science, 1998.
- L. Ferro, L. Gerber, I. Mani, B. Sundheim, und G. Wilson. TIDES 2005 Standard for the Annotation of Temporal Expressions. *The MITRE Corporation*, 2005.

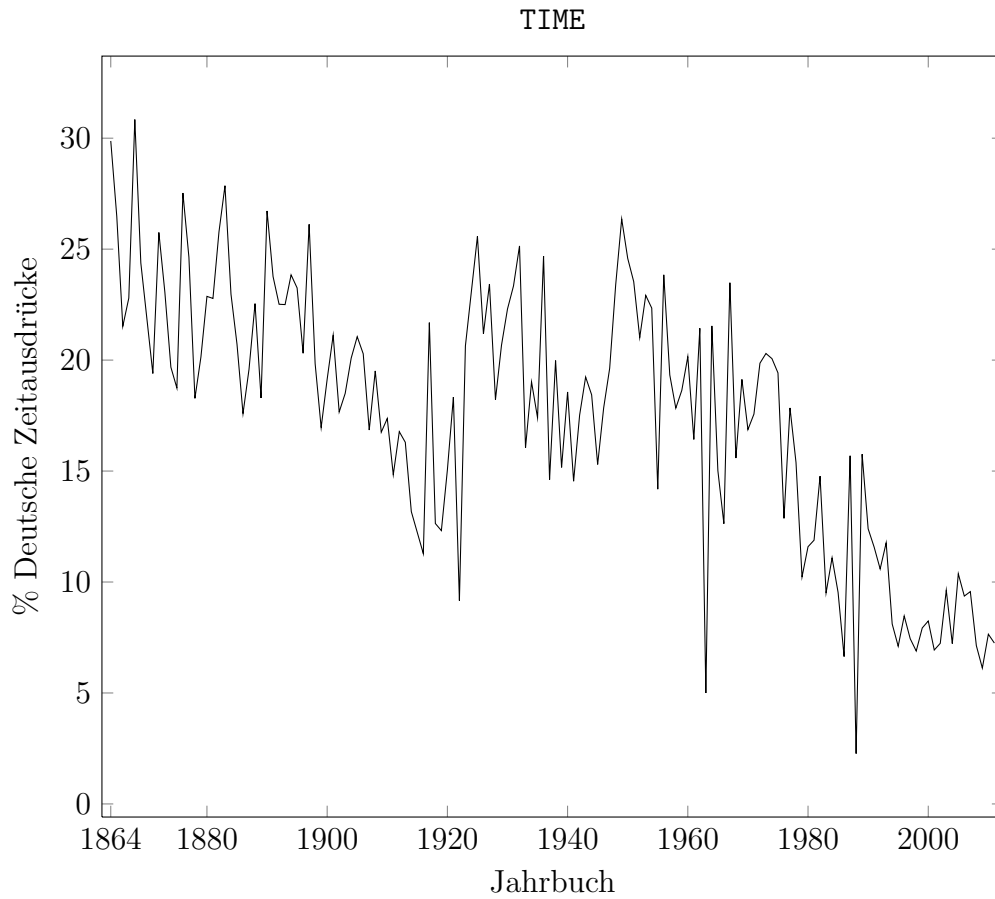
- W. A. Gale und K. W. Church. A Program for Aligning Sentences in Bilingual Corpora. In *Proceedings of the 29th Annual Meeting on Association for Computational Linguistics*, pages 177–184, 1991.
- A. Göhring und M. Volk. The Text+Berg Corpus: An Alpine French-German Parallel Resource. In *TALN 2011*, 2011.
- M. Gross. Les Déterminants Numéraux, un Exemple: les Dates Horaires. *Langages*, 145:21–37, 2002.
- I. Gurevych, M. Mühlhäuser, C. Müller, J. Steimle, M. Weimer, und T. Zesch. Darmstadt Knowledge Processing Repository Based on UIMA. In *Proceedings of the First Workshop on Unstructured Information Management Architecture at Biannual Conference of the Society for Computational Linguistics and Language Technology*, 2007.
- P. Koehn. *Statistical Machine Translation*. Cambridge University Press, 2010.
- I. Mani, J. Pustejovsky, und R. Gaizauskas. *The Language Of Time: A Reader*. Oxford University Press, 2005.
- P. Muller und X. Tannier. Annotating and Measuring Temporal Relations in Texts. In *Proceedings of Coling 2004*, pages 50–56, 2004.
- G. Parent, M. Gagnon, und P. Muller. Annotation d’Expressions Temporelles et d’Événements en Français. In *Traitement Automatique des Langues Naturelles (TALN), Avignon, 2008*, 2008.
- M. Piotrowski, S. Läubli, und M. Volk. Towards Mapping of Alpine Route Descriptions. In *Proceedings of the 6th Workshop on Geographic Information Retrieval (GIR’10)*, pages 15–16, 2010.
- J. Pustejovsky, R. Knippen, J. Littman, und R. Sauri. Temporal and Event Information in Natural Language Text. *Language Resources and Evaluation*, 39: 123–164, 2005.
- F. Schilder und C. Habel. From Temporal Expressions to Temporal Information: Semantic Tagging of News Messages. In *Proceedings of the Workshop on Temporal and Spatial Information Processing - Volume 13*, pages 9:1–9:8, 2001.
- F. Schilder, G. Katz, und J. Pustejovsky. Annotating, Extracting and Reasoning About Time and Events. In *Proceedings of the 2005 International Conference on Annotating, Extracting and Reasoning About Time and Events*, pages 1–6, 2007.

- R. Sennrich und M. Volk. MT-based Sentence Alignment for OCR-generated Parallel Texts. In *The Ninth Conference of the Association for Machine Translation in the Americas (AMTA 2010)*, 2010.
- R. Sennrich und M. Volk. Iterative, MT-based Sentence Alignment of Parallel Texts. In *NODALIDA 2011, Nordic Conference of Computational Linguistics*, 2011.
- K. Spreyer und A. Frank. Projection-based Acquisition of a Temporal Labeller. In *Proceedings of IJCNLP*, pages 489–496, 2008.
- M. Stede. *Korpusgestützte Textanalyse: Grundzüge der Ebenen-Orientierten Textlinguistik*. Narr Studienbücher. Gunter Narr Dr., 2007.
- J. Strötgen und M. Gertz. HeidelTime: High Quality Rule-Based Extraction and Normalization of Temporal Expressions. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 321–324, 2010.
- J. Strötgen und M. Gertz. WikiWarsDE: A German Corpus of Narratives Annotated with Temporal Expressions. In *Proceedings of the Conference of the German Society for Computational Linguistics and Language Technology (GSCL 2011)*, pages 129–134, 2011.
- J. Strötgen und M. Gertz. Temporal Tagging on Different Domains: Challenges, Strategies, and Gold Standards. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 3746–3753, 2012a.
- J. Strötgen und M. Gertz. Multilingual and Cross-domain Temporal Tagging. *Language Resources and Evaluation*, 2012b.
- N. Vazov. A System for Extraction of Temporal Expressions from French Texts Based on Syntactic and Semantic Constraints. In *Proceedings of ACL-2001: Workshop on Temporal and Spatial Information Processing*, 2001.
- M. Verhagen, R. Saurí, T. Caselli, und J. Pustejovsky. SemEval-2010 task 13: TempEval-2. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 57–62, 2010.

