FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

GOALS

Using TimeML Annotations for an Information Extraction Approach to Support the Modeling of Clinical Guidelines

DISSERTATION

zur Erlangung des akademischen Grades

Doktor der Technischen Wissenschaften

eingereicht von

Dipl.-Ing., Dipl.-Wirtsch.-Ing.(FH) Reinhardt Wenzina
Matrikelnummer 8225629

an der Fakultät für Informatik
der Technischen Universität Wien

Betreuung:
Univ.Prof.in Dr.in Silvia Miksch
Mag.a Dr.in Katharina Kaiser

Diese Dissertation haben begutachtet:

Univ.Prof.in Dr.in
Silvia Miksch

Mor Peleg

Assoc. Prof.
Mor Peleg, PhD

Wien, 15. April 2015

Reinhardt Wenzina

FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

GOALS

Using TimeML Annotations for an Information Extraction Approach to Support the Modeling of Clinical Guidelines

DISSERTATION

submitted in partial fulfillment of the requirements for the degree of

Doktor der Technischen Wissenschaften

by

Dipl.-Ing., Dipl.-Wirtsch.-Ing.(FH) Reinhardt Wenzina
Registration Number 8225629

to the Faculty of Informatics
at the Vienna University of Technology

Advisors:
Univ.Prof.in Dr.in Silvia Miksch
Mag.a Dr.in Katharina Kaiser

The dissertation has been reviewed by:

Univ.Prof.in Dr.in Silvia Miksch	Mor Peleg Assoc. Prof. Mor Peleg, PhD

Vienna, 15th April 2015

Reinhardt Wenzina

Erklärung zur Verfassung der Arbeit

Dipl.-Ing., Dipl.-Wirtsch.-Ing.(FH) Reinhardt Wenzina
Wilhelm Mantlergasse 608
A-3571 Gars am Kamp

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 15. April 2015

Reinhardt Wenzina

Acknowledgements

First of all I would like to thank my supervisor Silvia Miksch, who gave me the opportunity to do my PhD studies at the Vienna University of Technology (Institute of Software Technology and Interactive Systems). Furthermore, I am highly grateful to Katharina Kaiser who supported me from the very beginning with her experience and scientific knowledge. She has managed to guide me through the ups and downs of scientific life, relentlessly encouraging me to perform to the best. I would also like to thank Mor Peleg who immediately expressed her willingness to review my thesis.

Special thanks go to my family – to my daughter Judith who motivated me with her youthful energy to start such a project beneath my full-time job as a teacher, and to my wife Brigitte for her patience and academic advice. Without their help I would not have mastered this challenge. Last, but not least, I thank my parents, Helga and Anton Wenzina, who – from my childhood onwards – taught me to finish things once begun.

Thanks to the Institute of Software Technology and Interactive Systems (Vienna University of Technology, Austria), and to the FWF (Austrian Science Fund grant TRP71-N23) for the partial compensation of my travel expenses to the conferences and workshops I participated in.

Kurzfassung

Klinische Leitlinien und Protokolle enthalten methodische Vorgaben, um die medizinische Behandlungsqualität zu gewährleisten. Diese Leitlinien müssen jedoch – bevor sie in einer klinischen Anwendung verwendet werden können – in eine formale Sprache übersetzt werden. Dieser Vorgang ist sehr zeitintensiv und bedarf der Zusammenarbeit von Experten unterschiedlicher Fachrichtungen. In diesem Projekt haben wir eine Methodik mit dem Namen GOALS[1] entwickelt, welche den Übersetzungsprozess – unabhängig von der Zielsprache – unterstützt.

Die zeitlichen Konzepte einer Leitlinie werden mit Hilfe der Auszeichnungssprache TimeML annotiert. Diese bilden die Grundlage für die Anwendung von Informationsextraktionsmethoden (regelbasierte Algorithmen und Methoden des maschinellen Lernens), um schrittweise Teile einer Leitlinie automatisch in die Zielsprache zu übersetzen. An Hand eines konkreten Szenarios wird die Anwendbarkeit der Methodik gezeigt, indem zeitlich zusammenhängende Sätze eines klinischen Protokolls in ein semi-formales Modell transformiert werden.

[1]GOALS ist ein Akronym abgeleitet aus den einzelnen Schritten der Methodik

Abstract

Clinical practice guidelines and protocols aim at raising the quality of healthcare. They are written in a narrative style and have to be translated into a computer-interpretable format to be usable in clinical software applications. In order to ease this challenging and laborious task for the modeler we developed a methodology called GOALS[2]. It is specified independently from the target computer-interpretable guideline language and uses a guideline's text annotated with temporal concepts provided by TimeML as a starting point. It describes step-by-step how parts of the guideline's model can be generated and finally assessed by means of an evaluation scheme.

Information extraction techniques – machine learning algorithms and knowledge engineering methods – are applied to support the different steps in order to generate parts of the model automatically. A scenario-based application of GOALS shows the translation of temporally-related sentences of a clinical protocol into the corresponding semi-formal model.

Evaluation results are clear indicators for the GOALS methodology's easing of the time-consuming modeling process.

[2]GOALS is an acronym of the verbs defining the individual steps of the methodology

Contents

List of Figures

List of Tables

Part I

Analysis

CHAPTER 1

Introduction

1.1 Motivation

Clinical Practice Guidelines (CPGs) are defined as *"systematically developed statements to assist practitioners and patient decisions about appropriate healthcare for specific circumstances"* [FL90, p.38]. CPGs are published as textual guidelines, but in order to deploy them in some kind of computerized tool (e.g., a clinical decision support system) they have to be represented in executable computer-interpretable guideline (CIG) formalisms (e.g., Asbru, PRO*forma*, GLIF3 [dCBKH04, dCKH08, IM08, LBTvdHH10]).

Several editing/authoring tools exist for these languages, but still the authoring task remains complex and labor-intensive, and requires comprehensive knowledge in medical and computer science. For this reason, various approaches have been developed to deal with automated modeling using natural language processing (NLP) and information extraction (IE) methods (see for instance [KAM07, KSM11, MK13]). Major challenges in modeling – whether manually or (semi-)automatically – that have to be tackled, exist, because documents are often long and confusing, concepts are vaguely or incompletely described, and the text contains redundancies that have to be identified.

The complexity of a software-supported modeling process (in order to transform the contents of a guideline into its corresponding model automatically) is multifaceted. Various ongoing research projects are dealing with this challenge. The most extensive approach to our knowledge is the "SIMPLE" project – a research cooperation between the company "ID Berlin"[1] and various universities of Germany. It was started in 2011 and is aimed at translating and integrating guidelines into electronic hospital information systems [MH13] [2]. Up to now only the project setup has been announced, but no results have been published yet. Another project is "VeriCliG" which has been hosted at the

[1]ID Information und Dokumentation im Gesundheitswesen GMBH & CO KGAA, D-10115 BERLIN

[2]According to Mrs. Moreno, the leader of the project, more than 70 experts are still working on it - 2014

Free University of Bozen-Bolzano. It intends to help medical staff to save time and resources by automating the task of authoring and revising careflows with syntactic and semantic annotation techniques [TMC+13]. They base their approach on business process modeling notation (BPMN) model extraction from natural language texts.

Lexico-syntactic patterns were developed by Phil Gooch [Goo12b] in order to recognize clinical concepts, events, temporal relations, disambiguated terms, and abbreviations in clinical texts. Additionally, he focused on the resolution of coreferential and anaphoric relations in discharge summaries and progress notes. He adapted tools and resources from the biomedical domain to identify processes of care in clinical narratives.

A further, multi-step approach based on information extraction methods was developed by Kaiser et al. [KAM07] in order to ease the formalizing of treatment processes in clinical practice guidelines. The approach contained several heuristics which were applied to guidelines in the medical discipline of otolaryngology.

To sum up, what these approaches have in common is the focus on the identification of clinical care-paths. These care-paths contain temporal concepts to describe the chronological order of clinical activities [TGS08]. Therefore, the use of temporal reasoning methods based on a guideline's temporal representation (consisting of temporal expressions, concept primitives, and temporal relations [SRU13]) may lead towards a (semi-) automatic modeling approach. The challenging, various vague and/or complex temporal dimensions in a guideline are illustrated in Figure 1.1.

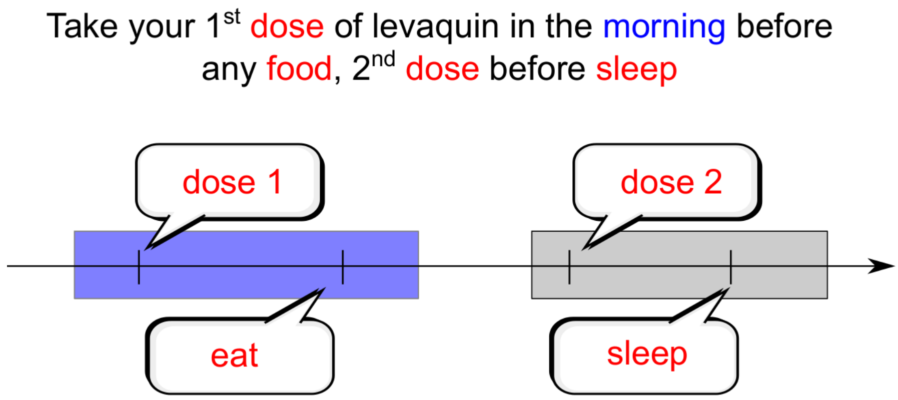


Figure 1.1: Temporal vagueness in clinical process descriptions (taken from [Ver12]).

The temporal expression 'morning' describes a period which is not bound to exact timestamps. The word 'before' defines a temporal relation between the first administration of the drug and breakfast – both within the 'morning'-period. The second dose must be taken before falling asleep – here again 'before' defines a temporal relation. For clinical staff it is obvious that this has to happen in the evening without defining a temporal anchor, like 'at night'.

The formalized description of temporal concepts and relations in such sentences is feasible by means of the specification language TimeML. This was developed for annotating events and temporal expressions in natural language. The TimeML addresses the problems of (a) identifying an event and anchoring it in time, (b) chronologically arranging events, (c) reasoning with contextually underspecified temporal expressions, and (d) reasoning about the duration of events [PCnI+03].

TimeML has become the de-facto standard for the annotation of events and temporal expressions in natural language [KPS05] and can be used for reasoning on events. The TimeML specification ([PCnI+03]) was developed in the news-wire domain, but also applied to clinical narratives and to clinical discharge summaries [SBS+09]. As CPGs describe activities (corresponding to events) and contain (temporal) expressions that allow ordering and relating them, we propose applying TimeML for annotating. Based on these annotations, knowledge engineering methods as well as machine learning algorithms will help to detect and to extract the various information dimensions in order to generate parts of the CIG model. We split the challenge into the following sub-tasks:

- Application of the TimeML specification in order to describe the temporal relations between temporal concepts.
- Reuse and adaptation of existing software tools for the generation of TimeML annotations.
- Use of temporal relations to identify the different information dimensions and aspects in a guideline.
- Identification of condition-based activities that have a major impact on the clinical care-paths, based on their subordination relation, a specific relation defined in TimeML.
- Automatic generation of CIG model parts describing condition-based activities.
- Extension of TimeML to facilitate the automatic modeling approach of condition-based activities within clinical guidelines.
- Development of a methodology which defines steps that lead to a (semi-) automatic generated model of a guideline.

Overview of the thesis:

We divided the thesis into three parts:

Part I contains the introduction of our research topic and describes our hypothesis. The related work sections discuss the present research activities in the fields of our work.

Part II contains the main topics according to our defined sub-tasks listed above. At first we describe our GOALS methodology, followed by implementations of information extraction methods to fulfill selected steps of GOALS. All these chapters are equally structured. Firstly, the knowledge sources and tools are discussed. Secondly, the scientific methods are described and thirdly, evaluated. A conclusion of every chapter describes the contribution to our research goal. A scenario-based approach illustrates the proper functioning of our GOALS methodology.

Part III sums up the obtained results delivering answers to our research question. The future work section lists activities which will be executed in upcoming research projects.

1.2 Research Questions

Researchers in artificial intelligence (AI) are working on information extraction (IE) methods in natural language processing (NLP) to locate and extract important information from unstructured natural texts. As clinical practice guidelines are also written in an – mostly unstructured – essay style, these methods can support the translation of the guidelines' texts into their formal representations. This translation process requires – whether manually or automatically executed – the identification of the various information dimensions of a clinical guideline. The most prominent one is the description of control flow related aspects containing condition-based clinical activities. We focus our prospective research activities on the temporal relation between the *condition* and the *action* in order to answer the following central research question:

Can the temporal relations among condition-based clinical activities be detected automatically by use of IE methods in order to support the manual modeling task of a guideline into its formal representation?

Based on this question, the following hypotheses can be generated:

1. If model relevant *condition-action* sentences are related to recurring linguistic and semantic patterns, information extraction methods may be used to identify the antecedent and the consequence of a conditional sentence of a guideline.

2. The medical concepts – identified by means of semantically grounded medical knowledge bases, such as the UMLS and its Semantic Network – contained in these sentences, may help to distinguish the different information dimensions of a guideline.

3. The introduction of a weighting coefficient may help to determine the relevance of a *condition-action* sentence to be included in the formal model.

4. Specification formalisms for annotating events and temporal relations in narrative texts (e.g., TimeML) from other domains may also be applied to clinical guidelines in order to describe the temporal relations among clinical activities.

5. The antecedent and the consequence of a *condition-action* sentence are temporarily related. This relation may be used for automatic reasoning in order to support the transformation process of a guideline.

6. If it is possible to extend the specification of TimeML to describe condition-based activities of clinical guidelines in a formal way, a mapping of the temporal model into a guideline model may be conceivable.

Proving the hypotheses described above, the manual modeling process of a guideline will be simplified substantially.

1.3 Publications, Workshops and Conferences

Our thesis includes methods and results from the following publications ([WK13, WK14a, WK14b, WK15]):

- Reinhardt Wenzina and Katharina Kaiser. Identifying Condition-Action Sentences Using a Heuristic-based Information Extraction Method. In *Proceedings of the Joint International Workshop: KR4HC'13+ProHealth'13*, pages 17–29, 2013.

- Reinhardt Wenzina and Katharina Kaiser. Towards the Application of TimeML in Clinical Guidelines. In *Modellierung im Gesundheitswesen. Tagungsband des Workshops im Rahmen der Modellierung 2014*, pages 37–48, 2014. ICB-Research Report.

- Reinhardt Wenzina and Katharina Kaiser. Using TimeML to Support the Modeling of Computerized Clinical Guidelines. *Studies in Health Technology and Informatics*, 205:8–12, 2014.

- Reinhardt Wenzina and Katharina Kaiser. GOALS - Modeling Clinical Guidelines Based on TimeML Concepts, May 2015. Accepted at the 5th International Conference on Digital Health 2015. http://dx.doi.org/10.1145/2750511.2750520.

CHAPTER 2

Related Work in Formalizing Guidelines

Clinical practice guidelines (CPGs) – also known as medical guidelines – are accepted as instruments for improving the quality of healthcare. They are designed to support the decision-making process of practitioners, clinical medical staff, and patients concerned. The guidelines should always be based on the latest findings of medical knowledge and, therefore, have to be revised and updated regularly.

In order to implement such a guideline for a medical decision support system, the content has to be manually translated into a computer-interpretable model which is a rather time consuming task. The time delay, however, should be kept as short as possible to ensure that the patients are treated ideally – relying on the latest scientific evidence. Based on cultural differences, constraints, and often resources, guidelines have to be adapted to different care settings. These adapted guidelines are called protocols, this is to distinguish them from the original guidelines. Research shows that a CPG without adaptation often fails to be effective in clinical applications [GR93]. Protocols typically provide detailed information about duration, dose, treatment procedures etc. and, therefore, omit a lot of scientific medical knowledge from the original guideline. As a consequence, often only the procedures that can be described in a formal way, like business workflows, remain.

Knowledge engineers and medical experts have to work together to generate a computer-interpretable guideline (CIG) in formalized languages, like Asbru, GLIF3, PRO*forma*, etc. (for an overview see [PTB+03, MvdAP07, IM08, Pel13]). These languages were mostly developed independently, but share many characteristics. The main reasons why they have not been widely deployed yet, are controversially discussed (e.g., [SSTM+08], [ZPJH10]).

Initial research on CIG representations was done in the 1990s and early 2000s. A methodological review of Peleg in 2013 [Pel13] unearthed still emerging topics in CIG research caused by many open challenges. The identified topics (illustrated in Figure 2.1) are arranged according to the CIG's life-cycle.

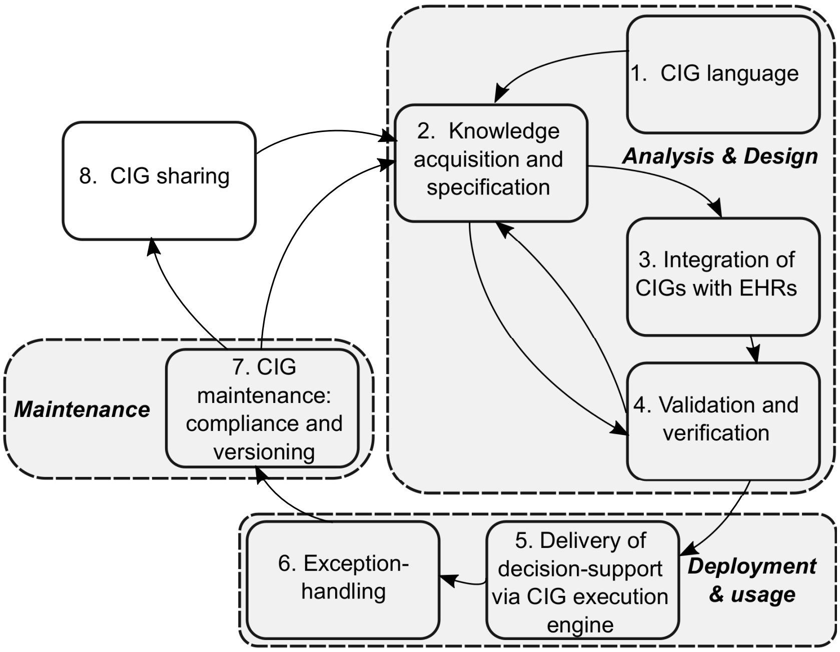


Figure 2.1: Emerging topics in CIG research (taken from [Pel13]).

In the analysis and design phase the modeling language is used to generate the model of a guideline by teams of knowledge engineers and domain experts. They combine the content of a CPG with implicit medical knowledge and create the corresponding CIG model by means of specific authoring tools. Sometimes adaptions have to be made to integrate EHRs (electronic health records) into the models and to apply them to different workflow settings. Validation and verification steps ensure the accuracy of the formal model. In case of discrepancies, the model has to be reviewed/redesigned until its correctness is guaranteed – proved by means of expert validation, formal verification, and validation through testing. In the deployment and usage phase the developed model is used for decision support. Possible exceptions during the execution are caught by specific error handling mechanisms. The maintenance phase comprises different compliance checks and versioning methods in order to ensure a model up to the latest medical findings. The sharing of CIGs helps to reduce the efforts for encoding by reusing the model or at least model parts.

Approximately, thirty guideline formalisms have been developed, but none of them has gained a leading position yet [SH06]. In the following section we discuss a selection of CIGs which is based on a literature study including articles describing the formalisms as *'well-known approaches'* [dCKH08, p.23], and as *'major ontologies in use'* [ZPJH10, p.5].

2.1 Executable CIG Formalisms

CPGs have to be encoded in executable CIG formalisms in order to be used in clinical decision support systems and, hence, in daily clinical practice.

2.1.1 Asbru

The Vienna University of Technology (Institute of Software Technology and Interactive Systems) and the Stanford Medical Informatics were the leading partners in the Asgaard project [SMJ98]. One of the project's goals was the development of a formalism to represent the semantics of clinical guidelines and protocols called Asbru. The following specific requirements concerning the representation formalism were met:

- Representation of continuous actions and states,
- description of intentions, conditions, and states as temporal patterns,
- expression of vaguely and incomplete temporal concepts by bounding intervals,
- execution of plans in parallel, sequential, and repeating modes or in a combination of them, and
- definition of intentions and preferences for each plan.

Reusable templates – called skeletal plans – describing actions and roles were used to describe domain-specific procedural knowledge. These templates were hierarchically grouped in plan specification libraries together with the corresponding arguments and time annotations. The main components of a plan were (1) a compulsory name, (2) a set of arguments (e.g., time annotation), (3) different knowledge roles (e.g., constraints, intentions, etc.), and (4) a plan body containing the description of the activities and their temporal ordering [Mik99]. The basic building block of a plan is the temporal pattern shown in Figure 2.2.

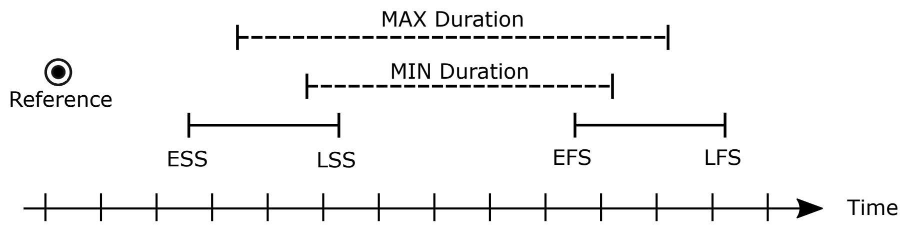


Figure 2.2: Time interval in Asbru (adapted from [SMJ98]); Reference: temporal reference, ESS: earliest starting shift, LSS: latest starting shift, EFS: earliest finishing shift, LFS: latest finishing shift – resulting annotation: ([ESS,LSS],[EFS,LSS],[MIN Duration, MAX Duration], Reference).

The unspecific time stamps for the beginning, the termination and the duration of an interval are managed by this flexible pattern. Additionally, a temporal reference concerning the beginning of every template can be specified.

Several tools were implemented to support the adoption of Asbru. The Document Exploration and Linking Tool with Add-ons DELT/A [VMK04] is a general authoring tool for clinical guidelines and facilitates a specific interface for applying Asbru. The graphical tool AsbruView supports the development and visualization of Asbru encoded guidelines [MKSJ98] and CareVis provides an interactive visualization approach to represent computerized protocols and temporal patient data [AM06].

2.1.2 GLIF

The Guideline Interchange Format (GLIF) was developed by the InterMed Collaboratory at Stanford Medical Informatics, Harvard University, McGill University, and Columbia University. In comparison to other CIGs, one of GLIFs development principles was to design a standardized, sharable language for modeling and disseminating clinical guidelines to be used by different medical institutions and system platforms. In 2000 the latest version GLIF3 [PBO+00] was developed and supports, amongst other features, the computer-based execution of a guideline's model. The GLIF3 specification consists of an extensible object-oriented model (the major classes in GLIF3 are shown in an UML class diagram in Figure 2.3) and an RDF[1]-based syntax [BPT+04].

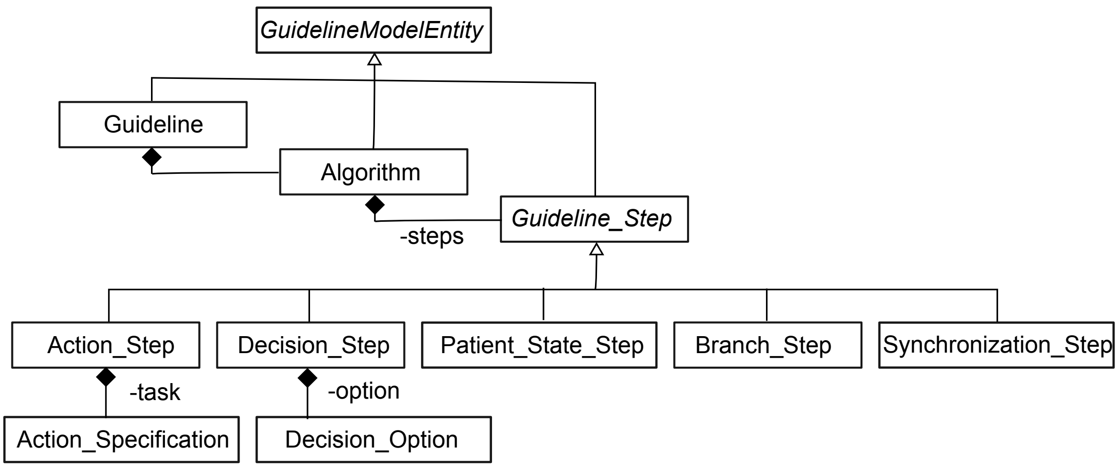


Figure 2.3: High-level view of the major classes in GLIF (taken from [BPT+04]).

The specification of GLIF distinguishes three levels of abstraction:

Conceptual Level: Flowcharts are used to represent the temporal process flow in guidelines by defining steps for actions, decisions, concurrency definitions (e.g., branch and synchronization), and patient states.

[1] http://www.w3.org/TR/rdf-schema/ (last accessed March 16, 2015)

Computable Level: Logical consistency and completeness checks are enabled by adding information about decisions, patient data, and iterations.

Implementable Level: The specification can be incorporated into particular institutional information systems (e.g., by mapping actions to institutional procedures).

A model can be authored by loading an ontology scheme and a specific graph widget into the Protégé tool [GMF+03]. The encoding of medical decisions (e.g., automatable 'case steps', and non-automatable 'choice steps' describing interactions through medical staff) is done by means of GELLO – an object-oriented query and expression language [SOBG03]. The Guideline Execution Engine (GLEE) provides the features for the execution of guidelines encoded in GLIF3 [WS02]. Such guidelines are practically applied in various environments (e.g., at the Columbia University for post-CABG – Coronary Artery Bypass Grafting – patient care planning).

2.1.3 PRO*forma*

The Advanced Computational Laboratory of Cancer Research in the UK has been working on PRO*forma* since the Nineteen-nineties [SF03]. PRO*forma* is a formal knowledge representation language (expressed by a first-order-logic) for developing and publishing executable guidelines. The goal is to develop a formal representation language which (1) can be used to describe different clinical processes, (2) is applicable to different clinical domains, (3) contains a minimal set of concepts to be easy to use, (4) is executable by machines, (5) owns a sound semantic, and (6) offers mechanisms for consistency checking.

A PRO*forma* model consists of a set of tasks and data items. The tasks are hierarchically organized in a network and can be parameterized (e.g., constraints, pre- and post-conditions, etc.). Generally, tasks can be of class 'Plan', 'Decision', 'Action', or 'Enquiry'. Figure 2.4 shows the symbols of the four classes[2] on the left-hand side and a sample guideline diagram [SF03] on the right-hand side:

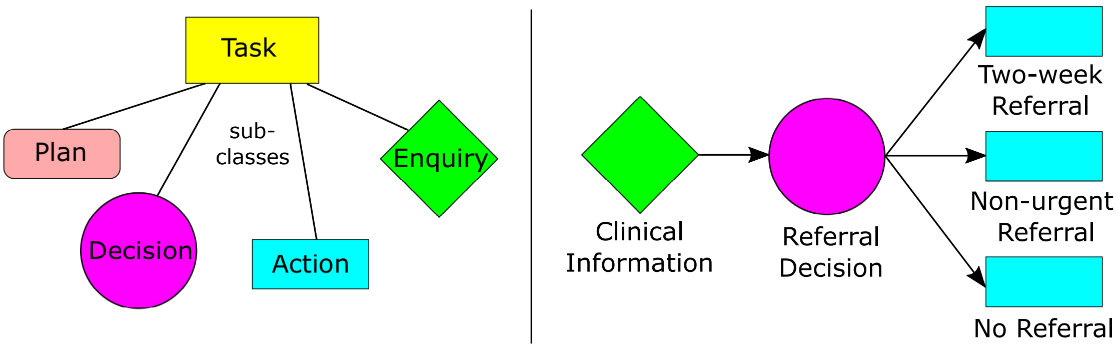


Figure 2.4: Left: classes of PRO*forma* – Right: sample diagram (taken from [SF03]).

[2]http://archive.cossac.org/proformaModelling.html (last accessed March 16, 2015)

Plan is a set of tasks, which may contain sub-tasks as well as other plans.

Decision represents a specific point in the process flow of a task, where a decision has to be made based on different options, relevant information, a set of rules, and/or according to current data (e.g., whether to treat a patient or to make a referral).

Action describes some procedure independently from the machine executable workflow (e.g., a specific task to be performed by a person, like the administration of an injection).

Enquiry represents information deficits to be remedied by persons or by means of external data sources.

PRO*forma* has been successfully applied in clinical applications, but shows some limitations. The model is specified in a proprietary formalism and does not use existing standards for data storage and exchange. Furthermore, the modeling of patient data is limited.

Different authoring and execution tools for PRO*forma* have been developed. The Arezzo Composer and the Arezzo Performer[3] are commercial products, implemented and distributed by InferMed Ltd. The Tallis toolset[4] is freely available for collaborative research at the Laboratory of Cancer Research.

2.1.4 SAGE

In a joint project, the IDX Systems Corporation, Apelon Inc., Intermountain Health Care, Mayo Clinic, Stanford Medical Informatics, and University of Nebraska Medical Center developed the Standards-based Shareable Active Guideline Environment (SAGE) in 2002 [TCG+07]. The SAGE project was built on the research results of the EON guideline representation ontology [MTDS96]. Its main goal was to develop an infrastructure to execute guidelines across different information systems based on the following sub-tasks:

1. Specification of a sharable guideline model.
2. Usage of existing knowledge-authoring tools, like Protégé, to develop an open source toolkit.
3. Application of standard medical terminologies for the representation of clinical concepts (e.g., SNOMED-CT[5], LOINC[6], HL7[7]).
4. Integration of technical interfaces to support clinical information systems of different vendors.

[3] http://www.infermed.com/en/Clinical-Decision-Support/Arezzo-Pathways-Solutions.aspx (last accessed March 15, 2015)

[4] http://archive.cossac.org/tallis/index.html (last accessed March 16, 2015)

[5] http://ihtsdo.org/snomed-ct/ (last accessed March 16, 2015)

[6] https://loinc.org (last accessed March 16, 2015)

[7] http://www.hl7.org/ (last accessed March 16, 2015)

A guideline or a guideline segment, respectively, are modeled as a sequence of recommendation sets in order to represent workflows, roles, entities and actions within health organizations. These sets combine specific model elements, like 'Context', 'Decision', 'Action', and 'Route' and can be structured in a cyclic or iterative way:

Context Node builds the basic building block of a model and specifies and declares the context (trigger events, clinical settings, patient states, etc.) in which it is applied.

Decision Node indicates the need of data (e.g., from the electronic medical record) in order to select the correct activity path.

Action Node models activities of the information system (e.g., messaging to system devices, goal specification, database retrieval, etc.).

Route Node acts as synchronization element to join previously forked activity paths.

The authoring of a guideline in the SAGE specification is done by Protégé and a customized plug-in called Kwiz[8]. The first execution engine was developed by Ram et al. [RBT+04] and tested with an immunization guideline encoded in a SAGE guideline model, but only in a simulated environment at Mayo Clinic and University of Nebraska Medical Center. Another attempt was made by Kim et al. [KCK08] who proposed to convert SAGE-based guidelines into a formalism executable by commercial engines.

2.1.5 NewGuide

The NewGuide project was started in 2001 by the Laboratory for Medical Informatics, Department of Computer and System Science at the University of Pavia, Italy [CQK02, CCQS05]. The project focused on the development of a component-based, multi-level architecture framework in order to provide an effective medical knowledge management including the representation and execution of clinical practice guidelines. The implemented architecture allows different views of the formalized knowledge (e.g., the patient's view, the phycisian's view, etc.) and the integration of the electronic patient record in the workflow system to react on dynamic changes (e.g., patient conditions). Additionally, it offers evaluation and control strategies for the elicited knowledge and supports feedback mechanisms for guideline based care paths.

The concept of Virtual Electronic Medical Records[9] (vEMR) is implemented in the framework and this framework consists of the following components:

- An editor to formalize a clinical guideline and to generate its model,
- a repository for the models,

[8] http://protege.cim3.net/cgi-bin/wiki.pl?SAGE (last accessed March 6, 2015)

[9] http://wiki.hl7.org/index.php?title=Virtual_Medical_Record_(vMR) (last accessed March 16, 2015)

- an inference engine working in a multi-user environment, and
- a logging system that allows to track the individual decision processes.

The overall focus of the project is the careflow, its exception management, and the flexibility to follow another part of the care path as recommended.

The framework is implemented on a Java 2 Enterprise Edition (J2EE[10]) platform and uses the Simple Object Access Protocol (SOAP[11]) to communicate with legacy systems. It is applied in four hospitals in the Lombardia region to support the management of stroke patients.

2.1.6 GLARE

GLARE is a prototypical system to acquire and execute clinical guidelines, developed by the Computer Science Department of the Universita del Piemonte Orientale of Alessandria (Italy) in cooperation with Azienda Ospedaliera S. Giovanni Battista of Torino [TMT01]. It specifies an expressive formalism to deal with clinical guidelines of different areas and various temporal aspects.

The representation language defines the following types of actions: Plans, query actions, decisions, work actions, and conclusions (see Figure 2.5).

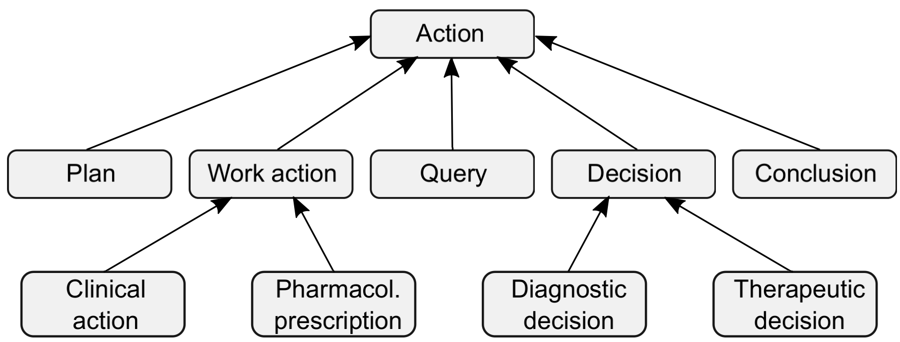


Figure 2.5: Hierarchy of action types in GLARE (taken from [TMB+04]).

Plans may consist of hierarchically organized actions.

Query actions specify the need for additional information to enable decisions about alternative care paths.

Work actions describe activities executed by the clinical staff.

Conclusions specify the different outcomes of a decision process.

[10] http://www.oracle.com/technetwork/java/javaee/overview/index.html (last accessed March 16, 2015)

[11] http://www.w3.org/TR/soap/ (last accessed March 16, 2015)

As temporal constraints play an important role in clinical guidelines, GLARE defined a "high-level" representation formalism to deal with temporal aspects, like 'temporal indeterminacy', 'constraints about duration', 'delays between actions', and 'periodic repetitions of actions' [TMB+04]. Additionally, qualitative temporal constraints such as 'before', 'after', and 'during' are supported.

GLARE's knowledge authoring tool offers a user-friendly interface usable also by non IT experts and supports the detection of syntactic and semantic inconsistencies. The encoded guideline is executed by a flexible engine which can be integrated in clinical decision support systems.

2.1.7 HELEN

A modular framework named HELEN has been developed for the Department of Neonatology of the Heidelberg University Medical Center [SGvdH+04]. The framework comprises a tool for authoring, a server for web-based browsing, and an execution engine limited to specific elements of a CPG and was published under the GNU public license. HELEN offers algorithmic steps, like actions, decisions, branching, and nested subplans – comparable to GLIF. HELEN's modular document model is illustrated in Figure 2.6.

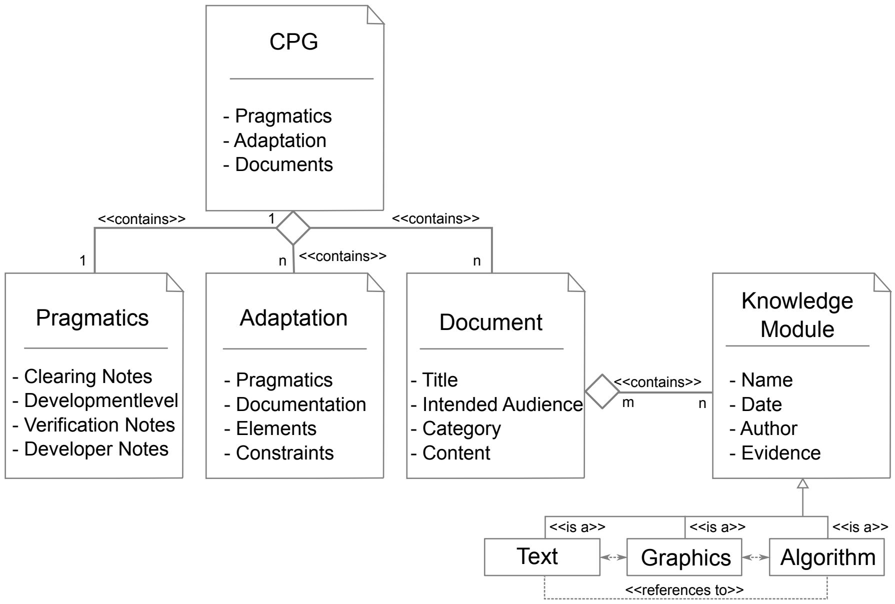


Figure 2.6: Modular document model of HELEN (adapted from [SGvdH+04]).

In a further extension of the specification, HELEN [SGL04] focused on

- the support of guidelines from different medical domains,
- the integration of adaptation mechanisms for the life-cycle of CIGs,
- the application of the models in the daily clinical work, and

- the creation of a central repository of literature references.

The Protégé-2000 toolset with specific plug-ins was used as knowledge acquisition tool. The *Guideline Viewer* was implemented as a JavaServlet executed on an Apache-Tomcat application server. The server-based *Guideline Execution Engine* was responsible for the traversing of the encoded workflows and the interaction with the clinical staff to make decisions, actions, or recommendations.

2.2 Intermediate CIG Representations

In addition to the executable formalisms above, semi-formal and semi-structured representations of clinical guidelines have been developed in recent years – the most important ones are discussed below.

2.2.1 MHB

The Many-Headed Bridge (MHB) [SMPC+05, SMM+06] specifies an intermediate representation language which provides a more semi-structured and less formal format than executable CIGs and, hence, bridges the gap between the free-text guideline and its corresponding formalized model (e.g., Asbru, GLIF, etc.).

In the modeling process the guideline document is split into chunks of information (e.g., a paragraph, a sentence, etc.) representing independent information dimensions (control flow, data flow, temporal aspects, evidence, background information, patient-related aspects, resources, and document structure) and their corresponding aspects. The dimensions and aspects are described by means of an XML-based syntax (Figure 2.7 illustrates a minimal MHB model where sets of chunks are grouped in a 'chunk-group'). An example of a 'control flow dimension' – one of the most frequent dimensions found in guidelines – is shown in Figure 2.8.

```
<xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE root SYSTEM "~/MHB_1.03.dtd">
<!--MHB document created by r. wenzina using DELT/A on 22/02/15-->
<root>
    <chunk-group title="chapter">
        <chunk-group title="subheading">
            <chunk chunk-id="#CHUNK-00001">
             .......
            </chunk>
        </chunk-group>
    </chunk-group>
</root>
```

Figure 2.7: Minimal MHB model in XML format.

The usage of MHB is described in an annotation guideline developed in the PROTOCURE[12] project [SMV04]. The DELT/A tool (see 2.1.1) supports the manual modeling process.

Example sentence:

Intravenous administration of hydrating or glucose solutions should be reserved for those patients who refuse to eat with a protracted labour.

MHB representation:

```
<control>
    <if-then condition="refuse to eat AND protracted labour"
              result="intravenous administration of hydrating or
                       glucose solutions"
              degree-of-certainty="should"  />
</control>
```

Figure 2.8: Example of a sentence modeled as 'control flow dimension' in MHB (taken from [KS10]).

In 2011 the decision was taken to split MHB into two specifications to reduce the problems for non-IT experts. The MHB-F [SK11] describes guidelines in free text, MHB-S additionally contains semantically enriched information.

2.2.2 GEM

The Guideline Elements Model (GEM) aims at the representation of clinical guidelines in an XML-based format [SKA+00]. It was developed at the Yale University in 2000, updated in the following years and adopted as an international ASTM (American Society for Testing and Materials) standard. GEM defines more than 100 hierarchically structured elements to represent the heterogeneous information of a guideline. The high-level concepts of the model are shown in Figure 2.9. It contains the root element 'Guideline Document' and its child elements for identity, developer, purpose, intended audience, method of development, target population, knowledge components, testing, and revision plan.

The modeling of a guideline is a markup process supported by the tool GEMCutter[13] and needs no programming skills. The resulting model is an abstraction of the guideline document. Consequently, it also contains ambiguities frequently found in guidelines – an often critiqued drawback of GEM [HKMS11]. The applicability of GEM was approved in a real world example. Shiffman et al. translated a guideline on management of chronic asthma into its model [SMET04] which has subsequently been integrated in a decision support system that operates within the Logician Electronic Health Record system.

Georg et al. [GSB05] proposed extensions to the representation elements of decision processes. The authors' intention was to make a step towards the automatic generation of decision rules based on guidelines' texts. A translation of the model from XML into

[12] http://www.openclinical.org/prj_protocure.html (last accessed March 16, 2015)
[13] http://www.openclinical.org/dld_gem.html (last accessed March 16, 2015)

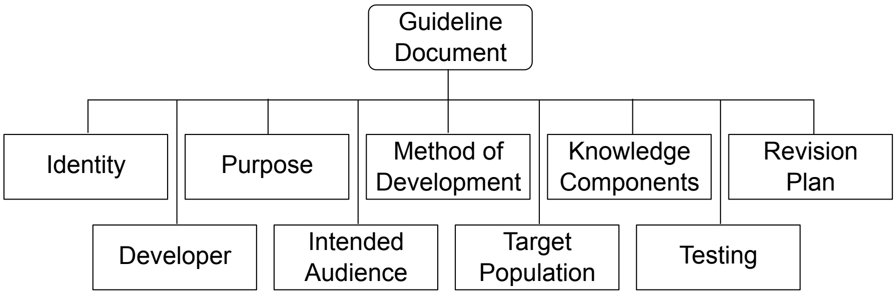


Figure 2.9: High-level concepts of the Guideline Elements Model (taken from [SKA+00]).

OWL[14] was an important step to disseminate GEM in medical informatics [TMKS09].

2.2.3 Hybrid-Asbru

Hybrid-Asbru is an extension of Asbru, developed as part of the DeGeL project [YSL+07]. It aims at the support of a gradual conversion process of clinical guidelines into their machine-comprehensible representation. Figure 2.10 gives an overview of this step-wise process.

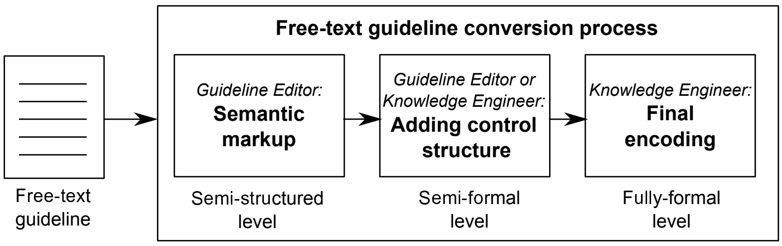


Figure 2.10: Hybrid-Asbru: General conversion process (adapted from [YSL+07]).

In the first step the guideline editor transforms the guideline's narrative text into the semi-structured Asbru representation format by applying knowledge roles of the original Asbru ontology (e.g., intentions, conditions, and plan-body) as well as newly defined knowledge roles (e.g., clinical settings, and actors). The semi-structured format is then extended by control structures (e.g., in parallel) in order to deliver the guideline's contents in a semi-formal format. This step does neither assume any programming skills nor the knowledge of the exact syntax of Asbru, and, therefore, can be executed by a guideline editor and does not require the expertise of a knowledge engineer. In the final step, the semi-formal model is translated by a knowledge engineer into the fully-formal Asbru language.

[14] http://www.w3.org/TR/owl2-syntax/ (last accessed March 16, 2015)

This gradual transformation process is supported by the Gesher tool, which offers a graphical user interface for all parties involved (expert physicians, clinical editors, and knowledge engineers) [HYSS08].

This chapter gave an overview of related work in formalizing clinical guidelines. Although research on CIG representations started nearly 20 years ago, it is still an active research field. During these two decades executable CIG formalisms as well as intermediate CIG representations were developed to (semi-) formalize the knowledge contained in clinical guidelines. In the closing discussion the characteristics and different application areas of selected formalisms were presented.

CHAPTER 3

Related Work in Information Extraction

Information extraction (IE) is a field of computational linguistics and aims at extracting entities, relations, and events (*"who did what to whom where and when"* [PY13, p.23]) from unstructured and/or semi-structured machine-readable documents by means of natural language processing (NLP). It can be seen as a *"process of identifying within text instances of specified classes of entities and of predications involving these entities"* [Gri12, p.2]. In general, an IE system is built for a specific domain or topic and provides relevant information for the user by identifying predefined classes, but ignores the rest.

Semi-structured texts – in comparison to unstructured texts – implicitly provide information depending on the physical layout of the document (e.g., an HTML file contains tags to markup the document's structure – 'head', 'body', 'headline', etc. and, therefore, automatically adds a certain meaning to the enclosed phrases). Hence, information can already be extracted based on its position in the document. In recent years also other sources of information than documents (e.g., images, audio-files, etc.) have been used for content extraction.

OpenIE is an upcoming research field which focuses on the extraction of factual information from multiple documents of different domains, unprecedented heterogeneity, and massive size [EFC+11]. Consequently, it has to deal with the additional challenges of cross-document coreference resolution, information fusion and velocity.

Challenges and competitions in information extraction have a long history going back to the late 1970s. The series of Message Understanding Conferences[1] (MUC) from 1987 till 1998 – initiated and financed by the Defense Advanced Research Projects Agency (DARPA) – encouraged scientists to develop new IE methods and concepts. The goal was to find the most mature solutions for the tasks given. The Automatic Content Extrac-

[1] http://www.itl.nist.gov/iaui/894.02/related_projects/muc/proceedings/muc_7_toc.html (last accessed March 1, 2015)

tion[2] (ACE) evaluations continued this kind of competitions and aimed at the automatic processing of human language from a variety of sources (such as newswire, broadcast conversation, and weblogs) in textual form. In the past few years NLP challenges were increasingly organized as parts of conferences and workshops (e.g., semEval[3], CoNLL[4], i2b2[5]).

Metrics have been developed in order to compare the quality of different IE methods. The most prominent measures are *Recall*, *Precision*, and the *F-Measure* [WLCF+94, FH98]. In order to calculate them, a corpus of documents is analyzed by experts and the identified information is defined as gold standard. Then the output of the IE method is compared to the gold standard and the number of correctly and incorrectly identified items is determined.

Precision shows how good the IE method sorts out what is irrelevant by relating the number of correctly identified information items to the number of all identified items. *Recall* compares the number of correctly identified items to the actual number of correct items in the gold standard. Consequently, it shows how accurately the system finds what is relevant. The discussion which measures provide better answers depends on the general goal of the task. The F–measure tries to combine both values – as a weighted harmonic mean – in order to compare IE systems with each other (the measures are discussed in detail in chapter 6.2.3).

3.1 Processing Steps in Information Extraction Systems

The typical IE system consists of five levels of processing [HR10] in order to identify

1. complex words – recognition of named entities, such as people, companies, countries, temporal expressions, numeric values, etc.

2. basic phrases – segmentation of sentences into noun groups, verb groups, and particles

3. complex phrases – identification of complex noun groups and verb groups

4. domain events – generation of semantic structures based on patterns which can be applied to the identified words and phrases from the previous levels

5. merging structures – combination of semantic information concerning the same entities spread over the whole text to fill the information slots of the entity's template (in general a template is an attribute-value pair describing an entity or its relations).