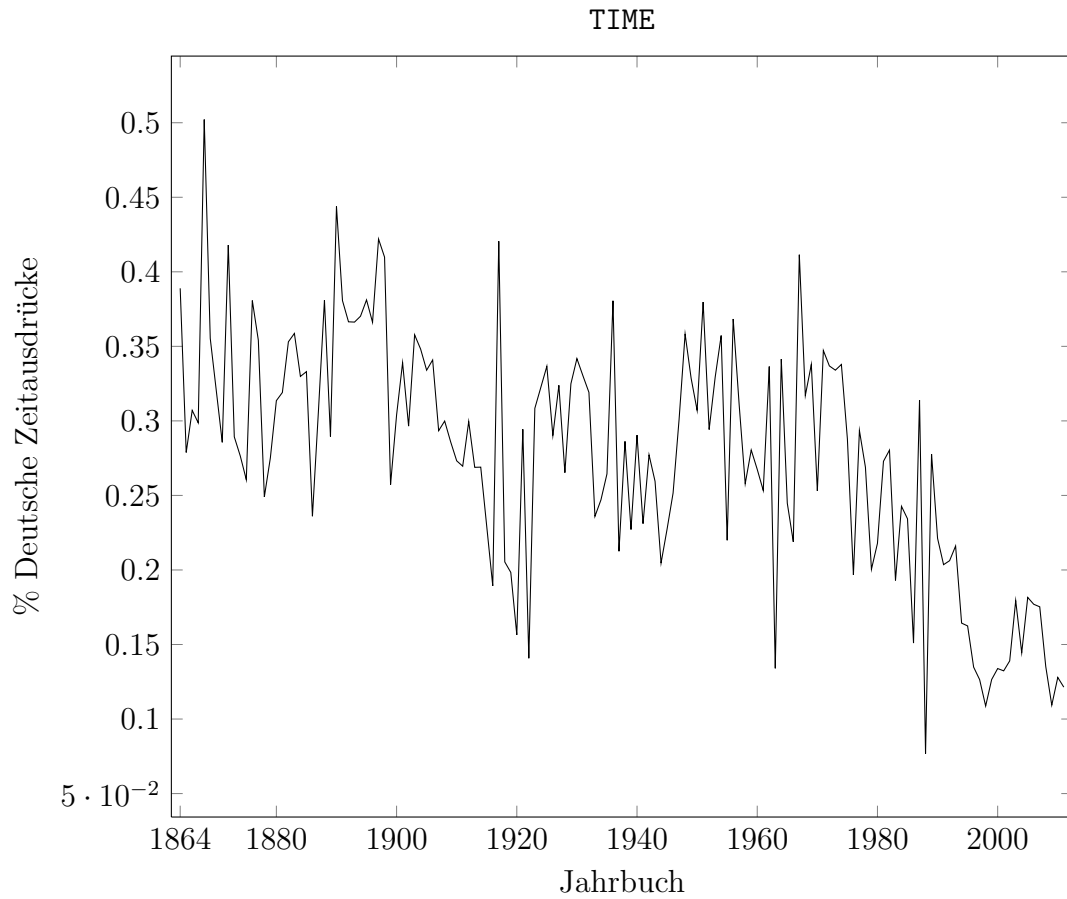
# A. Diagramme

Die folgenden Diagramme präsentieren die deutschen temporalen Annotationen in Text+Berg, generiert durch das angepasste Annotationssystem HeideTime. Für jeden Typ (`TIME`, `DATE`, `DURATION` und `SET`) wurden drei Diagramme erstellt, welche die folgenden Informationen darstellen:

- Verhältnis der temporalen Annotationen des Typs zu allen `TIMEX3`-Annotationen
- Verhältnis der temporalen Annotationen des Typs zu allen Token im betreffenden Jahrbuch
- Verhältnis aller annotierten Token des Typs zu allen Token im betreffenden Jahrbuch