[2] http://www.itl.nist.gov/iad/mig/tests/ace/ (last accessed March 1, 2015)
[3] http://alt.qcri.org/semeval2015/ (last accessed March 16, 2015)
[4] http://www.aclweb.org/portal/content/conll-shared-task-2015 (last accessed March 16, 2015)
[5] https://www.i2b2.org/ (last accessed March 16, 2015)

The last level, additionally, comprises the identification of relations between the found entities [Kon14] and the generation of templates representing specific scenarios [AHA14]. In order to implement these identification tasks, the process has to be split into a series of sub-tasks. Every task can be modeled with "cascaded finite-state transducers" which are arranged in a pipeline where the output-data of one transducer forms the input-data of the following one. The general architecture of such a processing pipeline [App99] is shown in Figure 3.1.

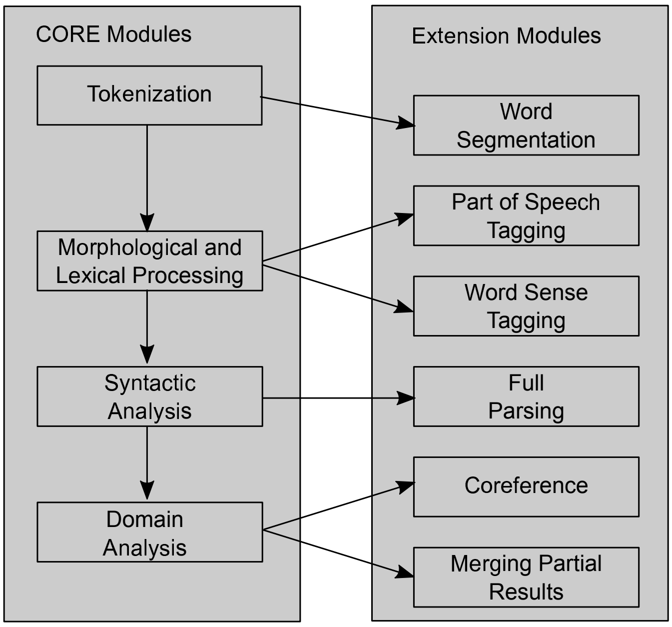


Figure 3.1: General architecture of an information extraction system (taken from [App99]).

Tokenization is the process of splitting documents into independent lexical units (e.g., paragraphs, sentences, words, etc.). Its complexity heavily depends on the language of the document. In English-written documents, for example, the boundaries of words are identified by their surrounding white spaces or punctuation characters. However, for non-European languages, like Japanese, this is a major challenge and needs additional word **segmentation** modules.

Morphological analysis deals with the rules on which words and forms of words (e.g., the suffix *-ly* indicates a word to be an adverb) are built. Based on these rules, words of a text are analyzed, the generic word identified and searched in a dictionary in order to determine its **'part-of-speech' (POS)** (e.g., noun, verb, etc.). Subsequently, the word is annotated with its corresponding POS-tag. These tags also help to reduce **word sense** ambiguity.

Another core functionality of this standard module is the Named Entity Recognition (NER), which is used to find out the meaning of a word or word group (e.g., 'New York' is annotated as a location). This task is achieved by manually created rules (Knowledge Engineering approach) or machine learning algorithms (see [NS07] for a comparison). Studies show that current named entity recognition systems have success rates of nearly 90% – similar to human performance [HR10].

The goal of the **syntactic analysis** is to identify predicate argument structures [SHWA03] – mostly associated with content verbs and noun phrases – where predicates take one, two, or more arguments. Depending on the domain, 'shallow parsing' may be sufficient, because mostly small sets of domain-relevant events and relationships are investigated. The **full parsing** process scans the whole text to find these structures, hence, it is very slow depending on the number of words in a sentence.

The **domain analysis** builds the core of an IE system. It uses all information generated from the previous steps to extract the relevant domain-specific knowledge by means of IE patterns or machine learning algorithms. The **coreference** resolution (see current trends in [CGB08]) is a module which identifies references between entities (e.g., expressions referring to the same person) and provides the basis for **merging** the contents of different IE templates executed by specific rules.

3.2 IE System Design

The implementation of the different process steps (e.g., named entity recognition, etc.) is done either by means of rule-based techniques, machine learning methods, or a combination of both – called hybrid approaches. Chiticariu et al. [CLR13] compared the spreading of the different techniques in industry and the scientific world. They discovered that rule-based systems are more prominent in commercial systems than in research (see Figure 3.2).

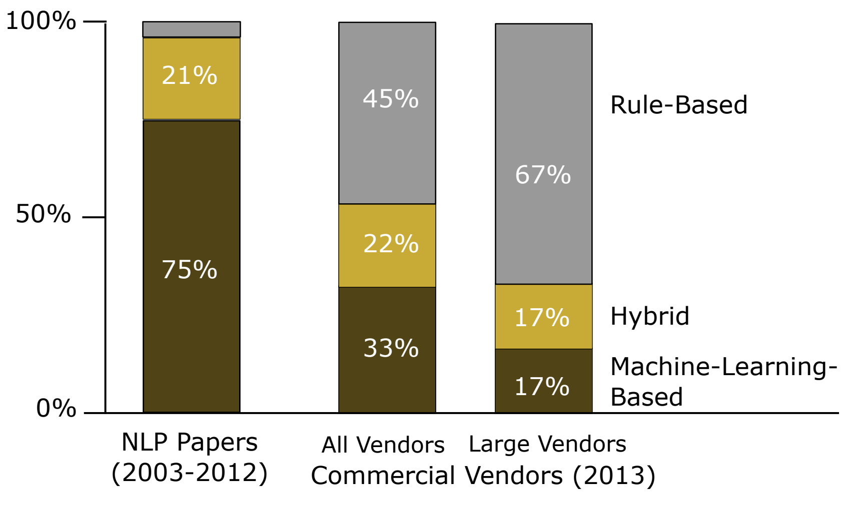


Figure 3.2: Comparison of used IE techniques for entity extraction in research and commercial systems (taken from [CLR13]).

They argued that the dominance of rule-based systems was closely connected to the evolving research field "Big Data Analytics" over documents written in an unstructured narrative style. The results of the study were based on the analysis of papers of NLP conferences about entity extraction compared to software tools applying this extraction technique in industry. They also proposed the definition of standard IE rule languages and data models in order to provide a sound scientific base for commercial IE products in the future.

3.2.1 Rule-Based Approach

In a rule-based IE approach, knowledge engineers and domain experts work together to generate rules for the identification of predefined classes in an unstructured text. The development of a domain-specific grammar based on these rules is very time-consuming, cumbersome and requires considerable skills [HR10].

The life-cycle of developing a rule-based information extraction system [Pav11] is shown in Figure 3.3.

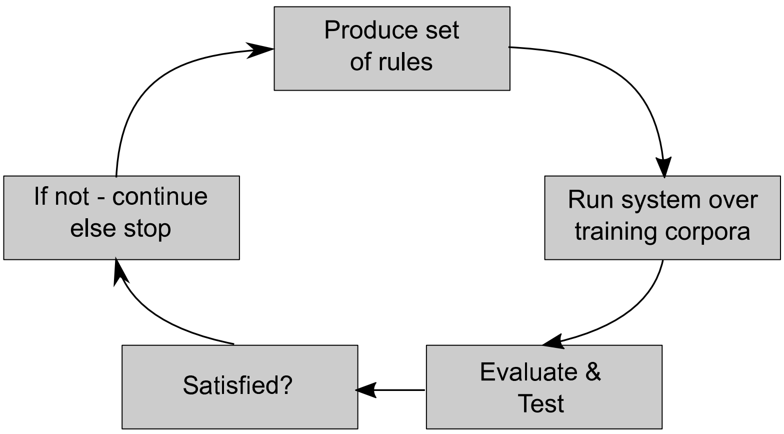


Figure 3.3: Development of a rule-based information extraction system (taken from [Pav11]).

Based on a small number of training sentences (training corpus), patterns of concepts are analyzed and corresponding rules developed. Then the rules are applied to the training corpus and the results evaluated. Furthermore, the rules are iteratively refined, extended, discarded, newly implemented, or accepted.

The following example shows an IE rule to identify the *antecedent* and the *consequent* of a simple conditional sentence in English:

If a sentence contains the word 'if' and a 'comma':

- *split the sentence into two phrases based on the position of the comma,*
- *mark the phrase with the word 'if' as the antecedent, and*

- *the other as the consequent.*

The rule-based approach – also called the Knowledge Engineering approach – is increasingly used to populate databases in order to enable the execution of structured queries over unstructured documents [LCC+10].

3.2.2 Supervised Machine Learning Approach

The goal to reduce the workload of knowledge engineers to create the rules by hand led to the development of statistically based systems which "learn" extraction rules by using machine learning algorithms [PY13]. The intention of these supervised learning approaches is to automatically generate a classification model (classifier) from given input instances – represented by selected features (attributes) and their predefined classes. The classifier is then used to predict the class of a new, unseen instance. In order to gain acceptable results, a large training corpus with manually classified instances is needed. The general process of applying a supervised machine learning to a real-world problem [Kot07] is shown in Figure 3.4.

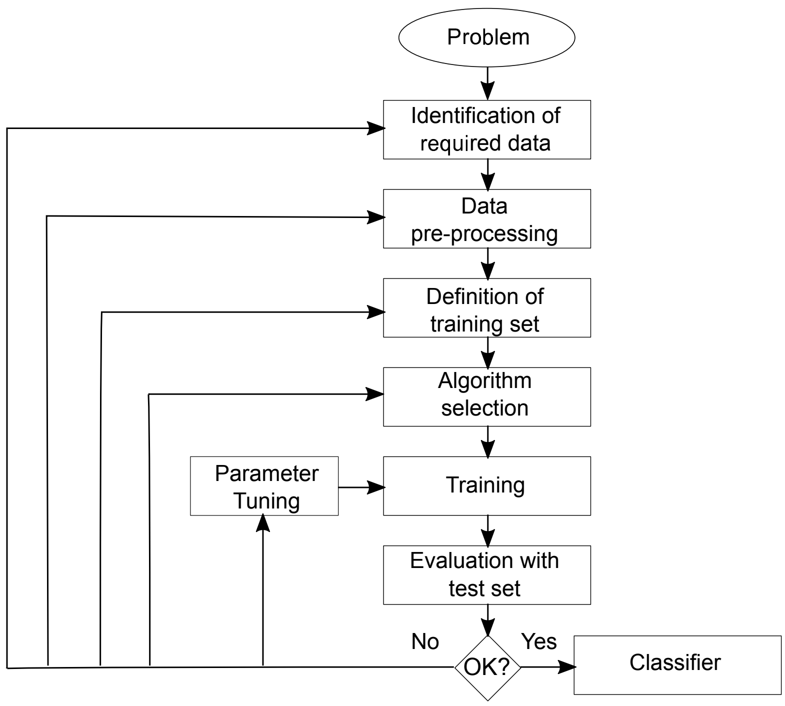


Figure 3.4: Process of supervised machine learning (taken from [Kot07]).

In the first step an expert defines the instances, their corresponding attributes, and their classes in the texts. If such an expert is not available, an alternative method called 'brute-force' can be applied. There, an arbitrary set of features, which hopefully includes the most significant features, is selected. In most cases relevant ones are missing or have no informative value, hence, a significant pre-processing is required. Irrelevant and redundant features have to be identified and the input data adapted accordingly [YL04]. New features are added by use of the feature construction/transformation technique [MR02]. Both methods aim at finding the ultimate features to create more concise and accurate classifiers.

The following methods for the definition of training sets – to evaluate the accuracy of different machine learning algorithms – are commonly used [Kot07]:

- The set is split, using two-thirds for training and the other third for testing.
- Cross-validation: The set is divided into mutually exclusive subsets of the same size. The performance rating is done by classifying every subset with a classifier trained on the union of the other subsets and then the average of the rates is calculated.
- Leave-one-out validation: This is a particular case of the cross-validation method where a subset contains only one single instance.

The next step is the selection of an appropriate machine learning algorithm (for a discussion about relevant available classifiers see [FDCBA14]). The classifier is trained with the selected training data and its evaluation based on the test data. If the accuracy of the mapping of unlabeled instances to classes is satisfactory, the appropriate algorithm for this problem is found. Otherwise, the previous steps have to be repeated in order to identify the reasons (e.g., wrong features, incomplete training data, etc.) for the poor performance to be improved.

3.3 General Frameworks for NLP Tasks

In recent years various NLP components, pipelines, and tools have been developed for different domains and application areas. In the next sections we describe general frameworks[6] solving natural language processing- and information extraction tasks.

3.3.1 Apache UIMA Project

The Unstructured Information Management Architecture (UIMA) defines interoperability standards for texts and multi-modal analytics [FL04, FLG$^+$06]. The Apache UIMA project implements this standard and supports the developers with UIMA frameworks, tools, and annotators to analyze different kinds of unstructured data, like text, audio, and video in order to discover, organize, and deliver relevant knowledge. The frameworks are released under the Apache 2 license and provide Eclipse plug-ins for the development of UIMA-based applications. Figure 3.5 shows the main components of the project.

The frameworks provide the environment for running the various components and are available for Java and C++. The scaleout frameworks support UIMA pipelines for high

[6]http://emerge.mc.vanderbilt.edu/natural-language-processing-nlp-survey-tools-resources (accessed March 2, 2015)

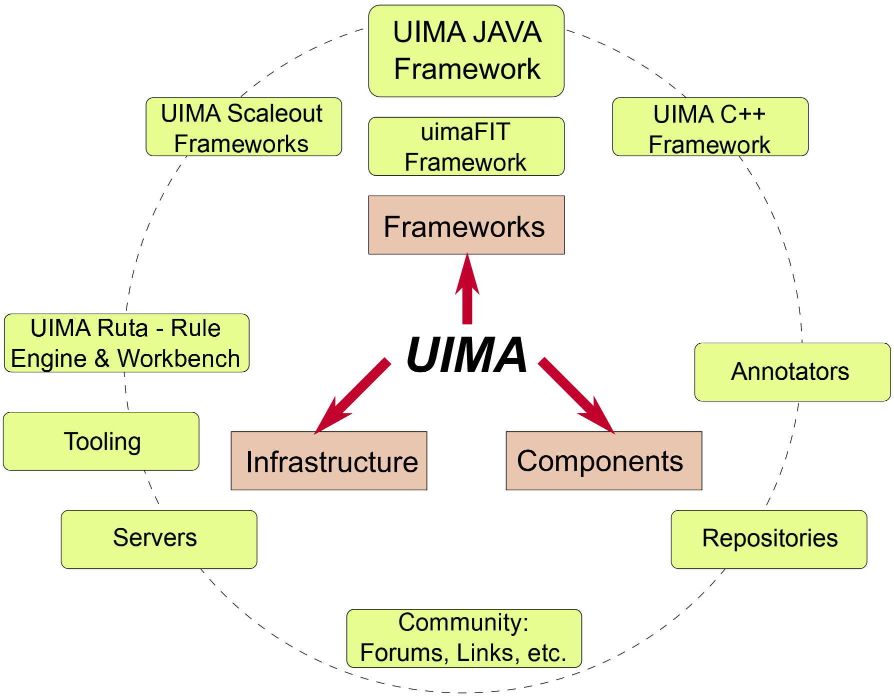


Figure 3.5: Components of the Apache UIMA project (taken from [Fou15b]).

throughput processing jobs and low latency real-time applications. The configuration of UIMA components is defined by means of Java annotations provided by the uimaFIT module.

The major task of the frameworks is to run pipelines of annotator components to analyze and annotate unstructured data. Such annotators are created newly and implemented due to special requirements or can be selected from various repositories (e.g., Apache cTAKES[7]).

The infrastructure component includes a simple server based on the REST[8] protocol to communicate with other web services to deliver annotation results. The UIMA Rule-based Text Annotation (Ruta) module comprises a workbench for developing rules and an interpreter for their execution. Comprehensive tools (e.g., for debugging, editing, packaging, etc.) supporting the development of applications are also available.

A large community of developers and users communicate via forums, mailing lists, and wiki tools to learn from each other, take care of bugs, and make contributions to new releases.

[7]http://ctakes.apache.org/ (last accessed March 16, 2015)

[8]http://www.ics.uci.edu/ fielding/pubs/dissertation/rest_arch_style.htm (last accessed March 16, 2015)

3.3.2 GATE

The General Architecture for Text Engineering (GATE) is a popular text analysis toolkit developed at the University of Sheffield. The GATE research program started in 1995 and has been continually extended to support software developers, language engineers and research staff of different domains. GATE has become one of the most widely used NLP systems in both academic and industrial projects [CTRB13]. The system includes a number of domain-independent rule-based NLP components (e.g., ANNIE: A Nearly-New Information Extraction System) and wrappers for other Java-based NLP components (e.g., WEKA[9], OpenNLP[10], etc.). An inter-operation layer provides an interface to run UIMA-based applications within GATE. Interactive tools help to annotate documents, to evaluate different test cases, and to define analysis rules, grammars, and expressions. GATE is published under the GNU open source licenses and is available for all major operating systems.

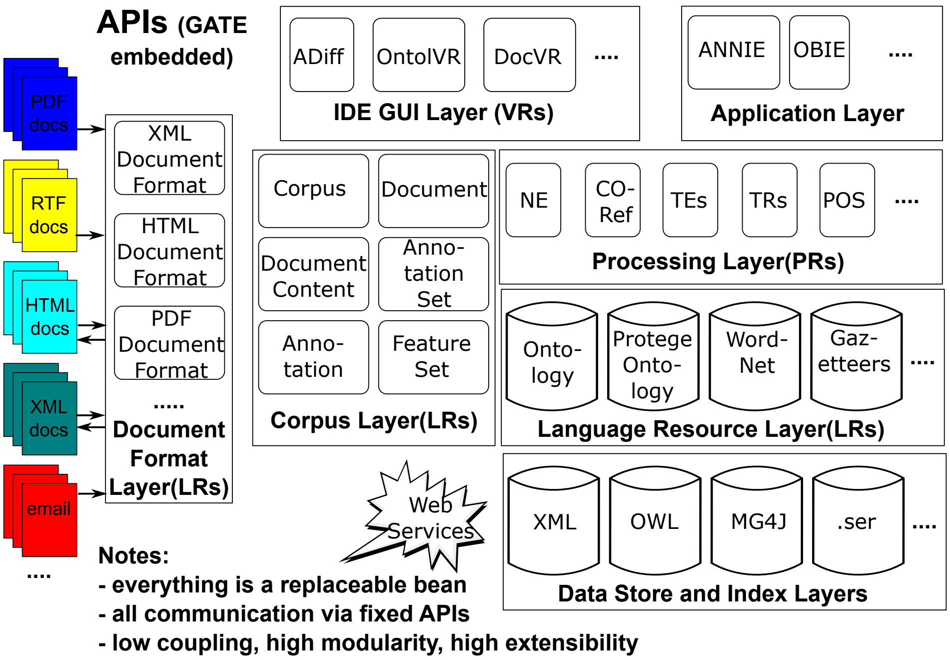


Figure 3.6: APIs of GATE embedded (taken from [CMB+11]).

The number of tools for the GATE project has grown over the years and comprises the following important products

- **GATE Developer** is an integrated development environment (IDE) that provides a graphical user interface to create, measure, and maintain software components for natural language processing.

[9]http://www.cs.waikato.ac.nz/ml/weka/ (last accessed March 16, 2015)
[10]https://opennlp.apache.org/ (last accessed March 16, 2015)

- **GATE Embedded** is an object-oriented framework which is implemented in Java and comprises the core modules of every GATE-based application. It contains a set of interlinked APIs (shown in Figure 3.6) based on a standard Java component model.

- **GATE Teamware** offers a collaborative annotation environment for distributed semantic annotation projects.

- **GATE Mímir** supports full-text search, concept search, and annotation structure search by means of only one index.

- **GATE Cloud** is a parallel, distributed processing engine for hosted large-scale text processing.

GATE Embedded (Figure 3.6) defines three different types of resources: (1) Language resources (LRs) holding linguistic data such as lexica, corpora, or ontologies, (2) processing resources (PRs) representing algorithms for data-processing (including the Java Annotation Pattern Engine JAPE, which provides finite state transduction over annotations based on regular expressions), and (3) visual resources (VRs) to build graphical interfaces. The modular structure and the APIs of the framework are the base of GATE's flexibility.

3.3.3 NLTK

The Natural Language Tool Kit (NLTK)[11] was developed at the Department of Computer and Information Science at the University of Pennsylvania and since then expanded continually.

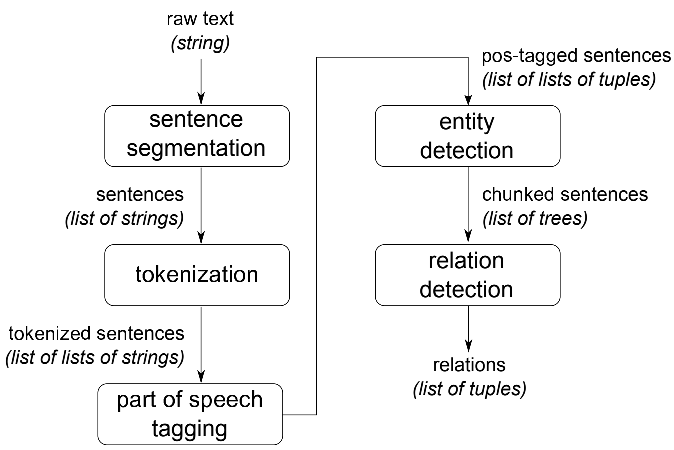


Figure 3.7: NLTK: Simple Pipeline Architecture for an Information Extraction System (taken from [BKL09]).

[11]http://www.nltk.org/ (last accessed March 16, 2015)

NLTK provides a library of different corpora and NLP tools – written in the Python programming language. NLTK offers standard functions (e.g., part-of-speech tagging, syntactic parsing, text classification, etc. – a typical pipeline is shown in Figure 3.7) and wrappers to solve common NLP tasks [Mad07].

The development of NLTK is based on the design principles: Simplicity, Consistency, Extensibility, and Modularity [BKL09]. It is a free, open source, community-driven project and available for the most common operating systems. It has been successfully used for teaching, as well as for prototyping and for building research systems. The latest version NLTK 3.0 was released in 2014.

3.4 IE Approaches for Modeling CIGs

Guideline developers edit CPGs in a free-text format. In order to transform the medical knowledge described in a guideline into execution models, a translation process is required. Moser and Miksch [MM05] detected prototypical patterns in free-text guidelines to bridge this gap. Serban et al. [StTvH$^+$07] proposed an ontology-driven extraction of linguistic patterns to pre-process a CPG in order to retrieve control knowledge. The evaluation showed that the modeling as well as the authoring process of guidelines were supported.

Language engineering methods were used in SeReMed [Den08] to detect diagnoses or procedures in medical documents. The method was successfully applied to X-ray reports. These documents, however, show a standardized structure and, therefore, are easier to handle by knowledge engineering methods than CPGs. Taboada et al. [TMRn$^+$10] identified relationships between diagnoses and therapy entities in free-text documents by matching the core information units of a sentence with a collection of predefined relationships, but the quality of this matching was not rated. To implement a rule-based approach to recognize medical entities the MeTAE (Medical Texts Annotation and Exploration) platform was used by Abacha and Zweigenbaum [AZ11]. Additionally, semantic relations between each pair of these entities were identified by means of MetaMap [AL10]. Consequently, relations between a problem (e.g., disease) and a corresponding treatment were found. The method was applied to selected articles and abstracts of PubMed, but not to CPGs.

In Kaiser et al. [KAM07] a heuristic-based approach using information extraction methods independent from the final guideline representation language was defined. This method was implemented and applied to several guidelines containing a high amount of semi-structured text. A set of semantic patterns representing activities based on semantic relations was generated by Kaiser et al. [KSM11] to identify medical activities in CPGs. Its effectiveness was proved by a study which showed that a large part of control flow related aspects could be identified. The relation between the activity and a corresponding condition, however, was not part of the method, but is an important requirement for the future automatic translation of a guideline.

Thorne et al. [TCM$^+$13] solved this problem by developing a supervised approach which identifies process fragments and determines the temporal relations among the medical activities. However, the detection of temporal relations is limited to simple before/after relations and, therefore, not appropriate to model the temporally complex, control-flow-related aspects of a guideline.

Semantic Web standards, like RDF, RDFS, and OWL, were used by Huang et al. [HtTvHAM14]. They developed a lightweight formalism of evidence-based clinical guidelines by means of XMedlan [AMBHR13], a Xerox NLP tool, to generate the semantic data of the guidelines – knowing that the OWL reasoning has its limits (e.g., reasoning about actions, description of uncertainty, temporal processing, etc.). The focus was set on the semantic interoperability and the formal representation of different levels of evidence.

This chapter discussed the basics and recent developments in information extraction. Two different approaches for the design of IE systems – the rule-based approach and the machine learning approach – were compared and general tools implementing these concepts were presented. In the final section we showed the numerous applications of IE algorithms for modeling clinical practice guidelines.

CHAPTER 4

Related Work in Modeling of Temporal Concepts with TimeML

The markup language TimeML is a specification formalism for annotating events and temporal relations in narrative texts and is widely spread in the natural language processing community. The fact that a revised and interoperable version of TimeML called ISO-TimeML [PLBR10, Ass08] was published in 2013 as an international standard for temporal annotations by ISO [Ass13], demonstrates its importance.

4.1 Temporal Concepts of TimeML

TimeML was developed in the project 'Time and Event Recognition for Question Answering Systems' (TERQAS[1]) and then applied to the TimeBank corpus containing 186 news articles. In the following years it has been extended and also transferred to other domains. It focuses on [PCnI+03]

- the identification of temporally anchored events,
- their ordering by means of temporal reasoning,
- the dealing with vaguely specified temporal expressions, and
- the reasoning about the duration of an event.

Generally, TimeML defines temporal concepts (e.g., *EVENTs*, *TIMEX*, etc.) and categorizes the relations (called *Links*) among them. The following list illustrates some simple examples to show these dependencies (***EVENTs*** in bold; <u>*TIMEX*</u> underlined).

- Time anchoring: *Joe* ***left*** *on* <u>*Friday*</u>*.*

[1] http://www.timeml.org/site/terqas/index.html (last accessed March 16, 2015)

- Event orderings: *The group* ***met*** *after midnight.*
- Embedded ordering: *Joe* ***said*** *Jack* ***visited*** *a doctor.*

In order to annotate TimeML concepts in a document, a particular XML-based syntax was recommended by Schilder et al. [Kat05].

EVENTs

The general definition of an *EVENT* is described in the TimeML annotation guideline as *".. a cover term for situations that happen or occur"* [SLG+06, p.2]. The XML-tag for an *EVENT* defines attributes to specify the characteristics of an *EVENT*. The most important one is the 'class' attribute. It distinguishes 'Occurence' (e.g., die, crash, build, merge), 'State' (e.g., on board, kidnapped, love), 'Reporting' (e.g., say, report, announce), 'Intentional Action' (e.g., attempt, try, promise, offer), 'Intentional State' (e.g., believe, intend, want), 'Aspectual' (e.g., begin, finish, stop, continue), and 'Perception' (e.g., see, hear, watch, feel) [PCnI+03]. Additionally, attributes for 'tense', 'aspect', 'polarity', and 'modality' have to be defined in order to enable temporal reasoning mechanisms.

TIMEX

The *TIMEX* expressions are primarily used to represent explicit temporal expressions (e.g., times, dates). The original formats *TIMEX* [Set01] and *TIMEX2* [FMSW01] have been extended due to TimeML's specific needs. The new format is specified as *TIMEX3*[2].

TIMEX expressions are grouped into fully specified temporal expressions (e.g., May 6th 1964; winter, 2015; 12 o'clock), underspecified temporal expressions (e.g., Tuesday, next week, three days ago), durations (e.g., three days, two months) and sets (e.g., every year, each day) [Pus12]. The value attribute is represented according to the ISO 8601[3] standard (e.g., 'PT6H' describes a 6 hour duration).

LINKS

The relation between *EVENTs* and/or between *EVENTs* and *TIMEX* expressions is defined as a *Link*[4]. TimeML differentiates three types:

- *TLink* defines a temporal relation in order to build up a chronology of events (e.g., to show a sequence of consecutive tasks) and implements all 13 temporal relations (e.g., simultaneous, before, after, etc.) defined by Allen [All83].

  Example:
  Jack *<EVENT id="e1">* learned *</EVENT>* for *<TIMEX3 id="t1"*

[2] Generally, we use the term TIMEX for the TIMEX3 tag of TimeML
[3] ISO 8601: http://www.iso.org/iso/home/standards/iso8601.htm (last accessed March 12, 2015)
[4] Some *EVENT* annotations in the examples are omitted for a better readability.

type="DURATION" value="P1H"> one hour *</TIMEX3>*.
<TLINK eventID="e1" relatedToTime="t1" relType="SIMULTANEOUS"/>

- *SLink* defines a subordination relation between *EVENTs*. They can be of type 'modal', 'factive', 'counter-factive', 'evidential', 'negative evidential', and 'conditional'.

  Example:
  Joe *<EVENT id="e1">* wants *</EVENT>* to *<EVENT id="e2">* drive *</EVENT>* home.
  <SLINK eventInstanceID="e1" subordinatedEvent="e2" relType="modal"/>

- *ALink* defines an aspectual relation, showing the progression or phases of an event (start, finish, etc.).

  Example:
  Mary *<EVENT id="e1">* started *</EVENT>* to *<EVENT id="e2">* work *</EVENT>* on task A.
  <ALINK eventInstanceID="e1" relatedToEvent="e2" relType="INITIATES"/>

TLink and *ALink* relations were often discussed and applied in projects dealing with temporal reasoning, whereas *SLinks* were hardly investigated. As they are found in conditional sentences, they play an important role for describing condition-based clinical activities in guidelines and, therefore, are further discussed in Chapter 8.

Time Anchoring

Every *EVENT* and *TIMEX* in a document are related to a temporal point. As TimeML was developed in the news-wire domain, the document creation time (DCT) acted as temporal point. If TimeML is applied to documents of any kind, this concept has its limits (e.g., the temporal linking of a medical activity in a clinical guideline to the DCT of the guideline is useless).

Therefore, Pustejowsky et al. [PS11] introduced the concepts of 'narrative time' and 'narrative container'. The 'narrative time' describes the current temporal anchor for *EVENTs* and changes during the reading process. This concept leads to fewer temporal links without losing temporal information. The 'narrative container' describes the interval of time between the earliest *EVENT* in the document and the one that is farthest in the future.

4.2 TimeML in the Medical Domain

The main challenges of text-based temporal reasoning in clinical texts are (1) the representation of temporal concepts, (2) their complexity (e.g., underspecified temporal relations, vagueness of tense and aspect, relative times), and (3) the linguistic style of the documents, which is fairly different to domain independent English texts [SRU13].

Independently from TimeML, the projects “TimeText”[ZPH08] (describing a temporal reasoning system designed to represent, extract, and reason about temporal information), “CLEF” [RGH+07] (providing a semantically annotated corpus to assist the development of IE methods), and “ConText”[CCD07] (identifying contextual features to categorize the condition of a patient) dealt with these challenges.

The first attempt to automatically discover temporal relations based on TimeML in clinical narratives, was executed by Savova et al. [SBS+09]. They transferred standard methods from NLP to the clinical domain and developed an annotation scheme based on TimeML in order to discover the timeline in clinical narratives. The temporal links (*TLink* and *ALink*) provided the necessary information for this task, whereas reasoning mechanisms over causal relations (described by *SLinks*) were not implemented.

UzZaman et al. [UA10a] implemented the TRIOS/TRIPS system containing a semantic parser to extract events, their linguistic features, and relations based on TimeML. The authors showed the flexibility of the specification language by extending the *SLink* attributes and introduced a new relation link called *RLink* to represent semantic roles. The TRIPS parser showed an F1 value of 69% when applied to the TimeBank corpus. Due to its domain independent implementation, it was evaluated on two medical text documents (patient reports) showing similar results (F1: 70%) [UA10b].

The most extensive project dealing with temporal reasoning in medical texts by means of TimeML concepts is the “THYME”[5] project. The first phase lasted from 2010 until 2014. It aimed at (1) developing a temporal relation annotation scheme and guidelines for clinical free texts, (2) creating an annotated corpus of more than 500k words of clinical narratives, (3) carrying out a descriptive study to compare the special needs of describing temporal relations in the medical domain in contrast to general domains, (4) developing new algorithms for the identification of temporal relations in the clinical domain, and (5) integrating the best methods and algorithms into the Apache cTAKES software [SMO+10]. The second phase[6] is scheduled from 2015 to 2020.

The first version of the annotation guideline was published for use in the ‘2012 i2b2 Clinical Temporal Relations Challenge’. It described in detail how to annotate the different temporal concepts of TimeML in the medical domain (e.g., the definition for the *EVENT* annotation was extended to *“.. anything that is relevant to the clinical timeline”* [SBF+14, p.145]). The latest version was published in 2014 based on the ISO TimeML specification. The annotated corpus can be downloaded from the project’s website and the Apache cTAKES tool is discussed later in Chapter 4.3.5.

The concept of ‘narrative containers’ in clinical texts was investigated by Miller et al. [MBD+13] as part of the THYME project. They examined sentences of clinical texts containing *EVENTs* and *TIMEX* in order to identify their ‘narrative containers’. They set the focus on *EVENT* – *EVENT* relations of type “CONTAINS” and developed a machine learning approach based on support vector machines with tree kernels to identify such sentences automatically. Despite of the fact that the ‘narrative containers’ were discovered in only a very specific field of the problem, Miller et al. showed that

[5] https://clear.colorado.edu/TemporalWiki (last accessed March 16, 2015)

[6] under review at the NIH (National Institutes of Health)

their approach outperformed rule-based methods. Furthermore, they confirmed that their approach led to more containers than the method proposed by Raghavan et al. [RFLL12] who defined only "coarse" temporal bins in relation to the admission date of a patient ('before admission', 'on admission', and 'after admission').

4.3 Tools to Generate TimeML Annotations

The Temporal Awareness and Reasoning Systems for Question Interpretation (TARSQI Toolkit – TTK) was one of the first implementations which generated TimeML-compliant annotations in order to enable temporally based questions about events in news articles [VP08]. The latest version of the toolkit concentrated – among other extensions – on the application to the medical domain and the introduction of narrative containers. Its implementation is still in progress [VP12]. In MED-TTK – a further extension of the toolkit – the TTK's time tagger was modified in order to improve the identification methods of temporal references in medical notes [ROM+13].

Gooch [Goo12a] used external resources (e.g., the UMLS – the Unified Medical Language System [LHM93]) in order to categorize selected medical concepts as events to be formalised in TimeML expressions. The solution was finally evaluated in a corpus of clinical discharge summaries.

In "TempEval 2013" – a competition in which researchers compare their temporal information extraction methods – the ClearTK-TimeML [Bet13] competed in all English tasks and succeeded in three different categories. It combines a pipeline of machine-learning models built upon the ClearTK framework [OWB08] in order to identify the different temporal concepts of TimeML.

The clinical Text Analysis and Knowledge Extraction System "cTAKES" was developed as part of the THYME project in 2010 [SMO+10]. It focused on the information extraction from clinical free-text medical records by using open-source natural language processing tools. For the i2b2 Natural Language Processing Challenge 2012[7] it was extended to "icTAKES" with the MedTagger tool to improve the identification of time expressions by means of a rule-based approach [SWL+13].

In the following sections we present the discussed tools in detail.

4.3.1 TARSQI Toolkit

The TARSQI Toolkit (TTK) combined different information extraction methods to identify temporal information from natural language texts. Its first version is still available at the TimeML website [8]. The current version (as described in [VP12]) has not been published yet[9], but contains extensions regarding to the medical domain and the TimeML concept of 'narrative containers'. Figure 4.1 shows the architecture of the toolkit.

[7] https://www.i2b2.org/NLP/TemporalRelations/ (last accessed March 16, 2015)
[8] http://timeml.org (last accessed March 16, 2015)
[9] Information from Marc Verhagen 30/10/2013 and still not published

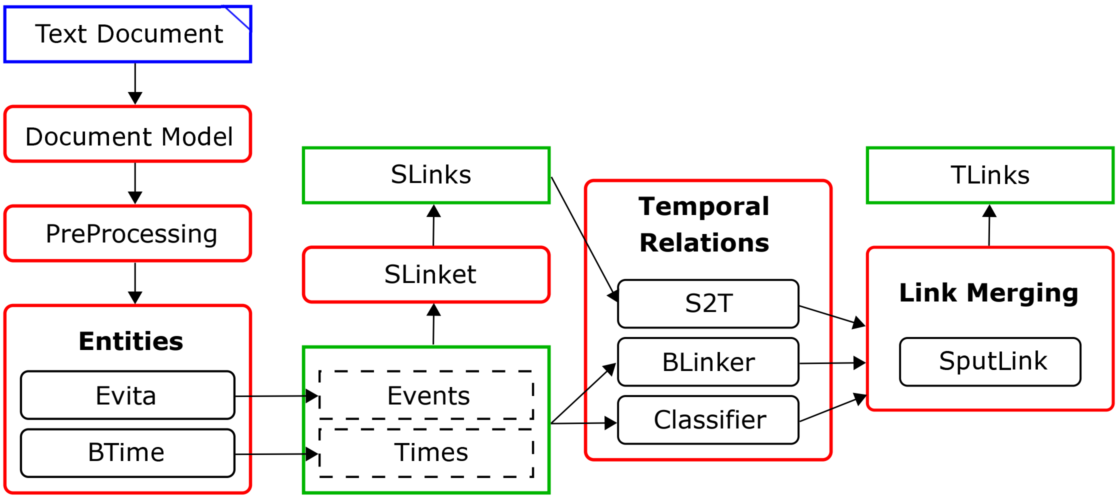


Figure 4.1: The TARSQI-Toolkit Architecture (taken from [VP12]) – the produced TimeML concepts in green frames and the processing components in red frames.

The text document is handled by the 'Document Model' in order to enable the processing of different document types with their meta tags and language encodings. The resulting text is pre-processed by means of standard NLP tools (e.g., tokenizer, POS-tagger, chunker, etc.). The next step focuses on the identification of the temporal concepts *EVENTs* and *TIMEX*.

The *EVENT* recognition is implemented in the *EVITA* component. It works domain-independently and identifies *EVENT*s and their grammatical features, like aspects and tenses. As the latest version of the TTK was exploratively applied to clinical discharge summaries, its functions were extended [SH10]. Firstly, medications were included in the concept of events and, secondly, the time anchors in the headers of the discharge summaries were processed.

The *BTime* component is responsible for the extraction of time expressions (by using the grammar of 82 context-free rules), the completion of underspecified *TIMEX* values, and their normalization. The latter task is done by heuristic rules and 'anchors' the time expressions to a temporal reference.

SLinks annotations are the result of the *Slinket* component and can be lexically-based or structurally-based [SLG+06]. Lexically-based ones are produced by means of a pattern library of verbal and nominal predicates, like *regret*, *say*, *promise*, etc. Structurally-based ones are created for purpose clauses and conditional constructions. However, the *SLinket* component identifies only *SLinks* of the first type [SVP06].

In the next stage *TLinks* are generated based on the *EVENTs*, *TIMEX*, and *SLinks*. The 'MaxEnt' classifier identifies temporal relations between events and times, and the *Blinker* module applies rules to find syntactic patterns. The *S2T* component maps certain kinds of *SLinks* (e.g., subordination links, in which both events are specified in the past tense and the dominating verb is of type 'reporting') onto *TLinks*.

A constraint propagation algorithm – implemented in *Sputlink* – compares the *TLinks* and checks the consistency of these links in order to select the best one based on its precision.

The TTK in the described version has not yet been compared to other toolkits in this domain, hence, an interpretation of its performance cannot be given.

4.3.2 Med-TTK

The identification of temporal relations relies on the quality of *EVENTs* and *TIMEX* annotations. The "Med-TTK"[10][ROM+13] was the first attempt to apply the TTK to the medical domain, specifically, to veterans' affairs clinical notes to improve the identification of medical temporal concepts. As the temporal terminology in medical texts differed from news in many ways, the algorithms of the TTK's time tagger had to be extended. The methods for the identification of time granularity and frequency were enhanced. Additionally, more flexible date-time formats were supported. Table 4.1 shows selected examples of these enhancements.

TIMEX	TTK examples	Med-TTK extensions	Med-TTK examples
time	14:35	extended notation	15:05:023
	the night	portions of the day (am, pm, morning,..)	morning of 5/12
	later this afternoon	word numbers with units of time	two hours, 3 minutes ago
	noon Thursday	smaller granularity and abbreviations	8ms 2ms 13:52:37.031
	08-16-90-2041EDT	time-date single entry format	January 10,2003@09:44:07
duration	two-hour (meeting)	unit abbreviations (min, h, yr,..)	15y/o 23 min
	nearly forty years	range durations	6-8 weeks

Table 4.1: Examples for Med-TTK enhancements (adapted from [ROM+13]).

Med-TTK performed significantly better than the TTK. In comparison to the initial version of TTK (recall value 14%, precision value 27%) it reached a recall value of 86% and a precision value of 85% for the identification of temporal references [VP08].

4.3.3 A Pattern-Based Approach by P. Gooch

Phil Gooch tried to define *EVENTs* from a clinical perspective. Therefore, he classified clinical concepts (e.g., 'diabetes mellitus type 1'), a verb or a verb group defining a process (e.g., 'should be referred'), and concept or process modifiers (e.g., 'decreasing')

[10]http://www.code.google.com/p/med-ttk (last accessed March 16, 2015)

as *EVENTs* [Goo12a]. Additionally, he found specific patterns for the identification of temporal references (e.g., 'for at least three days') to generate *TIMEX* annotations. Informatics for Integrating Biology & the Bedside (i2b2[11]) provided a corpus of 190 manually annotated discharge summaries building the source of the pattern identification task.

The identified patterns were implemented as resources in the General Architecture for Text Engineering (GATE) framework [CTRB13] (details of GATE are discussed in Chapter 3.3). The standard NLP features of the framework were used for tokenization, POS-tagging, etc. The specific rules for the pattern recognition were implemented by means of the Java Annotation Pattern Engine (JAPE) included in GATE. The Unified Medical Language System (UMLS) acted as external knowledge source. It helped to identify selected clinical concepts and their semantic types to be marked as *EVENT* [MBB01].

The evaluation of 120 discharge summaries showed a precision of 82% and a recall of 62% for the identification of *EVENTs* and their attributes 'negation' and 'possibility'. The error analysis showed different error types. One of them was the distinction between relative dates and durations, illustrated by the following example: For the phrase *'the three days prior to admission'* the words *[the three days]* were classified in the gold standard as 'duration' whereas the tool classified *[three days prior to admission]* as a 'date'.

Nevertheless, Gooch showed in his approach that the identification of TimeML concepts by means of a pattern-based approach was feasible.

4.3.4 ClearTK-TimeML

The ClearTK-TimeML was built on top of the ClearTK framework for machine learning, developed at the Center for Computational Language and Education Research (CLEAR) [OWB08] and was aimed at succeeding in the TempEval 2013 [12] tasks by applying machine learning classification methods. The features for these methods were derived from either tokens, POS-tags or concepts of a syntactic constituency parser. Three models – for *TIMEX*, for *EVENTs* and for temporal relations (*TLinks*) – were elaborated. The first two models were defined as a BIO token-chunking task, labeling each token as being the beginning of (B), the inside of (I), or entirely outside (O) of a span of interest. The temporal relation identification was implemented by a multi-class classification approach where a pair of *EVENTs* or a combination of an *EVENT* and a *TIMEX* were used to predict their temporal relations (e.g., before, after, etc.) [BMK07].

The implemented annotators for identifying and categorizing events, time expressions and temporal relations are listed in Table 4.2.

The tools Mallet[13], LIBLINEAR [14] and OpenNLP[15] provided the appropriate meth-

[11]https://www.i2b2.org/ (last accessed March 16, 2015)
[12]http://www.cs.york.ac.uk/semeval-2013/ (last accessed March 16, 2015)
[13]http://mallet.cs.umass.edu (last accessed March 16, 2015)
[14]http://www.csie.ntu.edu.tw/ cjlin/liblinear (last accessed March 16, 2015)
[15]http://opennlp.apache.org (last accessed March 16, 2015)

Annotator	Description
TimeAnn	identifies time expressions and adds *TIMEX* annotations.
TimeTypeAnn	sets the classes of *TIMEX* to DATE, TIME, etc.
EventAnn	identifies event expressions and adds *EVENT* annotations.
EventTenseAnn	sets the tense attribute of *EVENTs* to PAST, etc.
EventAspectAnn	sets the aspect of *EVENTs* to PERFECTIVE, etc.
EventClassAnn	sets the class of *EVENTs* to OCCURRENCE, etc.
EventPolarityAnn	sets the polarity of *EVENTs* to NEG or POS.
EventModalityAnn	sets the modality of *EVENTs* to will, should, must, etc.
TemporalLinkEventTo-DCTimeAnn	identifies temporal relations between *EVENTs* and the DCT and adds corresponding *TLinks*.
TemporalLinkEventTo-SameSentenceTimeAnn	identifies temporal relations between *EVENT* and *TIMEX* in the same sentence and adds *TLinks*.
TemporalLinkEventTo-SubordinatedEventAnn.	identifies temporal relations between syntactically dominated *EVENTs* and adds *TLinks*.

Table 4.2: Implemented annotators for the TimeML module of the ClearTK framework (adapted from[OWB08]).

ods for the machine learning tasks.

The ClearTK-TimeML tool was titled as "a minimalist approach" [Bet13], but reached the best scores (F1) for 'temporal relation extraction', 'time extent strict' and 'event tense accuracy' at the TempEval 2013, but the test-corpus did not contain medical texts.

4.3.5 Apache cTakes

The clinical Text Analysis and Knowledge Extraction System (cTAKES) was developed as part of the THYME project. The goal was *"...to develop a large-scale, comprehensive, modular, extensible, robust, open-source NLP system"* [SMO+10, p.507] to support the clinical research domain. The cTAKES was built on the Unstructured Information Management Architecture (UIMA)[16] and the OpenNLP natural language processing toolkit[17]. Both packages were part of the the Apache Software Foundation[18] initiative, consequently, cTAKES was also released under an Apache License (V2.0). The corpus on which the system was trained and tested by means of standard metrics was derived from clinical notes of the Mayo Clinic EMR[19]. Figure 4.2 shows an overview of the system's architecture.

[16]https://uima.apache.org/ (last accessed March 16, 2015)
[17]https://opennlp.apache.org/ (last accessed March 16, 2015)
[18]http://www.apache.org/ (last accessed March 16, 2015)
[19]http://www.mayoclinic.org/about-mayo-clinic/electronic-medical-record (last accessed March 16, 2015)

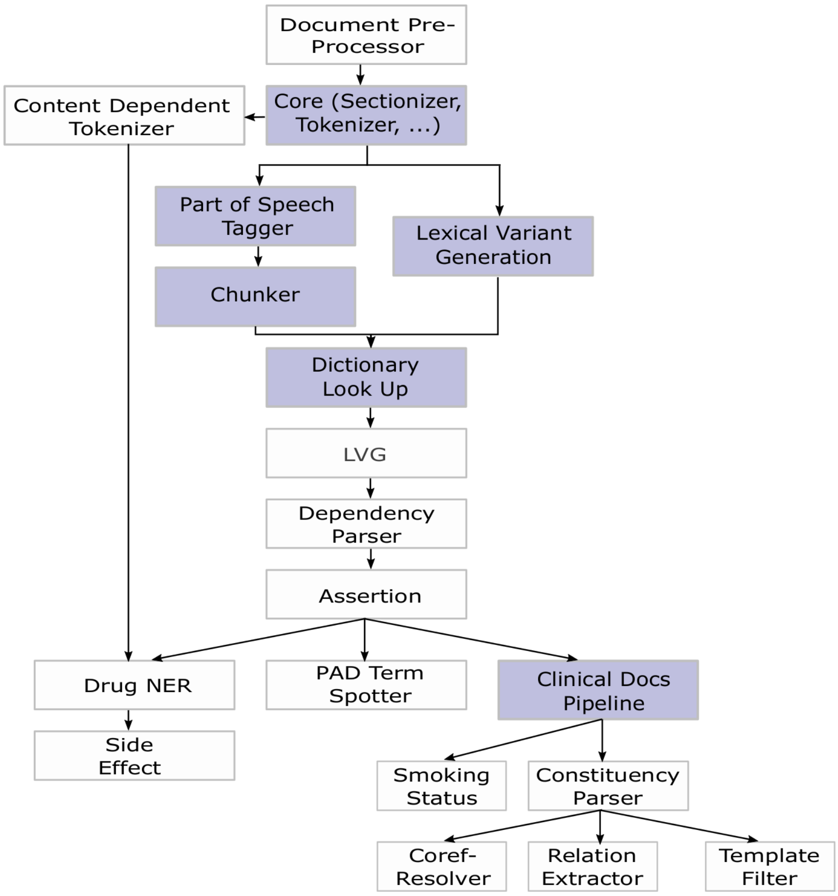


Figure 4.2: Components of Apache cTakes (taken from [Fou15a]) – core components in blue color

The components of cTAKES are arranged in a pipeline of optional and required components, each containing one or more analysis engines to generate the corresponding annotations. The system provides interfaces for plain text notes as well as for notes formatted conform to the Clinical Document Architecture (CDA) [DAB+06] which requires the 'Document Preprocessor' component.