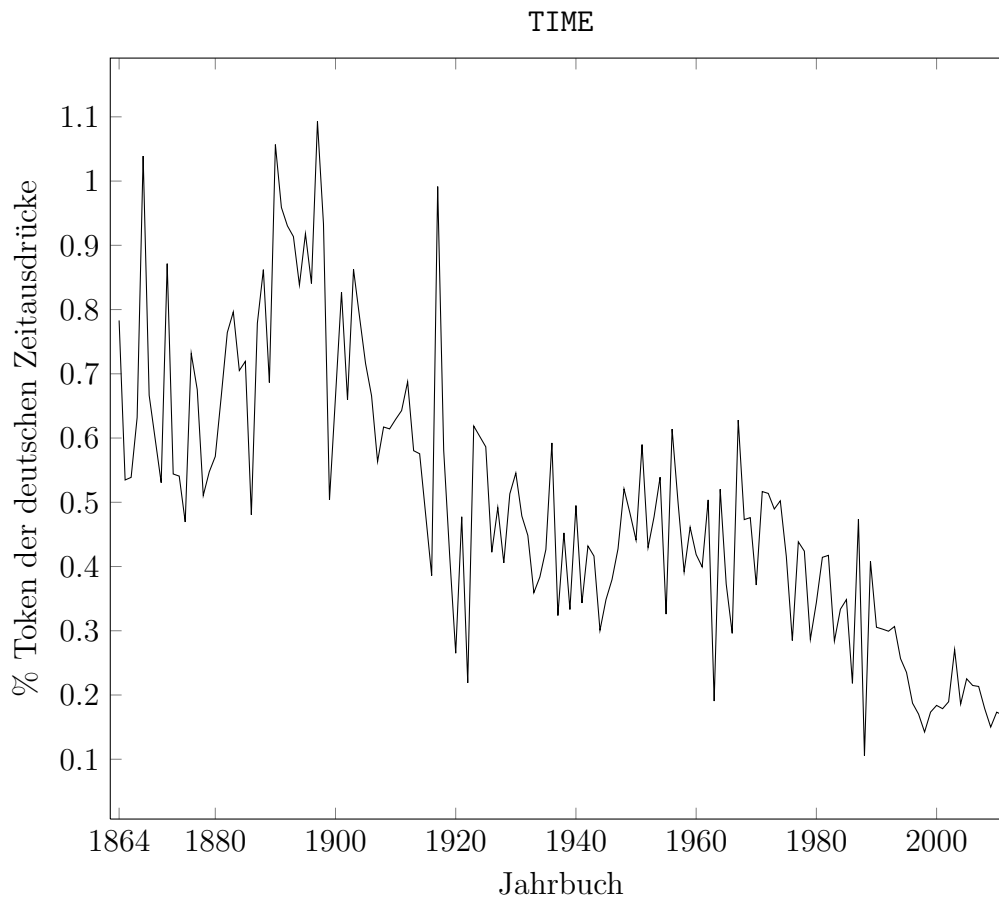


Prozentualer Anteil deutscher temporaler Ausdrücke des Typs TIME an allen gefundenen Zeitausdrücken im jeweiligen Jahrbuch.

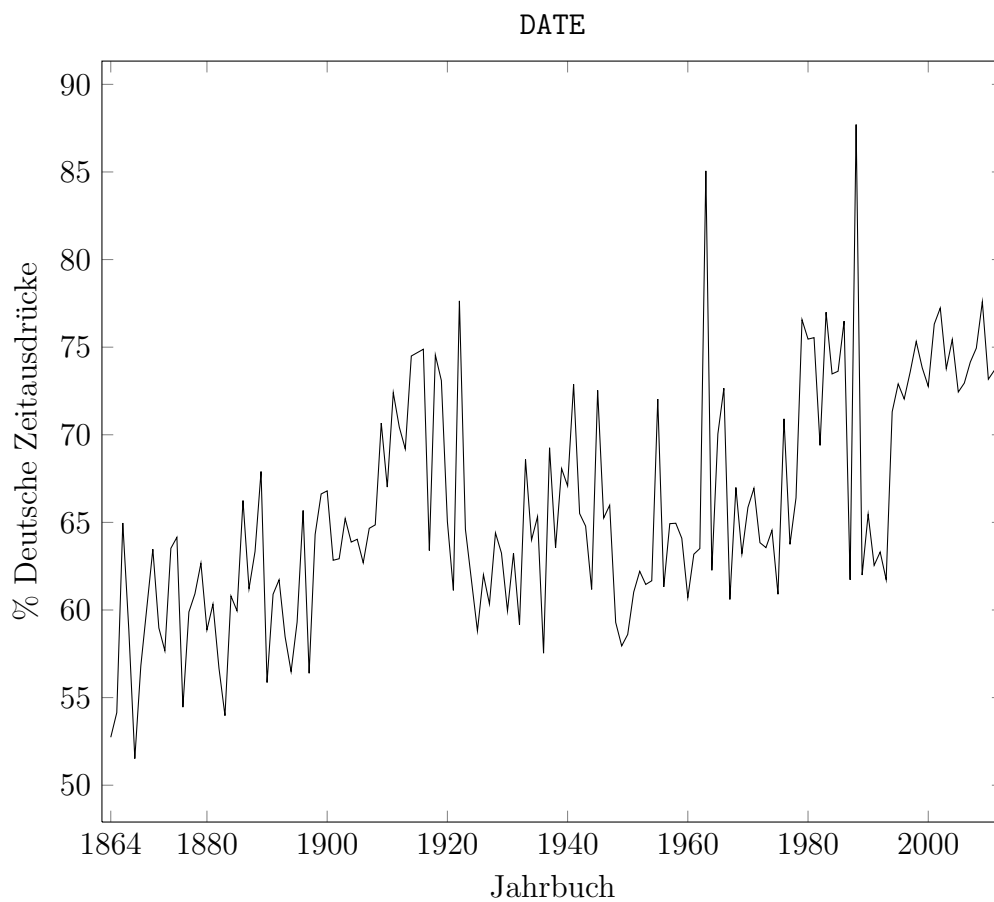


Prozentuales Verhältnis deutscher temporaler Ausdrücke des Typs TIME und aller Token im betreffenden Jahrbuch.