The 'Core' module contains standard NLP tools, like sectionizer, tokenizer, etc., and provides the input for the components 'Content Dependent Tokenizer', 'POS-Tagger' and 'Chunker', and the 'Lexical Variant Generation (LVG)[20]' – a package of utilities that generates, transforms, and filters lexical variants from the output of the 'CORE'. The 'Chunker' and the 'LVG' components provide the input for the 'Dictionary Lookup' until, eventually, the produced data is analyzed by the 'Dependency Parser' to fulfill the semantic role labeling task. The 'Assertion' module classifies an event or a named entity

[20]http://lexsrv2.nlm.nih.gov/LexSysGroup/Projects/lvg/2012/web/index.html (last accessed March 16, 2015)

in regard to be negated, uncertain, or conditional.

Depending on the used pipeline, the output can be processed by (1) the 'Drug Named Entity Recognition' followed optionally by the 'Side Effect' component, extracting physician-asserted drug side effects of the input data, by (2) the 'Peripheral Artery Disease (PAD) Term spotter', which extracts radiology notes concerning diagnosis, treatment, etc. and classifies each document accordingly, or by (3) the 'Clinical Documents Pipeline'. The latter combines general purpose IE modules ('Constituency Parser'[21], 'Coref-Resolver', 'Relation Extractor', and 'Template Filter') and the 'Smoking Status' module, which groups patients' medical records according to the categories 'past smoker', 'current smoker', 'smoker', 'nonsmoker', and 'unknown'.

The cTAKES was applied to EMRs of different medical areas (ascertaining cardiovascular risk factors, and pharmacogenomics breast cancer treatment). Compared to an expert-developed gold standard, the system showed F-scores between 70% and 83% [SMO+10] for the different tasks. During the i2b2 NLP challenge the system was also applied to electronic data of other clinical institutions (e.g., patient smoking status from medical discharge summaries) to show its portability. For detailed results see [UGLK08] and [Uzu09].

In this chapter we discussed the main concepts of the TimeML specification language and their applicability to describe the temporal concepts in clinical texts. Selected tools for generating TimeML annotations closed this chapter.

[21] A wrapper around the OpenNLP parser: https://opennlp.apache.org/ (last accessed March 16, 2015)

Part II

GOALS

Part II continues with the introduction of the GOALS methodology, its underlying concepts, its description, and a corresponding evaluation scheme. The next chapters are arranged according to the steps of GOALS (see Figure 5.2). At first we discuss the steps dealing with information extraction methods followed by a scenario-based application of the whole methodology.

CHAPTER 5

The GOALS-Methodology

GOALS defines a step by step process to model a guideline – annotated with TimeML – independent from the target language (e.g., Asbru). As TimeML represents temporal concepts (e.g., *EVENTs, TIMEXs, etc.* – see Chapter 4.1), only information dimensions of a guideline containing such concepts are focused on. Following the steps of GOALS, parts of a guideline's CIG model are (semi-) automatically generated. Consequently, the modeling process can be sped up while simultaneously increasing quality.

5.1 Temporal Concepts of TimeML in CPGs

As temporal aspects have great significance within clinical guidelines [TGS08] (e.g., the description of care-paths), temporal reasoning methods may support the automatic modeling process of a guideline. These methods are based on the temporal representation of a document consisting of temporal expressions, concept primitives, and temporal relations in order to handle vague and/or complex temporal dimensions. TimeML fulfills these requirements, and has already been successfully applied to medical texts such as clinical narratives and discharge summaries. However, existing research results can only partially be adopted, because clinical guidelines differ in many ways. Clinical narratives, for example, are full of abbreviations, contain explicit time information (e.g., laboratory tests, doctor's visits), and represent the patients's progression of illness, to name but a few [SRU13]. On the contrary, the language in guidelines is highly sophisticated, time and date information is only implicitly known (e.g., first trimester of pregnancy), and – moreover – often expressed vaguely (see Table 5.1). Despite these differences, TimeML is a possible solution to annotate the various temporal information aspects of clinical guidelines, as it includes Allen's algebra of intervals [All83], which is also implemented in CIG languages such as Asbru.

Aspects	Clinical Narratives	Clinical Guidelines
Patient	individual	categorised group (e.g., children, adults)
Language	sentence fragments, abbreviations, special terms	highly sophisticated, complex sentences, grammatically and stylistically mature
Chronology	absolute time stamps (e.g., laboratory tests, doctor's visits)	relative time specifications (e.g., after admission, post-dinner)

Table 5.1: Linguistic differences in clinical texts.

An *EVENT* in TimeML (as already discussed in Chapter 4.1) is described as *".. a cover term for situations that happen or occur"* [SLG+06, p.2] and was redefined for an i2b2 challenge in 2012 to *".. anything that is relevant to the clinical timeline"* [SBF+14, p.145]. The last statement shows a certain fuzziness in the definition, nevertheless, we will adopt it to clinical guidelines.

TIMEX expressions are primarily used to represent explicit temporal expressions (e.g., times, dates). Although clinical guidelines hardly contain such time stamps, *TIMEX* expressions are used to represent durations, frequencies, etc.

The relation between *EVENTs* and/or between *EVENTs* and *TIMEX* expressions is defined as a link. *TLinks* are used to describe temporal dependencies (e.g., to show a sequence of consecutive tasks). Condition-based clinical activities are often specified in conditional sentences. The subordination relation in these sentences is expressed via an *SLink*. *ALinks* define the progression of an event (start, finish, etc.), and are, therefore, often found in clinical guidelines.

Compliant with the original specification of TimeML, every temporal link references to the document creation time (DCT). As the DCT does not contain important information for clinical guidelines, we use the concept of *narrative containers* introduced in [MBD+13] and [PS11]. It describes the current temporal anchor for events in a guideline and it changes during the reading process. This concept leads to fewer temporal links without losing temporal information.

In Figure 5.1 the *narrative containers* for the clinical protocol *Management of active low-risk labour - Admission for Birth* [Rem10] are shown. In this case, the document structure has a major influence on the amount of *narrative containers*.

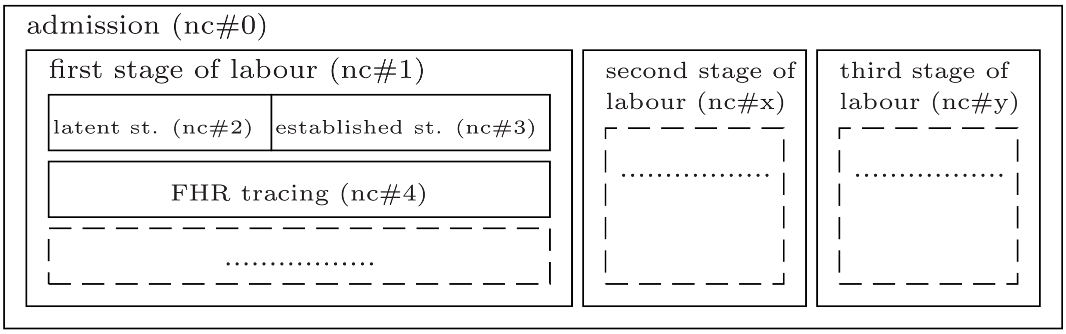


Figure 5.1: The narrative containers (nc#*) within the protocol.

5.2 The Multi-Step GOALS-Methodology

CIG languages, in general, are formal languages with a defined syntax and defined semantics. From the information extraction point of view we can interpret them as templates with information slots to fill. Consequently, in Step (1) of our methodology (see Figure 5.2) the mapping of time related concepts (*TLinks*, *SLinks*, and *ALinks*) to templates of the target language has to be defined and the information slots identified.

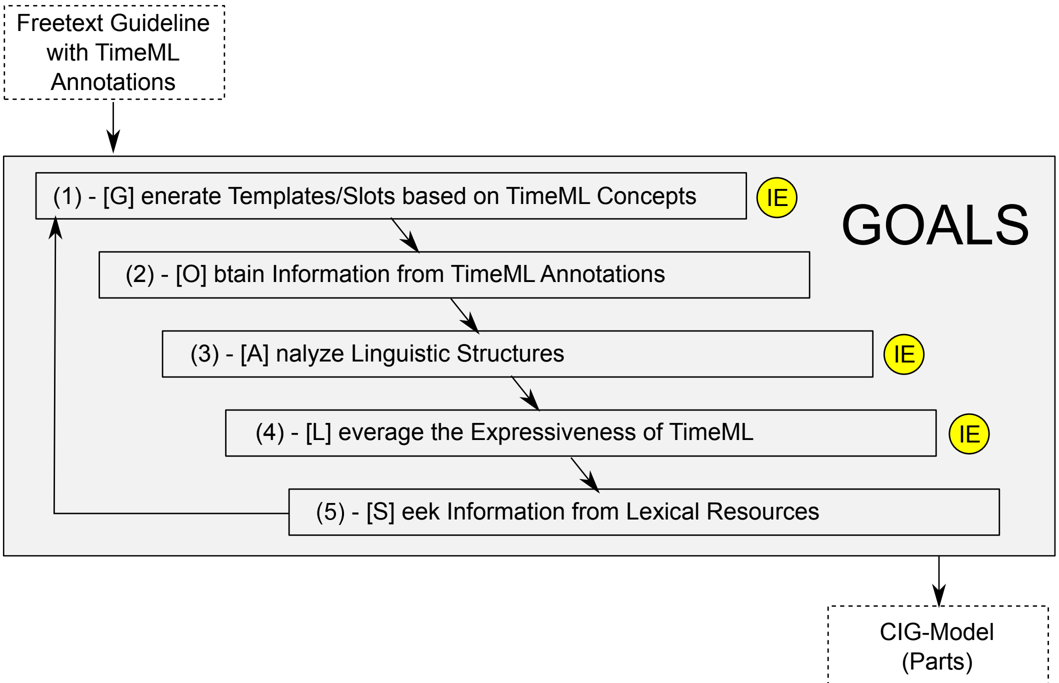


Figure 5.2: The GOALS-Methodology.

Step (2) deals with locating of information directly kept in the attributes of the TimeML annotations and the information which can be derived through transitive chains of TimeML links.

Step (3) linguistically preprocesses the original text (e.g., sentence splitting, co-reference resolution, identification of adverbial phrases, etc.) depending on the selected target language and the information needs.

In Step (4), empty slots or incorrect values in the slots have to be evaluated. Furthermore, we have to check if an extension to the TimeML specification can compensate those deficits. In such a case, we extend the TimeML specification and adapt the annotations accordingly. Otherwise, missing information must be sought from lexical resources (e.g., medical vocabularies) in Step (5).

If there are still any open slots, the process will have to be restarted at Step (1). This whole procedure is called the GOALS-methodology which is an acronym of the verbs defining the individual steps.

Finally, the outcome represents parts of a CIG model. Generally, every step of the process can lead to the generation of additional templates, which may also represent non-temporal aspects of a guideline. The yellow circles indicate that these steps are implementable by using information extraction methods (see Chapters 6, 7, and 8) while the other steps can be managed by simple mapping algorithms (see Chapter 9).

In order to prove the applicability of GOALS, we selected the intermediate CIG formalism MHB (see Chapter 2.2.1) as target language. MHB is less formal, but – anyway – contains all information dimensions for further translation into a formal language. Figure 5.3 illustrates the standard modeling process enriched by GOALS.

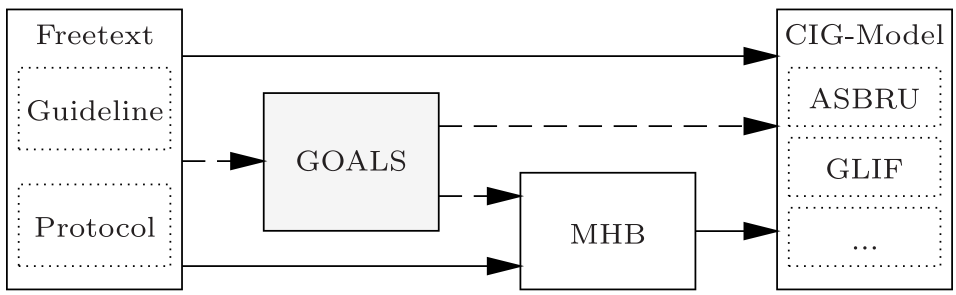


Figure 5.3: Standard modeling processes (solid arrows) - GOALS integrated processes (dashed arrows).

Supposing our methodology supports this process, the conclusion to apply it successfully to more complex tasks can be drawn.

5.3 Evaluation Scheme

The manual process of modeling a guideline is supported by various software tools such as the Document Exploration and Linking Tool with Add-ons DELT/A [VMK04]. This tool displays both the original guideline text and its corresponding (semi-)formal representation next to each other. The modeler marks a piece of the text in the original guideline and selects the appropriate structure (=template) of the target language, transfers the information manually into the slots, and adds medical knowledge where it is necessary. These modeling steps can be simplified by our GOALS methodology in many ways. Consequently, we define different levels of support which can also be used as a kind of evaluation scheme.

Level A: The templates and information slots of the CIG language are identified correctly.

Level B: The content of attributes of TimeML annotations to the corresponding slot of the target language is transferred accurately.

Level C: Linguistic processing of the analyzed text provides the appropriate phrases – semantically correct, but in different wording.

Level D: The information extraction methods deliver the phrases, words, or information entities in a standard notation format (e.g., *verb-object* notation for activities [MRvdA10]).

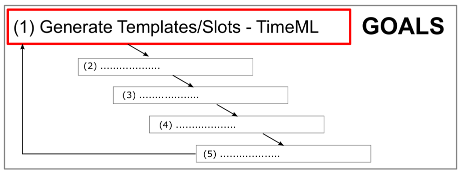
Level E: The templates are completely filled - no slot is left open and the result is completely consistent with the gold standard.

Every achieved level leads to a significant reduction of workload for the guideline modelers.

The next chapters show possibilities of how to automatise selected steps of GOALS by means of information extraction methods.

CHAPTER 6

TimeML Concepts and MHB Dimensions



The most prominent building blocks of CPGs are phrases describing clinical activities, the circumstances under which they are applied, their temporal context, and their clinical intention. In this chapter we describe a method to identify and to classify such building blocks by means of an information extraction approach acting on temporal concepts described by TimeML. This method supports the generation of information extraction templates, and – therefore – represents a realization of step #1 of GOALS.

The following example shows a sentence which should be identified by our method as relevant for the clinical care process and classified as a 'clinical activity' containing corresponding 'background information' (further examples are shown in Table 6.2).

Episiotomy should not be carried out routinely during spontaneous vaginal birth.

Based on this classification, the required IE templates can be generated depending on the target CIG language. As already discussed in Chapter 5.2, we have chosen the MHB ontology as target language which defines different information dimensions (e.g., 'control dimension', 'background dimension', 'time dimension', etc.) representing such IE templates (for detailed information about MHB see Chapter 2.2). MHB models are built on chunks of texts containing several sentences, but due to the 'narrative container'-concept of TimeML we focus on chunks with only one sentence. All the other dimensions of MHB (e.g., 'data', 'evidence', 'background', etc. – modeled as dimensions independently from control flow related aspects) are not handled, because (1) the existing guideline models do not provide sufficient training examples for our corpus, and (2) some dimensions (e.g., 'evidence') do not contain temporal concepts.

An analysis of sentences represented by their temporal concepts showed no distinctive recurring patterns to be useful for a rule-based IE approach. Consequently, we decided to solve this problem with machine learning algorithms.

6.1 Knowledge Sources and Tools

In our experiments we tried to combine as many mature software tools as possible in order to minimise the software development efforts.

6.1.1 Corpus

The selection of adequate training data for a machine learning approach is crucial to the quality of the expected results. Therefore, we assembled our corpus with MHB models of former projects (e.g., REMINE[1] and Mobiguide[2]). At the end, the resulting corpus consisted of 10 guidelines, respectively guideline chapters, from different medical domains (see Table 6.1) which were split into a training and a testing corpus. Consequently, both were mutually exclusive.

Guideline Topics	#	Corpus
Stroke Prevention and Educational Awareness Diffusion	260	test
Labour and Delivery Management	51	test
Procedure ER/DTP Stroke	11	train
MRSA Infection	23	train
Kauhajoki Hospital Acute Department 1	14	train
British Guideline on the Management of Asthma	46	train
Chronic Hypertension in Pregnancy	7	train
Type 2 Diabetes Mellitus in Adults	144	train
Hypertrophic Cardiomyopathy	56	train
Gestational Diabetes	7	train
Sum	619	

Table 6.1: Number of sentences per guideline used to build up the corpus.

All sentences of the corpus which were not modeled in MHB or represented other dimensions than the selected MHB dimensions were treated as negative examples.

[1] http://www.remine-project.eu/ (last accessed March 16, 2015)

[2] http://www.mobiguide-project.eu/ (last accessed March 16, 2015)

#	Annotated Examples
1	{When FHR tracing is reassuring at admission}$^{\text{CO}}$, {the woman should be allowed to move freely}$^{\text{AC}}$, {even if membranes are not intact.}$^{\text{BG}}$
2	{Women with pain but no cervical changes}$^{\text{CO}}$ {should be {re-examined after two hours}$^{\text{TI}}$.}$^{\text{AC}}$
3	Personnel that receives the alert {(as general rule the nurse at the "post triage" area)}$^{\text{BG}}$ will: {prealert the neurologist}$^{\text{AC}}$ {if the arriving time and the clinical status of the patient are available.}$^{\text{CO}}$
4	{Have the patient say a sentence.}$^{\text{AC}}$
5	{{Episiotomy}$^{\text{AC}}$ should not be carried out routinely during spontaneous vaginal birth.}$^{\text{BG}}$
6	{In case of abnormal FHR}$^{\text{CO}}$, {monitoring should be {continuous}$^{\text{TI}}$.}$^{\text{AC}}$
7	{{If analgesia is performed at full dilatation or within one hour to delivery}$^{\text{CO}}$, {meperidine could be administered}$^{\text{AC}}$ since fetal effects are minimal if maternal administration is done within 1 hour from delivery.}$^{\text{BG}}$

Table 6.2: Sentences containing annotations for the MHB control dimension (AC..activity, CO..condition), temporal dimension (TI), and background dimension (BG) – not every temporal concept (such as, 'one hour' or 'during') leads automatically to a modeled temporal dimension (TI) in MHB.

Table 6.2 shows a selection of sentences and their annotations corresponding to the appropriate MHB dimensions. Sentence #1 shows a condition-based activity including an explanation (labeled as 'BG'). The condition need not be expressed by a conditional clause, it can also be expressed as illustrated in sentence #2. Examples #4 and #5 are both describing clinical activities, the activity in the first one is described in a complete sentence, whereas the second one is described by a medical term only. A bigger complexity is evident in example #7 with a temporal statement (*"..within 1 hour from delivery.."*) included in the explanation, but not modeled as temporal dimension in MHB.

6.1.2 UMLS SN

The Unified Medical Language System (UMLS) [LHM93] combines selected health and biomedical vocabularies to facilitate the standardized exchange of medical data between computer systems. It offers three different components:

1. The Metathesaurus which is an aggregation of medical terms and codes of different vocabularies (e.g., MeSH, SNOMED CT, etc.).

2. The Semantic Network which reduces the complexity of the Metathesaurus by assigning semantic types to the concepts of the Metathesaurus in order to group and define relationships among them.

3. The SPECIALIST Lexicon and Lexical Tools which provide natural language processing tools.

Clinical concepts need to be recognized to create TimeML compliant *EVENT* annotations. Therefore, the UMLS (Methathesaurus + MetaMap) is used to identify and to classify such concepts using a more general categorization scheme (i.e., the UMLS SN).

6.1.3 Tools

The generation of TimeML annotations was the initial step of our machine learning task. Because of its promising results in the Temp-Eval 2013 task (it was ranked 1st for 'temporal relation F1', 'time extent strict F1' and 'event tense accuracy') [Bet13], we selected the "clearTK-TimeML" application for this challenge, although it was never applied to the medical domain. Hence, we supplemented the missing TimeML annotations by using the natural language processing tool GATE [CMB+11] with the tagger for MetaMap [AL10] to access the UMLS SN. The machine learning task was executed by means of the software tools "crf++"[3], which supported the conditional random fields' approach, and "SVM$^{\text{light}}$"[4], which implemented the support vector machine algorithm. The tools were arranged like a pipeline where the output of one application became the input of the next application, whereas the conversion from one data format into the next was implemented manually, as were the 'fuser'-component and the reporting software packages.

6.2 Method

The selection of the "optimal" classifier in text mining is widely discussed, because the success of supervised learning methods strongly depends on the amount of the available training data [MRS08]. Before selecting a specific machine learning application, we determined the type of our classification problem. As already discussed, one sentence may contain different MHB dimensions. In that case, binary classifiers – also called two-class classifiers – are not an appropriate solution. However, a multi-class problem can be split into multiple binary-class problems and the results then combined in multi-class scenarios [Joa02]. If the classes are mutually exclusive, the "one-of classification" approach (also called "multi-class") – if not "the any-of classification" (also known as "multi-label") is appropriate. On closer examination, our classification challenge can be treated as the first as well as the second type.

We clustered the sentences of the training corpus and found 8 different, mutual exclusive classes (see Table 6.3). Subsequently, every sentence was classified accordingly. The number of sentences in each class was strongly varying and no sentence was assigned to the inferred class 'CA_B_T'. Nevertheless, as every sentence exclusively belonged to one class, the prerequisites for a multi-class classification were met.

[3] CRF++ 0.58 – http://crfpp.googlecode.com (last accessed March 16, 2015)

[4] SVM$^{\text{light}}$ 6.02 – http://svmlight.joachims.org (last accessed March 16, 2015)

Class	Sentences describing/containing	#
CA	clinical activity	75
CA_B	clinical activity and background information	2
CA_T	clinical activity and temporal relations	17
CA_B_T	clinical activity, background information, and temporal relations	0
CI	condition based clinical activity	136
CI_B	condition based clinical activity and background information	3
CI_T	condition based clinical activity and temporal relations	38
CI_B_T	condition based clinical activity, background information, and temporal relations	5
	Sum	276

Table 6.3: Number of classified sentences for every class in the training corpus.

The analysis of our corpus also showed that multiple labels can be assigned to all sentences. We identified 4 different labels: "CA" for clinical activities, "CI" for condition based clinical activities, "T" for temporal relations, and "B" for background information. Due to the relationship between the labels "CI" and "CA", a condition based activity is a specialization (like in object-oriented designs) of the clinical activity. Hence, this multi-label classification turned into a hierarchical multi-label classification problem (HMC) [BK11]. The main difference to the normal multi-label approach is that a sentence marked with a specific label automatically gets the label of its superclass added (this fact is called the 'hierarchy constraint'). As an example, sentence #1 in Table 6.2 was labeled with "CA", "CI", and "B".

In our experiments we explored the multi-class as well as the multi-label approach in order to compare their performance values.

The next step was the selection of the appropriate machine learning algorithm. TimeML does not only distinguish between *EVENT* and *TIMEX* expressions, it also defines relations among them. We assumed that these relations show statistical dependencies, which can be used to train a classifier. The Conditional Random Field (CRF) method is based on such dependencies and on a rich feature set [SM10]. As both prerequisites were met, we applied this method to the experiments. This decision was additionally supported by works of Luo et al. [LJLW11], who showed that CRFs can be used to extract temporal constraints from clinical research eligibility criteria, and Sohn et al. [SWL$^+$13], who applied CRFs to detect comprehensive temporal information in clinical texts. Having the problem of the limited corpus in mind, we decided to complement our experiments with a second machine learning method in order to compare the results. The choice fell on the Support Vector Machine, because it is considered as *"a must try"* [WKRQ$^+$08, p.10] in text mining applications, and only needs a dozen examples for training.

Figure 6.1 shows the set up of our experiments. The training corpus was used to generate the classifiers which were applied to categorize the sentences of the test corpus. The single steps are discussed in the sections below.

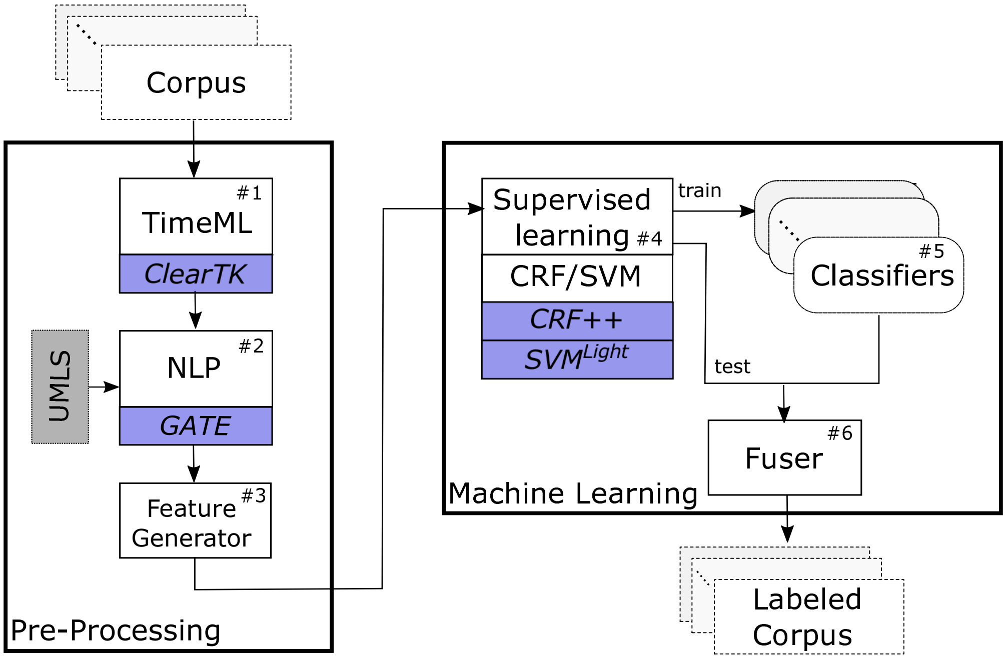


Figure 6.1: The general set up of the experiments – steps (marked with the # symbol), tasks (white rectangles) and corresponding software tools (blue rectangles), knowledge sources (gray rectangles), guideline documents (dashed rectangles), and the generated and applied classifiers (rounded rectangles).

6.2.1 Pre-Processing

In the first step (#1) of the pre-processing phase, TimeML annotations were assigned to the sentences of the corpus. Several tools were available for this task (see Chapter 4.3 for a detailed discussion), but none of them generated 'subordination links' and 'aspectual links' of TimeML. Finally, the 'clearTK-TimeML' application was selected despite the fact that it did not handle medical events. Hence, we had to apply natural language processing (NLP) techniques in step #2 to complete the TimeML annotations. The text engineering tool GATE 8.0 [CTRB13] with its configurable and extendable text processing resources fulfilled this task and generated TimeML compliant annotations or corresponding indicators by the following steps:

1. TIMEX annotations were created by means of the "Clinical Measurements and TimeML Annotator" developed by Phil Gooch. This is a GATE plugin to identify quantitative and temporal concepts in clinical texts [Goo12b].

2. Every medical term – found in the UMLS – which fell into the semantic group 'event' [CPH+02] led to an *EVENT* annotation.

3. If a sentence contained a trigger word of a lexicon (also called gazetteer in GATE) describing temporal relations and that word was positioned between two events, a *TLink* was generated.

4. Conditions in clinical guidelines are often expressed by different trigger words or phrases, such as 'if', 'in case of', 'patients with', etc. Sentences with such words induced lexically-based *SLink* annotations.

5. Aspectual links (*ALinks*) were produced when a sentence contained an aspectual verb, like *begin*, *stop*, etc. (for a list see [SLG+06]).

This stepwise approach produces a feature set based on identified TimeML concepts.

6.2.2 Feature Selection

Feature engineering is a human craft and the ultimate test for the optimal feature set has to be conducted empirically [MRS08]. Sentences in text classification can be represented by a large amount of features, but many of these could be irrelevant or noisy [MnFD+03]. For this reason the use of TimeML annotations as a starting point for the generation of the feature set (in the following referred as 'FS-A') seems to be appropriate. A second feature set 'FS-B', where clinical activities and trigger words for background information were used for additional features, complemented 'FS-A'. The clinical activities were identified based on the findings (verbs and semantic types of medical concepts) in [KSM10].

Two sentences and the mapping to their feature sets are shown in Table 6.4 and Table 6.5. The additional annotations for 'FS-B' were 'MM' for UMLS semantic types, 'ACT' for verbs describing activities, and 'BG' for words indicating background information. The feature generator component (step #3 in Figure 6.1) provided the features which were used in the machine learning applications.

6.2.3 Sentence Classification

Steps #4 to #6 in Figure 6.1 show an overview of the components used for our machine learning approach. In the training phase the selected algorithms generated the models for the different classifiers (step #5) which were then applied in the following testing phase. Every sentence in the test corpus was classified, and the individual results were aggregated in step #6 by means of the 'fuser' component (details for multi-classifier systems (MCS) can be found in [RP06]).

We started with the multi-class experiments and the feature set 'FS-A' followed by the multi-label classification with both feature sets. For multi-label classification we only used the CRF method, because in contrast to SVM it also offered a sequence labeling

Sentence	FS-A	FS-B	Annotations	Attributes
When	x	x	TLINK	overlap
	x	x	SLINK	conditional
FHR tracing is non				
reassuring	x	x	EVENT	occ., progr., none, pos, present
it should be				
maintained	x	x	EVENT	occ., none, should, pos, past
and				
reevaluated	x	x	EVENT	occ., none, none, pos, past
after	x	x	TLINK	after
a period of				
20 minutes	x	x	TIMEX	duration
because		x	BG	because
the fetus could be				
in a quiet period.				

Table 6.4: Feature sets - example 1; 'occ.'=occurrence, 'progr.'=progressive.

Sentence	FS-A	FS-B	Annotations	Attributes
After	x	x	TLINK	after
admission,	x	x	EVENT	occ., none, none, pos, none
		x	MM	[hlca]
FHR should be				
auscultated	x	x	EVENT	occ., none, should, pos, past
by				
Ultrasound	x	x	EVENT	occ., none, none, pos, none
		x	MM	[diap]
or				
using	x	x	EVENT	occ., none, none, pos, prespart
the stethoscop				
for a minimum of				
1 minute	x	x	TIMEX	duration
immediately	x	x	TIMEX	time
after	x	x	TLINK	after
a				
contraction	x	x	EVENT	occ., none, none, pos, none
		x	MM	[patf]
at least				
every 2 hours.	x	x	TIMEX	duration

Table 6.5: Feature sets - example 2; the values in square brackets indicate semantic types of the UMLS SN; 'occ.'=occurrence.

method to analyze the tokens of a sentence (the annotations, excluding the attributes, in Tables 6.4 and 6.5 defined the sequence of tokens).

The outcome of the different experiments was compared by selected standard performance measures in text mining (see Chapter 3 for details): Precision (Prec), Recall (Rec), Accuracy (Acc), and F1 based on the final sentence classification 'true-positive (tp)', 'true-negative (tn)', 'false-positive (fp)', and 'false-negative (fn)'.

$$Prec = \frac{tp}{tp + fp} \qquad Rec = \frac{tp}{tp + fn}$$
$$Acc = \frac{tp + tn}{tp + tn + fp + fn} \qquad F1 = \frac{2tp}{2tp + fp + fn}$$

For both classification approaches we empirically tested multiple feature sets, different parameters for the machine learning algorithms, and kernels for support vector machines, but in the following sections we only present and discuss the experiments (see Table 6.6) with the most significant results.

#	multi-class	multi-label	CRF	SVM	FS-A	FS-B	sentence	token
1	x		x		x		x	
2	x			x	x		x	
3		x	x		x		x	
4		x	x		x		x	
5		x	x		x			x
6		x	x			x	x	

Table 6.6: List and structure of the experiments.

Multi-Class Classification

We structured our classification task according to Manning et al. [MRS08]:

1. Training sets of sentences for every class were generated. A sentence that belonged to a specific class of a training set received a positive label while the others got a negative one. Based on the classes, 7 classifiers were implemented.

2. Each of the classifiers was applied to each of the test sentences.

3. A sentence was assigned according to

   - the maximum confidence value in CRF experiments, and
   - the greatest margin in SVM experiments,

   to its specific class (this was the task of the 'fuser' component).

Multi-Label Classification

The multi-label experiments were executed with the help of the CRF algorithm. The 'fuser' component was extended to handle the hierarchical multi-label classification based on the decision matrix shown in Table 6.7.

CI	CA	T	B	Labels
pos	pos|neg	pos	pos	CA, CI, T, B
pos	pos|neg	pos	neg	CA, CI, T
neg	pos	pos	pos	CA, T, B
neg	pos	pos	neg	CA, T
pos	pos|neg	neg	pos	CA, CI, B
pos	pos|neg	neg	neg	CA, CI
neg	pos	neg	pos	CA, B
neg	pos	neg	neg	CA
neg	neg	pos|neg	pos|neg	

Table 6.7: Decision matrix for the combination of labels implemented in the 'fuser' component('|' indicates a logical 'or' operation).

The algorithms for the calculation of the performance measures had to be adapted [VSS+08]. Let $tp_i/tn_i/fp_i/fn_i$ be the values for 'true-positive', 'true-negative', 'false-positive', and 'false-negative' for label i; the performance values are then defined as:

$$Prec = \frac{\sum_i tp_i}{\sum_i tp_i + \sum_i fp_i} \qquad Rec = \frac{\sum_i tp_i}{\sum_i tp_i + \sum_i fn_i}$$

$$Acc = \frac{\sum_i tp_i + \sum_i tn_i}{\sum_i tp_i + \sum_i tn_i + \sum_i fp_i + \sum_i fn_i} \qquad F1 = \frac{2\sum_i tp_i}{2\sum_i tp_i + \sum_i fp_i + \sum_i fn_i}$$

The software package CRF++ allowed not only unigram and bigram features, but also the construction of unigram features based on features of the predecessing and the following tokens in relation to the current token in the sentence[5]. Combined unigrams were designed, based on empirical studies in the 5-fold cross validation experiment (see Table 6.12) and, hence, reused.

6.3 Evaluation and Discussion

The quality of classifiers strongly depends on the available training data. The 'clearTK-TimeML' application, which was responsible for the initial TimeML annotations, per-

[5] A detailed description of the design practices can be found here: http://crfpp.googlecode.com/svn/trunk/doc/index.html?source=navbar (last accessed March 16, 2015)

formed pretty well at the tempEval competition, in comparison to other tools. However, the F1 scores for *EVENT* annotations (77.3%), for *TIMEX* annotations (82.7%), and for correctly identified relations (31.0%) [Bet13], heavily affected our classifiers' work. Furthermore, the generation of TimeML compliant annotations via GATE and the UMLS SN was far from complete. For example, the term "FHR" which is the abbreviation for "fetal heart rate tracing" was not listed in the UMLS database and, therefore, not annotated as *EVENT*. Additionally, the processing resource "Flexible Exporter" of GATE produced non-reliable results (e.g., the annotation of the word 'frequently' was sometimes exported, sometimes not). So far, we can assume that improvements of the used tools and knowledge sources would also raise our performance rates.

Experiment #1 was executed with the feature set 'FS-A' and the CRF algorithm. Based on this feature set, 204 features (unigram and bigram features) were generated in the training phase by CRF++. The performance results of the classifiers and the overall values are shown in Table 6.8 and the corresponding confusion matrix could be found in Table 6.9.

Class	tp	tn	fp	fn	Prec	Rec	Acc	F1
CA	1	254	27	26	3.6%	3.7%	82.8%	3.6%
CA_B	0	304	0	4		0.0%	98.7%	0.0%
CA_T	0	295	0	13		0.0%	95.8%	0.0%
CI	48	152	90	18	34.8%	72.7%	64.9%	47.1%
CI_B	0	302	0	6		0.0%	98.1%	0.0%
CI_B_T	0	298	0	10		0.0%	96.8%	0.0%
CI_T	4	267	13	24	23.5%	14.3%	88.0%	17.8%
overall	53	66	88	101	37.6%	34.4%	38.6%	35.9%

Table 6.8: Experiment 1: Multi-class, CRF, FS-A.

Class	CA	CA_B	CA_T	CI	CI_B	CI_B_T	CI_T	NO
CA	1	0	0	11	0	0	1	14
CA_B	0	0	0	1	0	0	0	3
CA_T	0	0	0	0	0	0	3	10
CI	0	0	0	48	0	0	2	16
CI_B	1	0	0	2	0	0	3	0
CI_B_T	0	0	0	6	0	0	1	3
CI_T	1	0	0	10	0	0	4	13
NO	25	0	0	60	0	0	3	66

Table 6.9: Confusion matrix for experiment 1: Multi-class, CRF, FS-A.

The SVM algorithm was used with a polynomial kernel in experiment #2. The features were derived from the feature set 'FS-A' and their weights were calculated by means of the 'term frequency - inverse document frequency' (tf-idf) [DTY+04] algorithm. The performance results are shown in Table 6.10 – the corresponding confusion matrix in Table 6.11.

Class	tp	tn	fp	fn	Prec	Rec	Acc	F1
CA	0	281	0	27		0.0%	91.2%	0.0%
CA_B	0	304	0	4		0.0%	98.7%	0.0%
CA_T	0	294	1	13	0.0%	0.0%	95.5%	0.0%
CI	63	82	160	3	28.3%	95.5%	47.1%	43.6%
CI_B	0	302	0	6		0.0%	98.1%	0.0%
CI_B_T	0	298	0	10		0.0%	96.8%	0.0%
CI_T	3	269	11	25	21.4%	10.7%	88.3%	14.3%
overall	66	42	112	88	37.1%	42.9%	35.1%	39.6%

Table 6.10: Experiment 2: Multi-class, SVM, FS-A.

Class	CA	CA_B	CA_T	CI	CI_B	CI_B_T	CI_T	NO
CA	0	0	0	20	0	0	1	6
CA_B	0	0	0	4	0	0	0	0
CA_T	0	0	0	3	0	0	3	7
CI	0	0	0	63	0	0	2	1
CI_B	0	0	0	3	0	0	0	3
CI_B_T	0	0	0	6	0	0	0	4
CI_T	0	0	0	16	0	0	3	9
NO	0	0	1	108	0	0	5	40

Table 6.11: Confusion matrix for experiment 2: Multi-class, SVM, FS-A.

The multi-class approach (see Tables 6.8 and 6.10), whether the CRF or the SVM version, produced F1 scores between 35% and 40%. The high scores (between 64% and 99%) for the accuracy for each classifier were based on the high 'true-negative' values of the individual classifiers, but in the end strikingly dropped to values between 35.1% and 38.6%. For the classes CA_B, CI_B, and CI_B_T the number of 'true-positive' values was zero due to the limited numbers of training sentences (2, 3, and 5 sentences). The confusion matrix in Table 6.9 showed the highest value for correctly identified C_I classes, but also the highest negative rate. One of the reasons why 60 sentences were incorrectly classified as C_I was that the test corpus also contained conditional sentences which did not describe condition based activities. They contained annotations for subordination links, but the other annotations provided too few distinguishing features. The same can be stated for the 25 sentences which were assigned to class CA. The confusion matrix of experiment #2 (Table 6.11) showed that most of the sentences were incorrectly classified as CI.

Due to the non-satisfying results of the multi-class approach and the fact that both algorithms (CRF and SVM) performed very poorly, our focus was set on the multi-label classification.

The 5-fold cross validation (see Table 6.12) was executed to check which results could be expected from the multi-label classification. The performance values ranged between 83.8% and 86.0%. The precision of the CI-label even showed a value of 100.0% and a recall value of 95.2%. Label B showed a 'true-positive' value of zero, caused by the lack of sufficient training data for this label.

Label	tp	tn	fp	fn	Prec	Rec	Acc	F1
CA	75	0	29	1	72.1%	98.7%	71.4%	83.3%
CI	100	0	0	5	100.0%	95.2%	95.2%	97.6%
B	0	105	0	0	0.0%			
T	28	44	6	27	82.4%	50.9%	68.6%	62.9%
overall	203	149	35	33	85.3%	86.0%	83.8%	85.7%

Table 6.12: Experiment 3: 5-fold cross validation – multi-label, CRF, FS-A, based on sentences.

Experiment #4 was executed with the feature set 'FS-A' and the CRF algorithm. More than 230 features were generated. The performance results of each classifier and the overall values are shown in Table 6.13. The results for the test corpus were in line with the expectations (see Table 6.13). Despite the fact that the scores were decreasing, this classification performed much better than the multi-class classification. A recall value of 78.0% (multi-class: 34.4%) and an accuracy value of 61.9% (38.6%) showed a substantial improvement to support our machine learning approach. The precision of 34.2.% was the consequence of the high number of 'false-positives'. The analysis of experiment #5 (see Table 6.14), which examined the tokens instead of the sentences, showed slightly higher performance values (recall 80.0% and accuracy 62.6%). The assumption that sentences with more tokens reached better scores seemed to be justified.

Label	tp	tn	fp	fn	Prec	Rec	Acc	F1
CA	51	13	242	2	17.4%	96.2%	20.8%	29.5%
CI	127	47	132	2	49.0%	98.4%	56.5%	65.5%
B	0	283	0	25	0.0%	0.0%	91.9%	0.0%
T	35	206	36	31	49.3%	53.0%	78.2%	51.1%
overall	213	549	410	60	34.2%	78.0%	61.9%	47.5%

Table 6.13: Experiment 4: Multi-label, CRF, FS-A, based on sentences.

Experiment #6 used the extended feature set 'FS-B' in the CRF algorithm (instead of 'FS-A' in experiment #4). This feature set led to more than 2500 different features in the training model. The results of each classifier and the overall values are shown in

Label	tp	tn	fp	fn	Prec	Rec	Acc	F1
CA	314	230	1263	24	19.9%	92.9%	29.7%	32.8%
CI	867	197	762	5	53.2%	99.4%	58.1%	69.3%
B	0	1650	0	181	0.0%	0.0%	90.1%	0.0%
T	332	995	336	168	49.7%	66.4%	72.5%	56.8%
overall	1513	3072	2361	378	39.1%	80.0%	62.6%	52.5%

Table 6.14: Experiment 5: Multi-label, CRF, FS-A, based on tokens.

Label	tp	tn	fp	fn	Prec	Rec	Acc	F1
CA	52	11	244	1	17.6%	98.1%	20.5%	29.8%
CI	123	41	138	6	47.1%	95.3%	53.2%	63.1%
B	0	283	0	25	0.0%	0.0%	91.9%	0.0%
T	30	217	25	36	54.5%	45.5%	80.2%	49.6%
overall	205	552	407	68	33.5%	75.1%	61.4%	46.3%

Table 6.15: Experiment 6: Multi-label, CRF, FS-B, based on sentences.

Table 6.15. We discovered that a richer feature set did not automatically lead to better results. The only slightly increasing score was the precision value (33.5%), all the others stayed low.

6.4 Conclusion

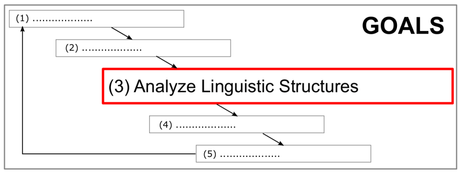
In this chapter we showed that TimeML annotations can help to automatically identify selected information dimensions of MHB. We used a supervised machine learning approach for multi-class and multi-label classification and applied different machine learning algorithms and feature sets. The limited number of sentences in the corpus and the quality of the top-ranked tools in this domain definitely affected the feature sets and, consequently, also the results. We detected that in multi-class classification the annotations did not deliver enough distinguishable features in order to justify this approach. In contrast, the multi-label classification produced expected results. The assumption that a richer feature set automatically increases the performance values was not confirmed[6]. However, the amount of tokens in a sentence had a distinctive influence on the results.

Finally, we can state that the multi-label classification is an adequate approach to support step #1 of our GOALS methodology.

[6]Experiments with features based on POS-tags and word stems also performed poorly

CHAPTER 7

Condition-Action Sentences



The linguistic structure of sentences (e.g., adverbial phrase, conditional phrase, etc.) is strongly related to specific information dimensions in a guideline, and – therefore – these relations can be used for step #3 of GOALS. In this chapter we focus on the identification of the *antecedent* (=condition) and the *consequent* (=consequence) in sentences describing condition-based activities. This process demands the correct recognition of the *condition-action* sentences (the terms *condition* and *action* are also used in the Guideline Elements Model (GEM) ASTM standard to describe *conditional recommendations* or *conditional statements* [SKA^{+}00]). As our machine learning algorithm produced a high false-positive rate for such sentences (see chapter above) we decided to rely on a rule-based approach to, firstly, identify the sentences, secondly, to split them into their linguistic parts, and thirdly, automatically select the relevant sentences for modeling.

7.1 Knowledge Sources and Tools

For the development of our method we used a chapter from an Asthma guideline developed by SIGN [Sco11], where *chapter 4: pharmacological management* had been modeled in the semi-structured modeling language MHB [SMM^{+}06] by a guideline modeling expert. The guidelines *Management of active low-risk labour - Admission for Birth* [Rem10] (chapters 1, 2.1, 2.2, and 2.3) and *CBO Treatment of Breast Cancer* (chapter 3) were applied to evaluate our method. These test guidelines were intentionally selected, because they cover a completely different medical application area, in contrast to the Asthma guideline, and – furthermore – an MHB-F[1] [SMM+06] model already existed, used as a "golden standard".

The Unified Medical Language System (UMLS) (see Chapter 6.1.2) with its semantic network was used to identify the semantic types of clinical concepts. The open source framework for text engineering GATE [CMB+11] handled the natural language processing. The following components were used in our method:

- ANNIE: A set of information extraction (IE) components, distributed within the GATE system and relying on finite state algorithms and the JAPE language [CMBT02].
- OpenNLPChunker supports the detection of phrases within a parsed text.
- MetaMap Annotator: A tagger that maps biomedical texts to the UMLS Metathesaurus and discovers Metathesaurus concepts and their semantic types [AL10].

7.2 Method

Generally, the discovery of *condition-action* combinations by means of heuristics is not a trivial one. On the one hand, *condition-action* sentences are rarely of the form '*if* condition *then* action', and on the other hand, conditions may refer to effects, intentions, or events and not activities, and these combinations must be sorted out by our method. Table 7.1 shows example-sentences in regard to their MHB aspects.

Condition-based medical activities are expressed in clinical practice guidelines in various ways and mostly found in single sentences. These sentences affect the clinical pathway and are, therefore, relevant to the computer-interpretable model of the guideline. In order to classify such a sentence as relevant, we based our approach on the following hypothesis:

1. A sentence owns a certain domain independent linguistic structure, and
2. contains recurrent domain dependent semantic key patterns.