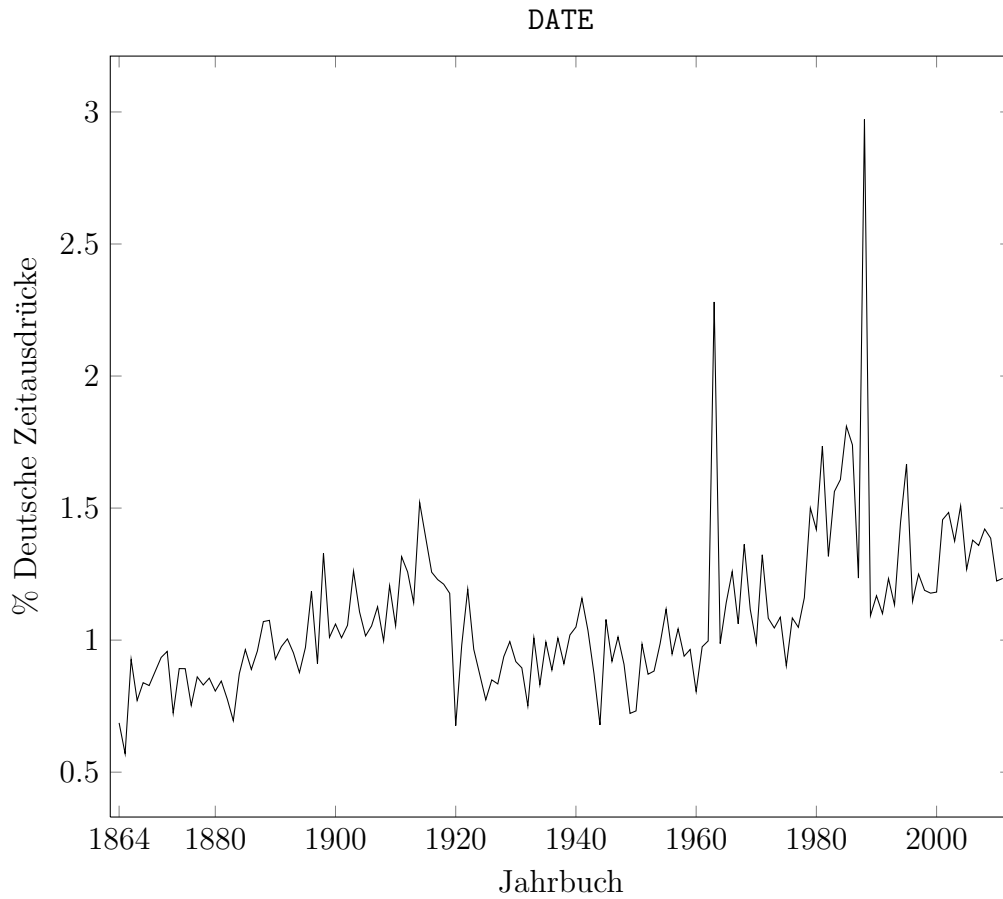




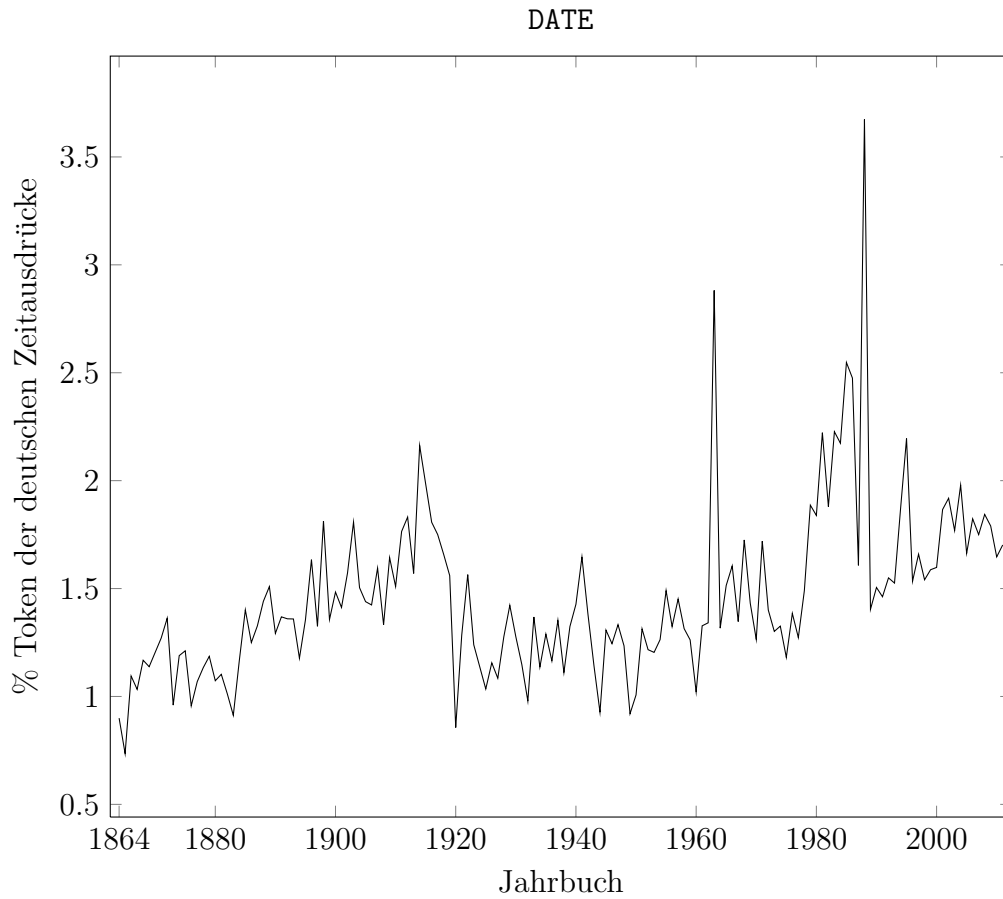
Prozentuales Verhältnis aller Token, welche als deutsche temporale Ausdrücke des Typs TIME annotiert wurden und aller Token im betreffenden Jahrbuch.



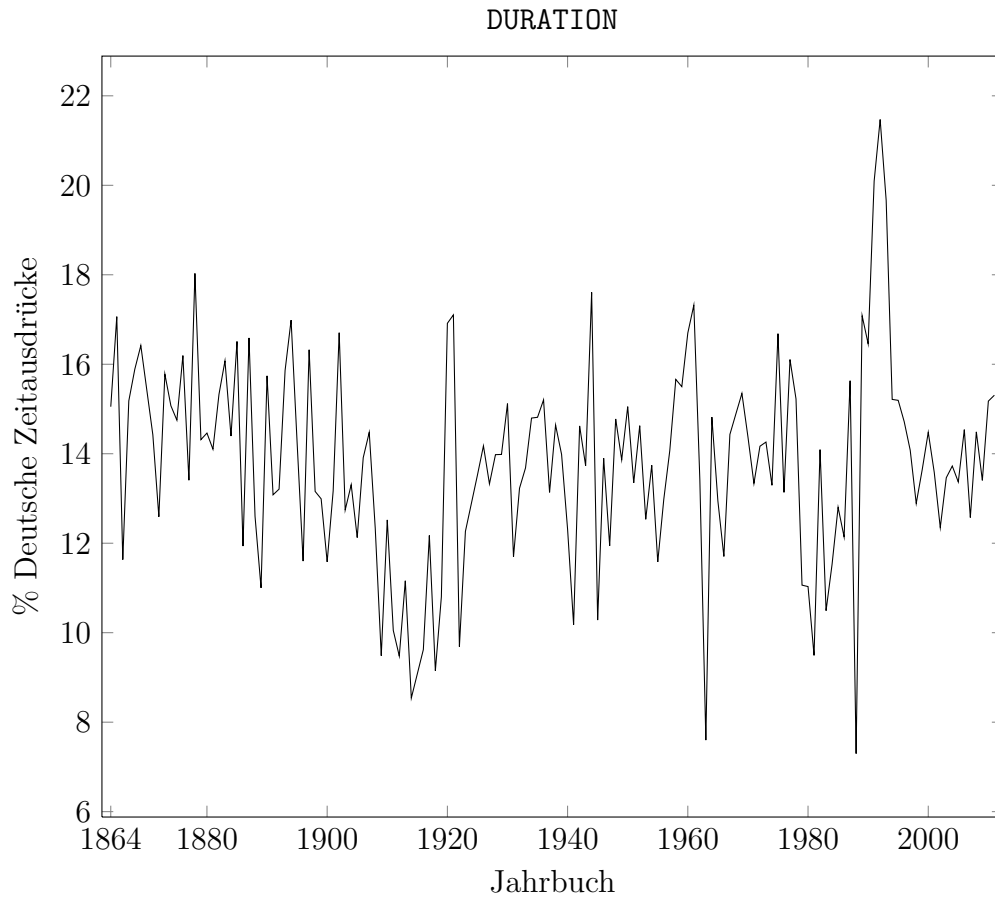
Prozentualer Anteil deutscher temporaler Ausdrücke des Typs DATE an allen gefundenen Zeitausdrücken im jeweiligen Jahrbuch.