We propose a rule-based, heuristic method using linguistic and semantic patterns to classify sentences in CPGs as relevant for describing conditional activities in order to move towards an automatic translation of such sentences into MHB in a following future step (an example is shown in Figure 7.1). Therefore, we analyzed a CPG document to develop a general linguistic pattern set and a semantic pattern set based on UMLS Semantic Types. These pattern sets then formed the basis for the subsequent classification by calculating the *relevance rate (rr)*.

The goal of the method is to automatically identify relevant *condition-action* sentences for modeling in order to support the modeler's work. As we use MHB as the

[1]MHB is the former version of MHB-F

sentence	MHB aspect
An episiotomy should be performed if there is a clinical need such as instrumental birth or suspected fetal compromise.	decision based activity
Women with pain but no cervical changes should be re-examined after two hours.	decision based activity
Women should be informed that in the second stage they should be guided by their own urge to push.	clinical activity
The partogram should be used once labour is established.	background information
Administration of inhaled steroids at or above 400 mcg a day of BDP or equivalent may be associated with systemic side-effects.	effect

Legend: activity, condition, effect, explanation

Table 7.1: Examples of sentences and their categorization in MHB.

example sentence

"If the patient does not fit the criteria for admission an assessment of fetal wellbeing including appropriate use of electronic fetal monitoring should be performed."

corresponding MHB chunk (XML)

```
<control>
    <if-then
        condition="NOT (admission criteria)"
        degree-of-certainty="Should"
        result="Assessment of fetal wellbeing including monitoring">
    ..................
</control>
```

Figure 7.1: A condition-action sentence represented as MHB-chunk.

target language, a formalized representation of complex conditions (such as the ERGO-annotation ontology proposed by Tu et al. [TPC+11] to transform free-text eligibility criteria into computable criteria) is not required. However, when moving towards a formal CIG language, in the future such a formalization step will be necessary.

7.2.1 Manual Development of the General Linguistic Pattern Set

We analyzed the selected chapter of the Asthma guideline with regard to the control flow aspects and started generating an initial linguistic pattern set based on trigger words (12 occurrences for 'if' and 4 occurrences for 'should').

#	rule type	pattern	weight(w)
1	IF	* [Ii]f *	0.5
2	IF	If {condition} {consequence}.	1.0
3	IF	If {condition}, {consequence}.	1.0
4	IF	{consequence} if {condition}.	1.0
5	IF	If {condition} then {consequence}.	1.0
6	SHOULD	* should have *	0.5
7	SHOULD	* should be *	0.5
...	..	...	...

Table 7.2: Selected general linguistic patterns.

In order to identify the semantic clauses of a sentence (these are the parts describing the *condition* and the *consequence*), the patterns had to be grouped into 6 different patterns for 'if' and 4 different patterns for 'should' (some straightforward patterns are listed in Table 7.2). Condition and consequence are distinguished according to the sentence's syntax, punctuation and its sequence of phrases. Two complex examples including conditions spread over multiple phrases are shown in Figure 7.2.

If there is a response to LABA , but control remains poor , continue with the LABA and increase the dose of inhaled steroid to ...

In children under five years , higher doses may be required if there are problems in obtaining consistent drug delivery.

Legend: condition , consequence

Figure 7.2: Sentences with multi-part conditions.

We assigned a weighting factor to every pattern of the set - the value 0.5 to show that only a trigger word was found and 1.0 to express that also the semantic clauses were identified. These constants can be adapted for new rule types in the future.

7.2.2 Generation of the Domain-Dependent Semantic Pattern Set

Amongst the general syntactic patterns we also used semantic patterns based on the UMLS Semantic Network. Therefore, we used the MetaMap plugin within the GATE framework to automatically identify medical concepts in our text and assign them to their corresponding UMLS concepts and semantic types (represented by four-letter abbreviations - e.g., 'popg' stands for 'Population Group'). By this way it was possible to find out the sequence of semantic types in the clauses of the sentences (see Figure 7.3).

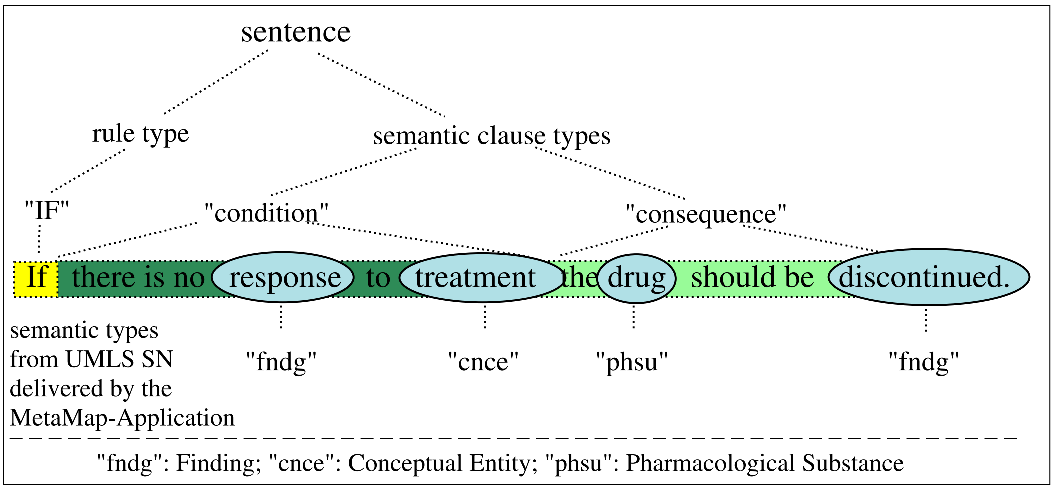


Figure 7.3: Semantic abstraction of a sentence.

Finally, the complete semantic abstraction of a sentence – including rule type, semantic clause type, and sequence of semantic types – was added to the semantic pattern pool (see Table 7.3). A total of 32 entries was automatically generated.

rule type	**semantic clause type**	**sequence of semantic types**
IF	condition	[fndg][cnce]
IF	consequence	[phsu][fndg]
IF	consequence	[idcn][qlco][resa][ftcn][ftcn]
SHOULD	condition	[aggp][podg][dsyn]
SHOULD	consequence	[qnco][tmco][resa][orch, phsu][idcn]
SHOULD	condition	[podg][qlco][ftcn][orgf][qlco][gngm][phsu]

Table 7.3: Structure of the semantic pattern pool (selected samples).

7.2.3 Calculation of the Relevance Rate

The relevance rate rr is a measure to find out whether a sentence contains a condition-action combination. Furthermore, it shall classify a sentence as crucial for the clinical pathway in contrast to other information aspects, like intentions or explanations, which are modeled in MHB-F in a different way. To find this semantic difference, the syntax of the sentences as well as the sentences' containing medical semantic types must be respected. In order to calculate the relevance rate for a selected sentence, its semantic abstraction has to be generated and compared with every entry in the semantic pattern pool. If the rule type and the semantic clause type are matching, the similarity of the sequences of the semantic types is calculated by using the Dice coefficient [MS99]. The highest value is selected for further calculation.

In general, the value of the relevance rate rr is the sum of

- the weight(w) of the applied general IE rule, and
- the sum of the maximum similarity value (s_i) for each semantic clause of the sentence, divided by the number of semantic clauses (n) identified by the general IE rules.

$$rr = w + \frac{\sum_{i=1}^{n} \max\{s_i\}}{n} \tag{7.1}$$

The weight of the IE rule and the arithmetic average of the similarity values - both in the range between 0 and 1 - have the same influence on the rr.

The similarity value s_i of the semantic clause of a sentence and a matching entry in the semantic pattern pool were calculated as follows:

- If the semantic clause contains only one semantic type, it is compared to those entries of the semantic pattern pool that also show only one semantic type (to receive better accuracy). In the case that both types are equal the value for s_i is set to 1.0, otherwise
- both sequences of semantic types are interpreted as a string each, and two sets of 4-letter string bigrams are composed out of them. Subsequently, these two sets are used for the calculation of the Dice coefficient.

Example:
Given are the sequence of semantic types of a new semantic clause S and the sequence of semantic types of a matching entry of the semantic pattern pool P.

S: [fndg] [orgf] [qlco] [gngm] [phsu] and P: [strd] [gngm] [phsu] [ortf].

The resulting 4-letter string bigrams are:

$S = \{``fndgorgf'', ``orgfqlco'', ``qlcogngm'', ``gngmphsu''\}$

$P = \{``strdgngm", ``gngmphsu", ``phsuortf"\}$.

The Dice coefficent is defined as twice the shared information over the sum of cardinalities:

$$s_i = \frac{2n_t}{n_s + n_p} \tag{7.2}$$

where n_t corresponds to the number of bigrams found in both sets, n_s is the number of bigrams in S, and n_p the number of bigrams in P. So the result of this example is $s = \frac{2*1}{4+3} = \frac{2}{7}$.

The general interpretation of the rr is shown in Table 7.4.

value	**interpretation**
$rr = 0.5$	only a trigger word or word combination were found; no semantic clause could be identified
$rr = 1.0$	an appropriate general IE pattern was found and semantic clauses could be identified
$1.0 < rr <= 2.0$	additionally, a semantic similarity between the sequences of the semantic types was detected

Table 7.4: Interpretation of the relevance rate.

7.3 Evaluation and Discussion

The IE rules for the domain independent patterns and the generation of the semantic patterns were implemented with GATE – the processing resources for the calculation of the relevance rate are shown in Figure 7.4. The IE rules were applied to the guidelines

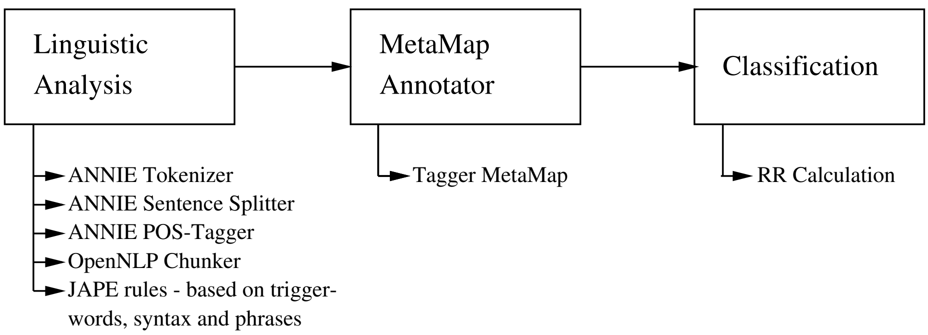


Figure 7.4: Processing resources used in GATE.

and the identified sentences were annotated with the semantic types retrieved from the

UMLS via the MetaMap plugin. 50 sentences were found which complied to the IE rules. Twenty out of the 50 sentences had been modeled in MHB as control flow related aspects (activities based on special medical conditions), and – therefore – their *rr* was expected to be higher than 1. The other 30 sentences represented information about intentions and explanations in MHB with an expected *rr* lower or equal 1. The evaluation results are shown in Table 7.5 differentiated by the rule types.

			type: IF		type: SHOULD		total	
Gold Standard	*tp*	*fn*	14	3	1	2	15	5
	fp	*tn*	1	2	1	26	2	28

tp = true positive; fn = false negative; fp = false positive; tn = true negative

Table 7.5: Evaluation results I (rr >1: tp and fp; $rr \leq 1$: fn and tn).

	type: IF			type: SHOULD			total		
Guideline	REC	PRE	NPV	REC	PRE	NPV	REC	PRE	NPV
Breast Cancer	100%	67%	100%	-	-	100%	100%	67%	100%
Adm. for Birth	80%	100%	0%	33%	50%	88%	72%	93%	75%
total	82%	93%	40%	33%	50%	93%	75%	88%	85%

REC the number of correctly identified sentences over the number of the modeled sentences in the golden standard (=recall)

PRE the number of correctly identified sentences over the entire number of identified sentences (=precision)

NPV the number of correctly not identified sentences over the number of not modeled sentences in the golden standard (=negative predictive value)

Table 7.6: Evaluation results II.

With the described method it was possible to identify 15 sentences which correctly contained control flow related aspects. Only two sentences got an incorrect *rr* higher than 1. They were not correctly rated, because they did not describe condition-based activities. Other 5 sentences got an *rr* lower than 1, although their *rr* should have been higher as they contained condition-based activities. Thus, the method had a recall of 75% and a precision of 88% (see Table 7.6). The negative predictive value of 85% was higher than the recall value and showed that 28 sentences had been correctly classified.

Generally, the results proved that the rules of type "if" showed much better results for the precision (93%) and the recall (82%) than the ones of type "should". Nevertheless, the negative predictive value for the latter type showed a rate of 93%.

The analyzed guidelines contained 7 sentences with control flow related information, but they were not classified, because their patterns did not exist in the Asthma guideline. Consequently, no corresponding general linguistic IE rule existed in the pattern pool. An extension of the general linguistic pattern set with rules for the trigger words "when", "could" and "in case of" should be taken into consideration. Additionally, one condition based activity could not be found, because the semantic information was distributed over more than one sentence. In Table 7.7 selected examples are shown with a relevance rate lower or equal than 1.0 together with the corresponding reasons.

sentence	rr	reason
If, notwithstanding these procedures cervical dilatation doesn't progress, consider cesarean section after 2-3 hours of regular and painful contraction with no cervical changes.	0.5	Sentences with such a linguistic structure did not exist in the Asthma guideline → no IE rule was implemented to identify semantic clauses
If dilatation progress is not regular (<1 cm/hour in nulliparous, <1,5 cm/hours in parous) consider: *- amniotomy;* *- oxytocin administration.*	0.5	A list of resulting activities was not found in the Asthma guideline → no IE rule was implemented to identify semantic clauses
In case of abnormal FHR, monitoring should be continuous.	1.0	The sentence was wrongly categorized with a rule for 'should' because no rules for "in case of" were implemented → no semantic similarity within the semantic pool pattern was found
After umbilical cord clamping, if the second stage of labour has been physiological, the baby is given to the mother and covered.	1.0	The semantic clauses were found, but no semantic similarity occured

Table 7.7: Selected sentences with $rr \leq 1.0$.

Even though only a small amount of training data (16 sentences) was available from the Asthma guideline, our method identified condition-based activities for control flow related aspects in a guideline document. Furthermore, it showed that the combination of domain independent information extraction rules and an automatically created semantic pattern pool leads to valuable results.

7.4 Conclusion

The aim of this method is to identify the antecedent and the consequent of a condition-based activity. By defining a set of linguistic patterns, we split up sentences semantically - from one selected training guideline - into their clauses showing the condition and the consequence. We used the UMLS Semantic Network to find out which types of medical concepts were applied in these clauses. The outcome was a semantic abstraction of every training sentence which then was stored in a semantic pattern pool. This pool facilitated the classification of new sentences regarding to their relevance to the corresponding MHB model expressed by the measure relevance rate (rr).

Modeling experts benefit from the method in two ways:

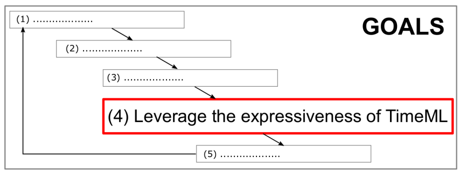
1. Condition-based activities in free-text guidelines, which must be modeled in MHB, are identified and rated.

2. These sentences are automatically split into the condition and the resulting activity.

Integrating the presented method into modeling tools will ease the work of all parties involved. Moreover, this information extraction method shows how step #3 of our GOALS methodology can be implemented.

CHAPTER 8

Extension of TimeML-Concepts



In the chapter above we demonstrated a method which extracts the *antecedent* and the *consequent* of a *condition-action* sentence. In this chapter we reuse parts of this method to generate subordination links (*SLinks*). In order to exploit these link annotations for the modeling process of guidelines, we extend the corresponding TimeML specification. Such extensions can be specified in two ways – either by introducing new attributes for existing concepts, or by providing new values for existing attributes. In this chapter we discuss both approaches – we expand the value-set of the 'relType' attribute of the *SLink*, and specify an additional attribute for the *EVENT* concept. These extensions show, how the expressiveness of TimeML can be increased as it is demanded in step #4 of GOALS.

8.1 Extensions to *SLinks*

The TimeML specification for subordination links (*SLinks*) distinguishes among different types of relations: *modal, factive, counter-factive, evidential, negative evidential,* and *conditional* (see examples in Table 8.1 [PCnI+03]). The *SLinks* are either lexically based (e.g., indicated by reporting verbs, perception verbs, intentional processes, etc.) or structurally-based (e.g., purpose clauses and conditional constructions) [SLG+06]. We set our focus on the latter ones, because they can be used to represent condition-based activities, which are often found in clinical care paths.

relType	example
modal	*Mary wanted John to buy some wine.*
factive	*John managed to leave the party.*
counter-factive	*John prevented the divorce.*
evidential	*John said he bought some wine.*
negative evidential	*John denied he bought only beer.*
conditional	*If John brings the beer, Mary will bring the chips.*

Table 8.1: Examples for subordination links and their relation types (‘relType’) (taken from [PCnI+03]).

The following sentences contain ‘structurally-based’ subordination links relating the introducing *EVENT* (marked in bold) to the *EVENT* in the consequent (underlined).

(1) *If there is no **response** the drug should be discontinued.*
(2) ***Women with pain** should be re-examined after two hours.*
(3) *The partogram should be used once labour is **established**.*

The *SLinks* in the sentences above are all ‘structurally-based conditional constructions’. However, in regard to the different information dimensions in CIGs, the sentences #1 and #2 represent ‘control-flow related aspects’ in contrast to sentence #3 which only describes ‘background information’.

As TimeML does not provide any annotation mechanism to distinguish among these information dimensions, we define the following additional values to the ‘relType’ attribute of *SLinks* (standard values for this attribute are listed in Table 8.1):

‘conditionalCF’ is used to specify conditional constructions to describe control-flow related aspects of a guideline.

‘conditionalBG’ is used to specify conditional constructions to describe background information.

```
The partogram should be <EVENT eid="e1">used</EVENT>
once labour is <EVENT eid="e2">established</EVENT>

<SLINK eventInstanceID="e2" subordinatedEventInstance="e1"
relType="conditionalBG">
```

Figure 8.1: Generation of the *SLink* annotation – example 1.

The newly introduced specification applied to sentence #3 is shown in Figure 8.1. Based on this categorization of *SLinks*, a tool that supports the formalization of a guideline can provide important information to the modeler (see also Chapter 9.1).

8.2 Extensions to *EVENTs*

Another possibility to expand the semantics of TimeML is the specification of new attributes for existing TimeML concepts. In this section we define a new attribute for the *EVENT* annotation. Based on it, we develop a method to identify sentences containing condition-based activities describing control-flow related aspects in a guideline, and to automatically generate *SLinks* of 'relType' *conditionalCF*.

8.2.1 Knowledge Sources and Tools

The TARSQI Toolkit (TTK) was one of the first implementations which generated TimeML compliant annotations in order to enable temporally based questions about *EVENTs* in news articles [VP08]. Due to the absence of structurally-based rules in the latest TTK version, we decided to implement our prototype built on the open source framework for text engineering GATE [CTRB13]. We reused the information extraction rules developed in Chapter 7 for identifying and splitting condition-action sentences.

8.2.2 Method

The generation of *SLinks* requires the correct identification and categorization of the underlying *EVENTs* within a sentence. Therefore, conforming to the TimeML specification, we manually annotated the *EVENTs* in every conditional sentence from an Asthma guideline [Sco11]. Then, we applied the structurally-based information extraction rules of Chapter 7 to these sentences in order to identify their antecedents (representing the condition) and their consequence clauses (indicating the subordination). Then we developed a categorization scheme for *EVENTs* to express their *role* in the clinical workflow. Based on the *EVENT's role* in the antecedent and the *EVENT's role* in the consequent, we developed filter rules to sort out sentences according to their control-flow relatedness in guidelines. Consequently, the subordination links of the sentences #1 and #2 should be labeled as control-flow related in contrast to sentence #3 which only describes 'background information'.

Finally, we manually annotated the *EVENTs* of condition-action sentences of six guidelines from different medical areas and evaluated our method.

8.2.3 Attribute for *EVENT* Annotations

As we needed additional semantic information for *EVENTs* in respect to our task, extensions to TimeML were necessary. TimeML specifies different types of *EVENTs* (e.g., occurrence, state, reporting, etc.) to enable temporal reasoning. However, these types did not prove sufficient. Therefore, we analyzed the training sentences of the Asthma guideline and identified three different *roles* of *EVENTs*. This additional information was then stored in an attribute called *role* (see Table 8.2).

role	ant	cons	examples
activity	x	x	add-on therapy; exercise; medication
status	x		is established; is required
action		x	discontinue; stop; increase; continue

Table 8.2: *Roles* of *EVENTs*; labels for being used in the antecedent (ant) and/or in the consequent (cons) of a sentence.

Some *roles* were only found in the antecedent, some in the consequent, and some in both. If a valid combination of these concepts (as defined in Table 8.3) is found in a conditional sentence then the sentence is classified as a control-flow related one. This classification is then stored in the attribute “relType” of the *SLink* annotation expressed by the value “conditionalCF” (see also Chapter 8.1).

antecedent	consequent
activity	activity
activity	action
status	activity
status	action

Table 8.3: Valid combinations of *roles* to categorize a sentence.

The following example[1] (see Figure 8.2) shows the manual annotations of the *EVENTs* of sentence #1 and the automatically generated annotation for the *SLink*.

```
If there is no response to <EVENT eid="e1" role="activity">treatment</EVENT>
   the drug should be <EVENT eid="e2" role="action">discontinued</EVENT>

         <SLINK eventInstanceID="e1" subordinatedEventInstance="e2"
                        relType="conditionalCF">
```

Figure 8.2: Generation of the *SLink* annotation – example 2.

The *role* ‘activity’ has been assigned to the *EVENT* in the antecedent and ‘action’ to the *role* of the *EVENT* in the consequent. This is an allowed combination of *EVENTs* (see Table 8.3) to categorize the subordination link as control-flow-related.

[1]The *EVENT*-tags only show the attributes relevant to our method

8.2.4 Evaluation and Discussion

We manually annotated the *EVENTs* in a corpus of six different guidelines[2] according to the definition presented in subsection 8.2.3. As these guidelines were already used in a former project, the condition-action sentences were already annotated. The corpus contained 68 conditional sentences including 147 *EVENTs*. 34 of these sentences described conditional activities which should be identified by our method. 35 *EVENTs*, which did not follow the categorization scheme, were annotated as type "other". After applying our prototype to the corpus, the sentences were classified as shown in Table 8.4.

classification	true	false	total # of sentences
positive	22	10	32
negative	24	12	36
total	46	22	68

Table 8.4: Results of the classification process of the 68 conditional sentences according to their control-flow-relatedness.

The precise identification of subordination links requires the identification of the antecedent and the consequence in a conditional sentence. Even though every if-sentence in the corpus was found, the segmentation of three sentences failed due to missing IE patterns.

The main reason why some subordinating clauses were not found was the fact that the *EVENTs* did not correspond to the defined *roles*. For example, in the sentence *"Consider referral if patient has **symptoms** of peripheral vascular disease such as loss of pulses and/or claudication"* a *role* for the *EVENT* "symptoms" was not defined (such a *role* was not provided by our training corpus). The next sentence showed an equal problem: *"If two different **tests** are available, then the test whose [sic] result is above the diagnostic threshold should be repeated"*. The *EVENT* "tests" had no corresponding *role*.

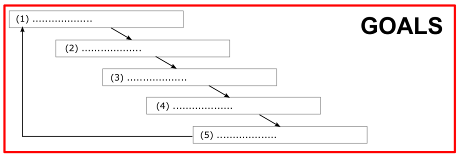
8.2.5 Conclusion

We showed that an additional categorization of *EVENTs* in conditional sentences helps to classify such a sentence as relevant to the clinical care-path. The main focus was set on showing how the TimeML specification can be extended to express this information (as demanded in step #4 of GOALS). By extending the pool of IE patterns for the segmentation of conditional clauses and by adapting and expanding the coarse grained *roles* for *EVENTs*, our method includes the potential of further development in this direction.