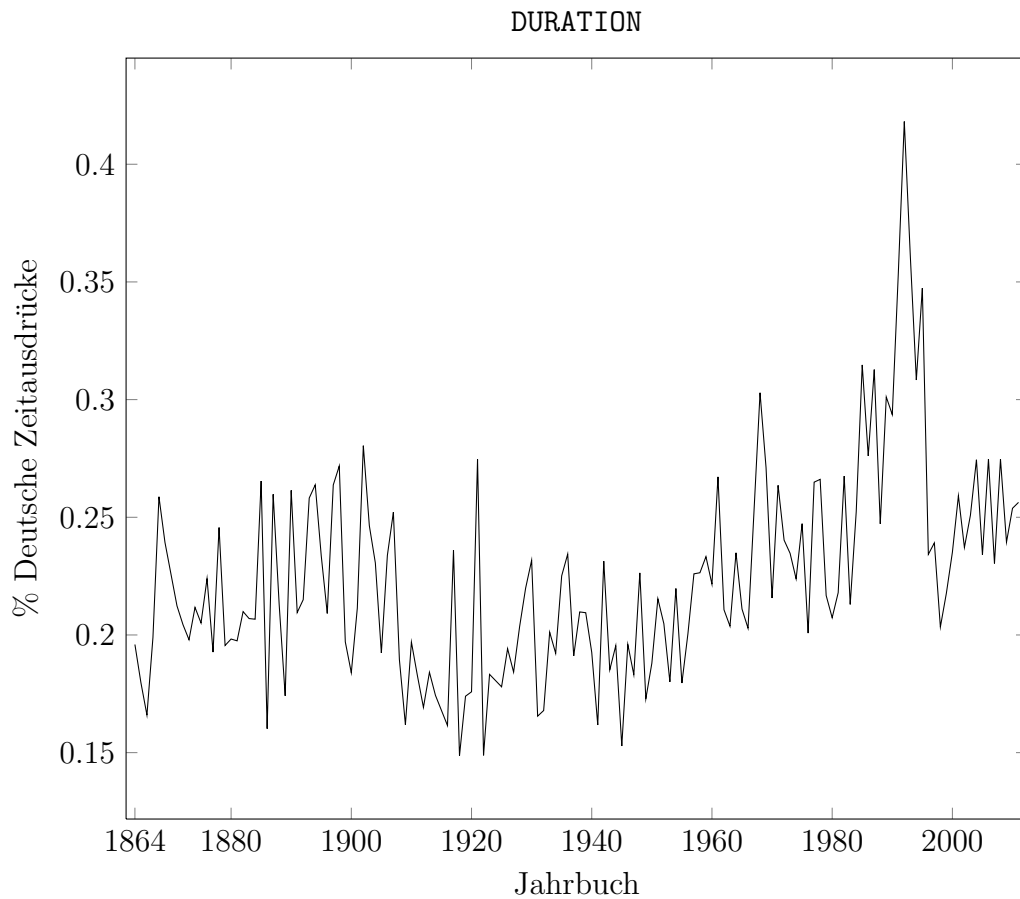
Prozentuales Verhältnis deutscher temporaler Ausdrücke des Typs DATE und aller Token im betreffenden Jahrbuch.



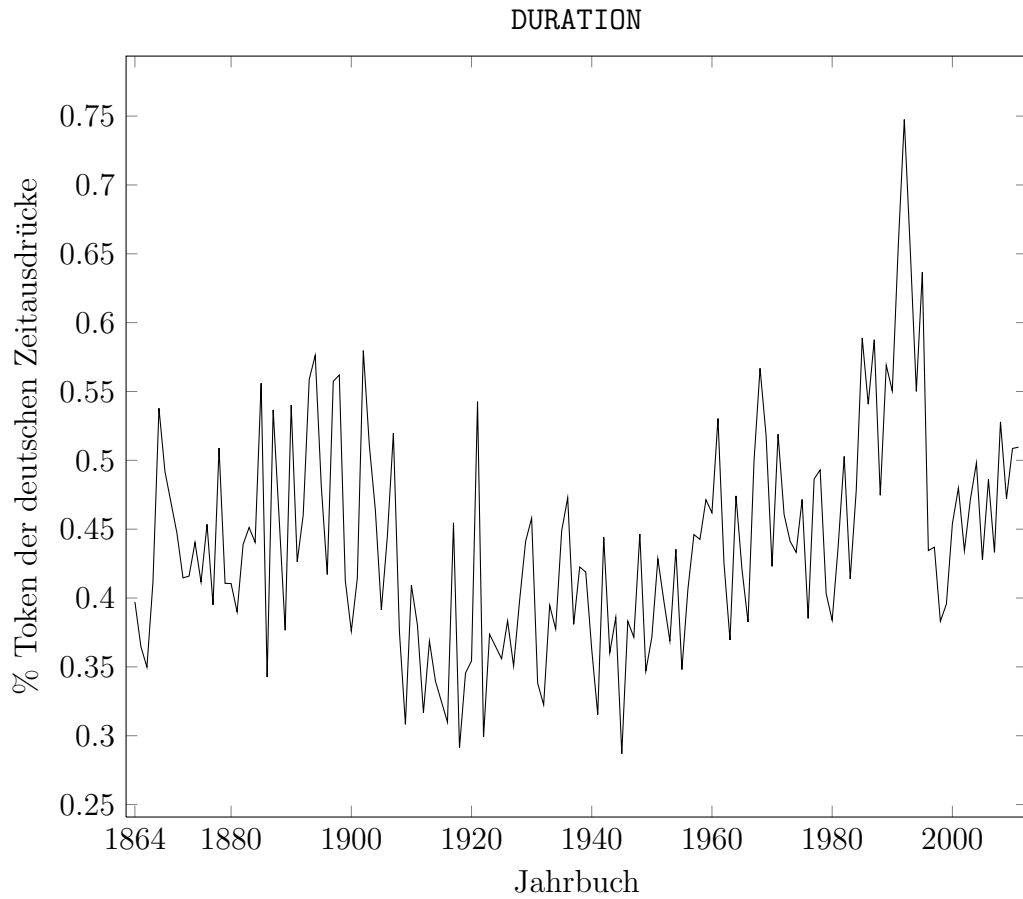
Prozentuales Verhältnis aller Token, welche als deutsche temporale Ausdrücke des Typs DATE annotiert wurden und aller Token im betreffenden Jahrbuch.



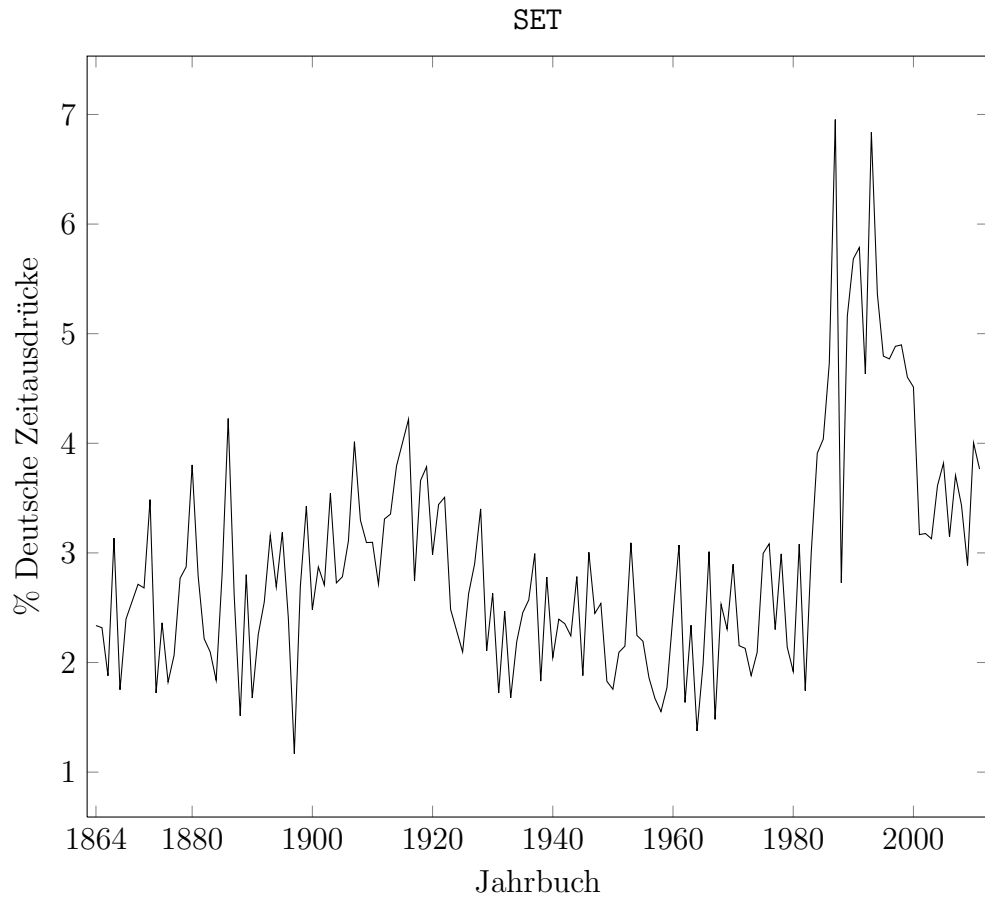
Prozentualer Anteil deutscher temporaler Ausdrücke des Typs DURATION an allen gefundenen Zeitausdrücken im jeweiligen Jahrbuch.



Prozentuales Verhältnis deutscher temporalen Ausdrücke des Typs DURATION und aller Token im betreffenden Jahrbuch.

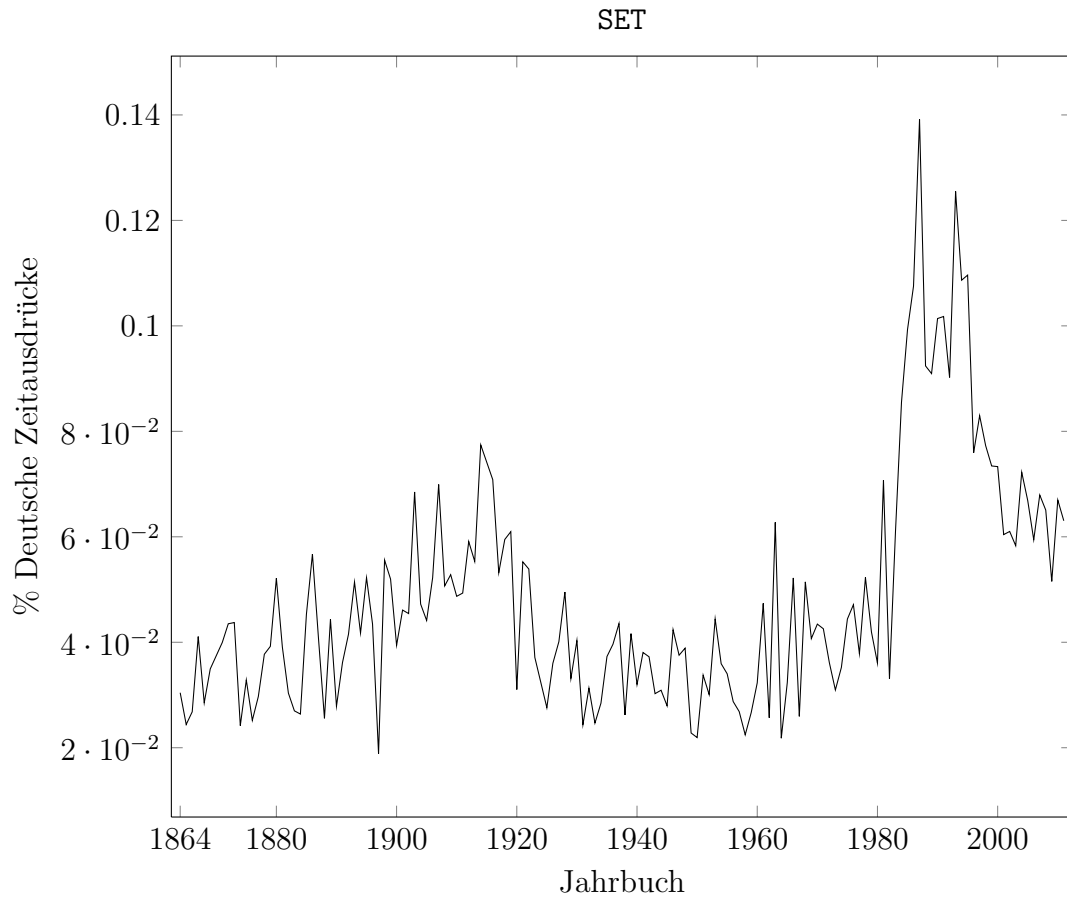


Prozentuales Verhältnis aller Token, welche als deutsche temporale Ausdrücke des Typs DURATION annotiert wurden und aller Token im betreffenden Jahrbuch.

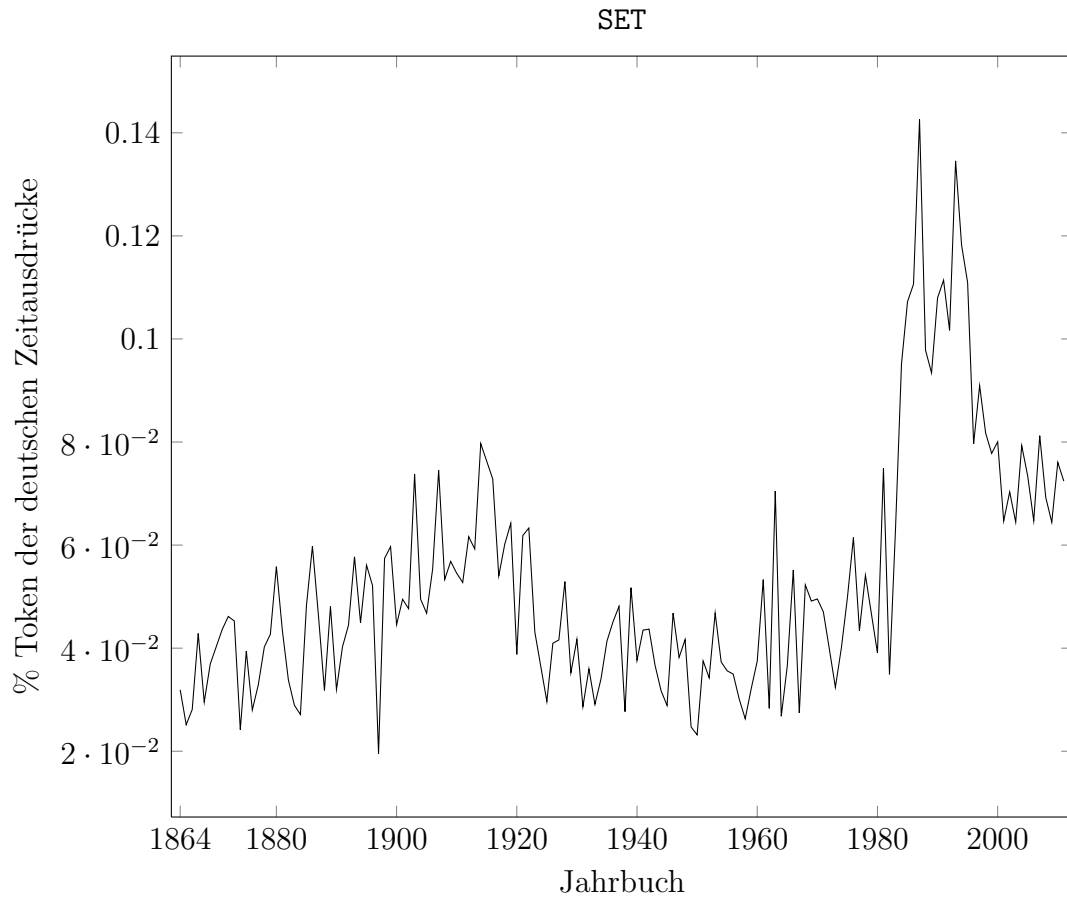


Prozentualer Anteil deutscher temporalen Ausdrücke des Typs SET an allen gefundenen Zeitausdrücken im jeweiligen Jahrbuch.





Prozentuales Verhältnis deutscher temporalen Ausdrücke des Typs SET und aller Token im betreffenden Jahrbuch.



Prozentuales Verhältnis aller Token, welche als deutsche temporale Ausdrücke des Typs SET annotiert wurden und aller Token im betreffenden Jahrbuch.

## B. Zusätzliche Quellenangaben

Die deutschen Ressourcen des temporalen Annotationssystems HeidelTime wurden für das Text+Berg-Korpus angepasst. Die hinzugefügten Datumsangaben für Feiertage in der Schweiz stammen aus den folgenden Quellen. Stand: 06.03.2013.

- <http://www.datumsrechner.de/FeiertageSchweiz.pdf>
- <http://www.feiertagskalender.ch/a-z.php?geo=0>
- [http://www.ejpd.admin.ch/content/dam/data/staat\\_buerger/zivilprozessrecht/kant-feiertage.pdf](http://www.ejpd.admin.ch/content/dam/data/staat_buerger/zivilprozessrecht/kant-feiertage.pdf)
- [http://de.wikipedia.org/wiki/Christi\\_Himmelfahrt](http://de.wikipedia.org/wiki/Christi_Himmelfahrt)
- <http://www.feiertage-schweiz.ch/fasnachtsmontag.php>
- <http://www.feiertage-schweiz.ch/nafelserfahrt.php>
- <http://de.wikipedia.org/wiki/Sechsel%3%A4uten#Datum>
- <http://de.wikipedia.org/wiki/Knabenschiessen>
- <http://de.wikipedia.org/wiki/Banntag>
- [http://de.wikipedia.org/wiki/Eidgen%3%B6ssischer\\_Dank-,\\_Buss-\\_und\\_Betttag](http://de.wikipedia.org/wiki/Eidgen%3%B6ssischer_Dank-,_Buss-_und_Betttag)
- <http://de.wikipedia.org/wiki/Sommerzeit>
- <http://de.wikipedia.org/wiki/Solennit%3%A4t>
- <http://de.wikipedia.org/wiki/Rutenzug>
- <http://de.wikipedia.org/wiki/Maienzug>
- <http://de.wikipedia.org/wiki/Ausschiesset>
- <http://de.wikipedia.org/wiki/Reformationstag>
- <http://de.wikipedia.org/wiki/Ewigkeitssonntag>
- <http://de.wikipedia.org/wiki/Zibelem%3%A4rit>

## C. Korpusauszug

Das Beispiel 3.3 in Kapitel 3.3 zeigt eine Textpassage aus dem Artikel 3 im Jahrbuch 1900. Der folgende Auszug aus der Korpusdatei des Jahrbuchs 1900 zeigt die betreffende Textpassage.

```
<s lang="de" n="3-86">
 <w lemma="d" n="3-86-1" pos="ART">Die</w>
 <w lemma="de" n="3-86-2" pos="NE">de</w>
 <w lemma="unk" n="3-86-3" pos="NE">d'</w>
 <w lemma="unk" n="3-86-4" pos="NE">Orny</w>
 <w lemma="," n="3-86-5" pos=",$,>,</w>
 <w lemma="im" n="3-86-6" pos="APPRART">im</w>
 <w lemma="Jahr" n="3-86-7" pos="NN">Jahr</w>
 <w lemma="@card@" n="3-86-8" pos="CARD">1893</w>
 <w lemma="neu" n="3-86-9" pos="ADJD">neu</w>
 <w lemma="erstellen" n="3-86-10" pos="VVPP">erstellt</w>
 <w lemma="von" n="3-86-11" pos="APPR">von</w>
 <w lemma="d" n="3-86-12" pos="ART">der</w>
 <w lemma="Sektion" n="3-86-13" pos="NN">Section</w>
 <w lemma="d" n="3-86-14" pos="ART">des</w>
 <w lemma="unk" n="3-86-15" pos="NN">Diablerets</w>
 <w lemma="," n="3-86-16" pos=",$,>,</w>
 <w lemma="unter" n="3-86-17" pos="APPR">unter</w>
 <w lemma="d" n="3-86-18" pos="PRELAT">deren</w>
 <w lemma="Obhut" n="3-86-19" pos="NN">Obhut</w>
 <w lemma="sie" n="3-86-20" pos="PPER">sie</w>
 <w lemma="stehen" n="3-86-21" pos="VVFIN">steht</w>
 <w lemma="," n="3-86-22" pos=",$,>,</w>
 <w lemma="sein" n="3-86-23" pos="VAFIN">ist</w>
 <w lemma="ein" n="3-86-24" pos="ART">ein</w>
 <w lemma="behaglich" n="3-86-25" pos="ADJA">behagliches</w>
 <w lemma="Heim" n="3-86-26" pos="NN">Heim</w>
```

<w lemma="." n="3-86-27" pos="\$.">.</w>  
</s>  
<s lang="de" n="3-87">  
<w lemma="nur" n="3-87-1" pos="ADV">Nur</w>  
<w lemma="erleben" n="3-87-2" pos="VVFIN">erlebten</w>  
<w lemma="wir" n="3-87-3" pos="PPER">wir</w>  
<w lemma="gleich" n="3-87-4" pos="ADJD">gleich</w>  
<w lemma="bei" n="3-87-5" pos="APPR">bei</w>  
<w lemma="d" n="3-87-6" pos="ART">der</w>  
<w lemma="Ankunft" n="3-87-7" pos="NN">Ankunft</w>  
<w lemma="ein" n="3-87-8" pos="ART">eine</w>  
<w lemma="herb" n="3-87-9" pos="ADJA">herbe</w>  
<w lemma="Enttäuschung" n="3-87-10" pos="NN">Enttäuschung</w>  
<w lemma="." n="3-87-11" pos="\$.">.</w>  
</s>  
<s lang="de" n="3-88">  
<w lemma="nach" n="3-88-1" pos="APPR">Nach</w>  
<w lemma="d" n="3-88-2" pos="ART">dem</w>  
<w lemma="Reglement" n="3-88-3" pos="NN">Reglement</w>  
<w lemma="sollen" n="3-88-4" pos="VMFIN">sollte</w>  
<w lemma="d" n="3-88-5" pos="ART">die</w>  
<w lemma="Hütte" n="3-88-6" pos="NN">Hütte</w>  
<w lemma="mit" n="3-88-7" pos="APPR">mit</w>  
<w lemma="Holz" n="3-88-8" pos="NN">Holz</w>  
<w lemma="versorgen" n="3-88-9" pos="VVPP">versorgt</w>  
<w lemma="sein" n="3-88-10" pos="VAINF">sein</w>  
<w lemma="." n="3-88-11" pos="\$.">.</w>  
</s>  
<s lang="de" n="3-89">  
<w lemma="aber" n="3-89-1" pos="KON">Aber</w>  
<w lemma="wie" n="3-89-2" pos="KOUS">wie</w>  
<w lemma="so" n="3-89-3" pos="ADV">so</w>  
<w lemma="oft" n="3-89-4" pos="ADV">oft</w>  
<w lemma="im" n="3-89-5" pos="APPRART">im</w>  
<w lemma="Leben" n="3-89-6" pos="NN">Leben</w>  
<w lemma="stimmen" n="3-89-7" pos="VVFIN">stimmte</w>  
<w lemma="auch" n="3-89-8" pos="ADV">auch</w>  
<w lemma="hier" n="3-89-9" pos="ADV">hier</w>  
<w lemma="d" n="3-89-10" pos="ART">das</w>

<w lemma="Reglement" n="3-89-11" pos="NN">Reglement</w>  
<w lemma="mit" n="3-89-12" pos="APPR">mit</w>  
<w lemma="d" n="3-89-13" pos="ART">der</w>  
<w lemma="Tatsache" n="3-89-14" pos="NN">Thatsache</w>  
<w lemma="nicht" n="3-89-15" pos="PTKNEG">nicht</w>  
<w lemma="." n="3-89-16" pos="\$. ">.</w>  
</s>  
<s lang="de" n="3-90">  
<w lemma="vom" n="3-90-1" pos="APPRART">Vom</w>  
<w lemma="letzt" n="3-90-2" pos="ADJA">letzten</w>  
<w lemma="Jahr" n="3-90-3" pos="NN">Jahr</w>  
<w lemma="sein" n="3-90-4" pos="VAFIN">war</w>  
<w lemma="nichts" n="3-90-5" pos="PIS">nichts</w>  
<w lemma="mehr" n="3-90-6" pos="ADV">mehr</w>  
<w lemma="übrig" n="3-90-7" pos="ADJD">übrig</w>  
<w lemma="bleiben" n="3-90-8" pos="VVPP">geblieben</w>  
<w lemma="und" n="3-90-9" pos="KON">und</w>  
<w lemma="dies" n="3-90-10" pos="PDAT">dies</w>  
<w lemma="Jahr" n="3-90-11" pos="NN">Jahr</w>  
<w lemma="noch" n="3-90-12" pos="ADV">noch</w>  
<w lemma="nichts" n="3-90-13" pos="PIS">nichts</w>  
<w lemma="heraufschaffen" n="3-90-14" pos="VVPP">heraufgeschafft</w>  
<w lemma="werden" n="3-90-15" pos="VAPP">worden</w>  
<w lemma="." n="3-90-16" pos="\$. ">.</w>  
</s>



## **Selbstständigkeitserklärung**

Hiermit erkläre ich, dass die Masterarbeit von mir selbst und ohne unerlaubte Beihilfe verfasst worden ist und ich die Grundsätze wissenschaftlicher Redlichkeit einhalte (vgl. dazu: <http://www.lehre.uzh.ch/index/LK-Plagiate-Merkblatt.pdf>).

Bern, 01.07.2013

Katrin Rettich

.....  
Ort und Datum

.....  
Unterschrift