[2]Guideline topics: Chronic Hypertension in Pregnancy; Diagnosis and Management of Type 2 Diabetes Mellitus in Adults; Hypertrophic Cardiomyopathy; Gestational Diabetes; Breast Cancer; Chronic Heart Failure

CHAPTER 9

Scenario Based Application of GOALS



The previous chapters presented different IE approaches to support steps #1, #3, and #4 of GOALS. In the following chapter we apply the multi-step methodology in a scenario-based approach (also including steps #2 and #5). As our method is defined on a high level of abstraction and aims at the support of the complete translation process of a guideline into its CIG model, it is not restricted to single condition-action sentences.

In this chapter we apply GOALS onto a non-trivial example of a protocol to show the generic approach of GOALS. Every step was done manually without using IE methods in order to exclude the given uncertainness. Knowing that the significance of a scenario-based evaluation is limited, an overall evaluation is not possible at this stage and should be part of a further research project. We selected MHB as target language (as already discussed in Chapter 5) to show the functioning of GOALS. In MHB the knowledge of a CPG is represented in chunks that correspond to specific information in the CPG (e.g., a sentence, part of a sentence, more than one sentence). These chunks are associated to predefined dimensions such as control flow, data flow, evidence, and temporal concepts. The aspects of every dimension are described by using natural language, partly copied from the original guideline text or subsumed and enriched with the knowledge of the modeler. As there is no explicit specification on how to express these aspects linguistically, their descriptions can vary syntactically from modeler to modeler.

9.1 GOALS in Action

We selected a text passage of the guideline *Management of active low-risk labour - Admission for Birth* [Rem10] according to the following criteria: It should (1) contain MHB-control dimensions, (2) express a condition-based activity, (3) consist of at least two consecutive sentences which are temporally related to each other in order to get a higher complexity involved, and (4) contain different kinds of subordination clauses (e.g., conditional and causal clauses).

The selected sentences and the contained TimeML concepts (annotated in **bold style** and marked up with '*_e?*' for *EVENTs* and '*_t?*' for *TIMEX* expressions) are shown below. The annotations were manually added as defined in the TimeML specification.

(1) *When **FHR tracing**_e1 is reassuring at **admission**_e2, the woman should be **allowed**_e3 to move freely, even if membranes are not **intact**_e4.*

(2) *When **FHR tracing**_e5 is non reassuring it should be maintained and **reevaluated**_e6 after a period of **20 minutes**_t1 because the fetus could be in a quiet **period**_e7.*

The verb 'maintained' in the second sentence is not tagged as an *EVENT*, because if we consider 'FHR tracing' as a *narrative container*, 'maintain' expresses no *EVENT* during the given time span. Table 9.1 shows the identified TimeML links and their relevance to the modeling process. In general, *SLinks* of type 'conditional' describe condition/action sentences, and are, therefore, relevant to the model (see Table 9.1: #1, #2, and #7).

Sent.	#	Link	relType	MHB dim	relevant
1	1	S(e1, e3)	conditional	control	yes
	2	S(e4, e3)	conditional	control	yes
	3	T(e1, e2)	simultaneous	time	yes
	4	T(e2, t0)	identity	time	yes
	5	T(e2, e4)	simultaneous	time	no
	6	T(e1, e4)	simultaneous	time	no
2	7	S(e5, e6)	conditional	control	yes
	8	S(e7, e6)	modal	background	yes
	9	T(e5, e7)	simultaneous	time	no
	10	T(e6, t1)	after	time	yes
	11	T(t0, t1)	after	time	no
1+2	12	T(e1, e5)	identity		yes

Table 9.1: TimeML-Links (S=subordination link and T=temporal link – including their attribute 'relType') and their mapping to MHB dimensions (t0 describes the date of admission).

Link #8 shows a structurally-based subordination relation representing a reason clause. Furthermore, *TLinks*, which anchor events to specific *TIMEX* expressions, are relevant for the time dimension in MHB (links #4 and #10). The *TLink* #3 is also included, because it is transitively linked to t0 via e2. Link #12 shows the identity e1 = e5, and – in consequence – link #3 has to be modeled, too.

[G]enerate Templates and Slots

When going through the list of temporal relations of Table 9.1, the corresponding MHB structure is built (see Fig. 9.1 – step #1). Link #1 indicates a condition based activity, therefore, a new control tag – containing an if-then tag and its attributes – is generated. As there is no container for the control tag so far, it is embedded within a surrounding chunk tag. As link #2 is also related to event *e3*, no new control tag is necessary. A new time tag is created for link #4, because it relates to the explicit *TIMEX* expression *t0* in contrast to link #3. The subordination link #7 belongs to a new sentence, hence, a new control tag with a subordinated if-then tag is created. The reason clause of link #8 shows an explanation for an event, and – therefore – is bound to the MHB background tag. The temporal link #10 mandatorily results in a time tag, and the identity of *e1* and *e5* expressed in link #12 shows that although two sentences are analyzed together, only one chunk tag is needed.

Step #1		Step #2	Step #3
chunk			
control			
if-then			
condition			FHR tracing is reassuring at admission Even if membranes are not intact [ERROR]
degree-of-cert.		should	should
result			the woman should be allowed to move freely
control			
if-then			
condition			FHR tracing is non reassuring
degree-of-cert.		should	should
result			FHR tracing [it] should be maintained and reevaluated [after a period of 20 minutes]
background			
explanation			
information			because the fetus could be in a quiet period

Figure 9.1: GOALS compliant modeling process: From protocol to MHB – steps #1 to #3.

[O]btain Information from TimeML Annotations

TimeML annotations contain specific information about *EVENTs* which is transferred directly into the open information slots (see Fig. 9.1 – step #2). In our case we identify the modality attribute ‘should’ of the subordinated events in links #1 and #7 and map them to the degree-of-certainty attribute of the control tag.

[A]nalyze Linguistic Structures

As MHB aspects represent chunks of information – commonly as parts of sentences – these sentences have to be linguistically analyzed. The control tag, for example, needs the antecedent and the consequent of the condition clause for its attributes ‘condition’ and ‘result’. Based on the linguistic patterns developed in Chapter 7.2.1 and their manual application to the first sentence, the ‘condition’ consists of two phrases. However, the second phrase describes some background information and should not be part of the antecedent (see the *ERROR* mark in Figure 9.1). In order to solve this problem, the identification of adverbial clauses is recommended.

In the second sentence, the anaphora resolution in *‘.. it should be maintained’* leads to ‘FHR tracing should be maintained’. This resolution is necessary because of the splitting of the conditional sentence. The resulting structure is shown in Fig. 9.1 – step #3.

[L]everage the Expressiveness of TimeML

The first sentence contains two different subordination links describing two antecedents. Hence, the condition attribute also contains both. Structurally, the second subordination clause is an *SLink* of type condition, but semantically it only contains background information. In order to solve this problem, we apply the extension to the *SLink* concept defined in Chapter 8. Consequently, we reset the ‘relType’ of the links #1 and #7 to *‘conditionalCF’* and the ‘relType’ of link #2 to *‘conditionalBG’*. Following this procedure, we are able to generate a new background tag, and solve the problem of the wrongly assigned antecedent in the first control tag. Furthermore, due to the relation chain S(e1,e3) - T(e1,e2) - T(e2,t0), we can now set the information slots for the time tag based on S(e1,e3) – subject = e1 and start = e3.

[S]eek Information from Lexical Resources

In order to complete the structure of the MHB chunk, we have to add the ‘*data*-tag’ describing the piece of information used in this chunk (e.g., FHR tracing). We use the Unified Medical Language System’s Semantic Network (UMLS SN) [LHM93] to assign clinical concepts to nouns or gerundive constructions in the sentence. Generally, if the semantic type of the clinical concept is ‘finding’, ‘organism function’, or ‘qualitative concept’, it can be assigned to the name attribute of the ‘MHB *data*-tag’. The results of steps #4 and #5 are shown in Fig. 9.2.

				Step #4 + #5	Gold Standard
chunk					
	control				
		if-then			
			condition	FHR tracing is reassuring at admission	FHR tracing is reassuring at admission
			d-o-c	should	should
			result	woman should be allowed to move freely	No limitations in movements of woman
	background				
		explanation			
			info	Even if membranes are not intact.	Even if membranes are not intact.
	time				
		subject		FHR tracing	FHR tracing
			start-pv	At admission	At admission
	data				
		usage			
			name	FHR tracing	FHR tracing
	control				
		if-then			
			condition	FHR tracing is non reassuring	FHR tracing is non reassuring
			d-o-c	should	should
			result	FHR tracing should be maintained and reevaluated after a period of 20 minutes	Maintain AND reevaluate FHR tracing
	time				
		subject		FHR tracing	Reevaluate FHR tracing
			start-pv	After 20 minutes	After a period of 20 minutes
	background				
		explanation			
			info	because the fetus could be in a quiet ...	because the fetus could be in a quiet period

Figure 9.2: GOALS compliant modeling process: From protocol to MHB – comparison to the Gold Standard.

9.2 Evaluation and Discussion

Knowledge engineers, when modeling a guideline in MHB, are allowed to change the wording of the text when conserving the original semantics. Consequently, an evaluation of our results with the gold standard on the basis of a 'word by word' matching technique does not seem adequate. Thus, a comparison based on semantics is the appropriate solution to handle this situation.

The analysis of the results of step #5 (see Figure 9.2) shows that the MHB chunk (=templates), its dimensions and aspects are correctly identified by our method. Moreover, all thirteen aspects (=information slots) are correctly filled. The 'result' slot of the 'if-then' aspect is set to 'woman should be allowed to move freely' instead of 'No limitations in movements of woman'. Semantically both phrases express the same statement, but the wording is different. The second 'result' slot shows a very similar effect. As there is no specification in MHB about the exact formulation of the aspects in natural language (as already discussed above), both phrases are equivalent. The different wording in the 'precise-value' of the 'time' dimension as well as the 'subject' in the second 'time' dimension do not miss important information.

The application of our GOALS-methodology for this particular case reaches 'Level C' of our evaluation scheme, and – therefore – shows that the usage of TimeML concepts can ease this modeling process.

A kind of practical evaluation can be made if we calculate the reduction of user interactions (mouse clicks and keystrokes) in a software tool (like the DELT/A tool) for modeling our particular example. If we assume that our methodology is integrated in this tool, we counted that 114 operations within these two selected sentences can be omitted (including the change of the wording to the gold standard). If a sentence is not identified as relevant, the effort for modeling with this tool is as high as in the standard manual modeling process. In case that the methodology wrongly marks sentences as relevant, the automatically assigned annotations can be removed with two mouse-clicks. The overall period of time that can be saved – because the phrases in the guideline have already been identified, the templates selected, and the slots pre-filled – depends on the experience of the modeler.

The scenario-based application and evaluation of our methodology showed promising results. Nevertheless, a more thorough quantitative evaluation has to be planned as future research (see Chapter 10.3).

9.3 Conclusion

In Chapter 5 we presented the methodology GOALS, which proposes a step-by-step guidance to generate a computerized model of a clinical practice guideline. In this chapter, we showed the proper functioning of GOALS in a scenario-based approach where temporally related guideline sentences were transformed into the CIG language MHB. The result showed that our methodology eases the laborious translation task of the modelers. As GOALS focuses on temporal concepts and the relations between them, certain information dimensions of a guideline (e.g., levels of evidence) have not yet been taken into consideration. However, our methodology can easily be extended to future requirements due to its flexible structure.

Part III

Conclusion

CHAPTER 10

Conclusion and Future Outlook

In this final chapter we sum up the most important aspects of our work and give answers to our research questions. At the end, we indicate future directions towards the objectives of GOALS and its corresponding information extraction approaches.

10.1 Summary

In the first part of our work we described the problem of transforming guidelines into their CIG representation. We did extensive literature research to discuss the related works on which our methodology has been based. This literature research comprised state of the art CIG formalisms and intermediate representations as well as the use of the TimeML specification for annotating temporal concepts in different domains. An insight into information extraction techniques and supervised learning approaches concluded the related work-sections.

The second part dealt with the step-wise approach of the GOALS methodology. GOALS defined instructions on how to develop CIG model parts out of guideline documents. Provided on the temporal concepts of the TimeML specification language, it was designed to work independently from the target CIG language. A 5-level evaluation scheme was developed in order to inspect the quality of the generated model parts.

The intermediate CIG representation language MHB was selected as target language to demonstrate the benefits of our methodology. Different advantageous information extraction techniques were developed for steps #1, #3, and #4 of GOALS.

A multi-class and a multi-label classification were executed to identify the different MHB dimensions in a sentence of a guideline in order to support step #1 of GOALS. Therefore, machine learning algorithms (CRF and SVM) were implemented based on features provided by TimeML annotations. The multi-class classification did not perform as expected. However, the results of the multi-label approach were promising and were mainly restricted by the limited corpus and the quality of the used tools for the generation of TimeML annotations.

Heuristic-based information extraction rules were developed to identify condition-based activities in a guideline. By defining a set of linguistic patterns, we split up sentences semantically - from one selected training guideline - into their clauses showing the condition and the consequence. We used the UMLS Semantic Network to find out which types of medical concepts were applied in these clauses (such a linguistic analysis is the task of step #3 of GOALS). The outcome was a semantic abstraction of every training sentence which then was stored in a semantic pattern pool. This pool facilitated the classification of new sentences regarding to their relevance to the corresponding MHB model expressed by the measure relevance rate (rr).

Extensions to different TimeML concepts were defined (as demanded in step #4 of GOALS) to automatically find condition-based activities for control flow related aspects in a guideline document. These extensions were based on a heuristic, derived from semantic types of clinical concepts, and were used to develop filter rules for sorting out relevant sentences.

Finally, we demonstrated the proper functioning of GOALS by selecting two temporally related sentences of a protocol and transformed them into their CIG model. Step by step we developed the model until all dimensions and aspects of the MHB model were correctly identified and the information slots filled. The result showed that guideline modelers can benefit enormously from our methodology.

10.2 Answers to Research Questions

We supplemented our hypothesis of Chapter 1.2 with the following answers:

✓ *If model relevant condition-action sentences are related to recurring linguistic and semantic patterns, heuristic-based information extraction methods may be used to identify the antecedent and the consequence of a conditional sentence of a guideline.*

  In Chapter 7 we developed a rule-based, heuristic method that combines domain-independent information extraction rules and semantic pattern rules to detect *condition-action* sentences. We identified 16 different semantic patterns and applied them to selected sentences of a guideline reaching a recall value of 75% and a precision value of 88%.

✓ *The medical concepts – identified by means of semantically grounded medical knowledge bases, such as the UMLS and its Semantic Network – contained in these sentences, may help to distinguish the different information dimensions of a guideline.*

  The UMLS SN was applied in the IE methods of Chapters 6, 7, and 8. The semantic types of the guideline's clinical concepts supported the filtering of relevant sentences and provided features for the machine learning approach.

- ✓ *The introduction of a weighting coefficient may help to determine the relevance of a condition-action sentence to be included in the formal model.*

  The formula for calculation of the measure relevance rate rr was provided in Chapter 7 together with a scheme for its interpretation.

- ✓ *Specification formalisms for annotating events and temporal relations in narrative texts (e.g., TimeML) from other domains may also be applied to clinical guidelines in order to describe the temporal relations among clinical activities.*

  The TimeML specification built the basis of our GOALS methodology. As a domain independent 'quasi standard' for temporal annotations it is the core concept of our methodology. The scenario-based application of GOALS in Chapter 9 showed the applicability of TimeML for clinical practice guidelines and protocols.

- ✓ *The antecedent and the consequence of a condition-action sentence are temporarily related. This relation may be used for automatic reasoning in order to support the transformation process of a guideline.*

  TimeML provided, amongst other temporal concepts, a specific relation between *EVENTs* describing subordination links (*SLinks*). These links of type 'conditional' combined with the semantic types provided by the UMLS SN were used to identify condition-based activities in a guideline.

- ✓ *If it is possible to extend the specification of TimeML to describe condition-based activities of clinical guidelines in a formal way, a mapping of the temporal model into a guideline model may be conceivable.*

  The expressiveness of TimeML was increased by defining additional attributes for the existing temporal concepts. In Chapter 8 we defined a new 'role' attribute for *EVENT* annotations and extended the value-set of the *SLink*-attribute 'relType'.

The *'checkmarks'* in front of the hypothesis' paragraphs above prove that the research question

Can the temporal relations among condition-based clinical activities be detected automatically by use of IE methods in order to support the manual modeling task of a guideline into its formal representation?

can be answered positively.

10.3 Future Work

The performance of our implemented methods showed that it is possible to support the modeling process of clinical guidelines with information extraction methods. The following list shows the next steps in order to grant further progress in this direction. It is structured according to increasing complexity illustrating the manyfold challenges:

- Independent from the IE methods, a comprehensive corpus of training and testing documents (multiple guidelines and protocols from different clinical domains) has to be created – for intermediate CIG representations as well as for executable CIG formalisms.
- Based on that extended corpus, the multi-label classification can be enhanced in order to improve the classification of the different information dimensions in the guideline documents. Likewise, the hand-crafted IE rules can be improved.
- The IE methods for the identification and modeling of conditional sentences have to be enriched by additional information extraction rules, on the one hand, and the extension of the semantic pattern pool, on the other hand.
- A step-wise integration of the developed methods into a guideline authoring tool will enable a comprehensive evaluation in a real world environment and help to develop user specific adjustments.
- The algorithm for the calculation of the relevance rate rr has to be extended to other linguistic structures. This will also help to extract supplementary information from the guideline (e.g., to determine if a clinical activity is bound to a certain intention).
- GOALS should be applied to an entire protocol or guideline in order to make it possible to define all extensions to the TimeML specification language, followingly published in an annotation guideline.
- Based on that specification, existing tools for the automatic creation of TimeML annotations have to be adapted (possible candidates were discussed in Chapter 4.3).
- IE methods have to be implemented to identify information dimensions which are independent from temporal concepts (e.g., patient related aspects). The possibilities to extend TimeML in this direction has to be analyzed and – if necessary – the annotation guideline expanded.
- The ultimate step is the application of GOALS to produce formalized executable formalisms.

As this whole topic is a highly interdisciplinary challenge, it can only be successfully managed if a professional team of computer linguists, medical experts, knowledge engineers, machine learning experts, software developers, and linguists work together. Only by this way, a fully automatized ‘compilation’ of a guideline into a specific CIG language based on our GOALS methodology is conceivable.

Bibliography

[AHA14] Muawia Abdelmagid, Mubarak Himmat, and Ali Ahmed. Survey on Information Extraction from Chemical Compound Literatures: Techniques and Challenges. *Journal of Theoretical and Applied Information Technology*, 67(2):284–289, Sep 2014.

[AL10] Alan R. Aronson and François-Michel M. Lang. An overview of MetaMap: historical perspective and recent advances. *Journal of the American Medical Informatics Association: JAMIA*, 17(3):229–236, May 2010.

[All83] James F. Allen. Maintaining Knowledge About Temporal Intervals. *Communications of the ACM*, 26(11):832–843, November 1983.

[AM06] Wolfgang Aigner and Silvia Miksch. CareVis: Integrated visualization of computerized protocols and temporal patient data. *Artificial Intelligence in Medicine*, 37(3):203–218, 2006.

[AMBHR13] S. Ait-Mokhtar, B. D. Bruijn, C. Hagege, and P. Rupi. Initial prototype for relation identification between concepts, D3.2., 2013. EURECA Project.

[App99] Douglas E. Appelt. Introduction to Information Extraction. *AI Community*, 12(3):161–172, August 1999.

[Ass08] Association francaise de normalisation and International Organization for Standardization. Language resource management - Semantic annotation framework (SemAf). Part 1., Time and events = Gestion des ressources langagières. Partie 1., Temps et événements, 2008. Draft International Standard.

[Ass13] Association francaise de normalisation and International Organization for Standardization. Language resource management. Semantic annotation framework (SemAf). Time and events SemAF-Time, ISO-TimeML, 2013.

[AZ11] Asma B. Abacha and Pierre Zweigenbaum. Automatic extraction of semantic relations between medical entities: a rule based approach. *Journal of Biomedical Semantics*, 2(Suppl 5):S4+, 2011.

[Bet13] Steven Bethard. ClearTK-TimeML: A minimalist approach to TempEval 2013. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 10–14, Atlanta, Georgia, USA, June 2013. Association for Computational Linguistics.

[BK11] Wei Bi and James Kwok. Multi-Label Classification on Tree- and DAG-Structured Hierarchies. In Lise Getoor and Tobias Scheffer, editors, *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, ICML '11, pages 17–24, New York, USA, June 2011. ACM.

[BKL09] Steven Bird, Ewan Klein, and Edward Loper. *Natural Language Processing with Python — Analyzing Text with the Natural Language Toolkit*. O'Reilly Media, 2009.

[BMK07] Steven Bethard, James H. Martin, and Sara Klingenstein. Finding Temporal Structure in Text: Machine Learning of Syntactic Temporal Relations. *Int. J. Semantic Computing*, 1(4):441–457, 2007.

[BPT+04] A. A. Boxwala, M. Peleg, S. Tu, O. Ogunyemi, Q. T. Zeng, D. Wang, V. L. Patel, R. A. Greenes, and E. H. Shortliffe. GLIF3: A representation format for sharable computer-interpretable clinical practice guidelines. *Journal of Biomedical Informatics*, 37(3):147–161, Jun 2004.

[CCD07] Wendy W. Chapman, David Chu, and John N. Dowling. ConText: An Algorithm for Identifying Contextual Features from Clinical Text. In *Proceedings of the Workshop on BioNLP 2007: Biological, Translational, and Clinical Language Processing*, BioNLP '07, pages 81–88, Stroudsburg, PA, USA, 2007. Association for Computational Linguistics.

[CCQS05] P. Ciccarese, E. Caffi, S. Quaglini, and M. Stefanelli. Architectures and tools for innovative Health Information Systems: the Guide Project. *International Journal of Medical Informatics*, 74(7-8):553–562, Aug 2005.

[CGB08] Jonathan H. Clark and José P. González-Brenes. Coreference resolution: Current trends and future directions, 2008.

[CLR13] Laura Chiticariu, Yunyao Li, and Frederick R. Reiss. Rule-Based Information Extraction is Dead! Long Live Rule-Based Information Extraction Systems! In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 827–832, 2013.

[CMB+11] Hamish Cunningham, Diana Maynard, Kalina Bontcheva, Valentin Tablan, Niraj Aswani, Ian Roberts, Genevieve Gorrell, Adam Funk, Angus Roberts, Danica Damljanovic, Thomas Heitz, Mark A. Greenwood, Horacio Saggion, Johann Petrak, Yaoyong Li, and Wim Peters. *Text Processing with GATE (Version 6)*. GATE, 2011.

[CMBT02] H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan. GATE: A framework and graphical development environment for robust NLP tools and applications. In *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics*, 2002.

[CPH+02] Zong Chen, Yehoshua Perl, Michael Halper, James Geller, and Huanying Gu. Partitioning the UMLS semantic network. *IEEE Transactions on Information Technology in Biomedicine*, 6(2):102–108, 2002.

[CQK02] Paolo Ciccarese, S. Quaglini, and Anand Kumar. NewGuide: a new approach to representing clinical practice guidelines, March 2002.

[CTRB13] Hamish Cunningham, Valentin Tablan, Angus Roberts, and Kalina Bontcheva. Getting More Out of Biomedical Documents with GATE's Full Lifecycle Open Source Text Analytics. *PLOS Computational Biology*, 2013.

[DAB+06] R. H. Dolin, L. Alschuler, S. Boyer, C. Beebe, F. M. Behlen, P. V. Biron, and A. Shabo Shvo. HL7 Clinical Document Architecture, Release 2. *Journal of the American Medical Informatics Association*, 13(1):30–39, 2006.

[dCBKH04] Paul A. de Clercq, Johannes A. Blom, Hendrikus H. M. Korsten, and Arie Hasman. METHODOLOGICAL REVIEW: Approaches for Creating Computer-interpretable Guidelines That Facilitate Decision Support. *Artificial Intelligence in Medicine*, 31(1):1–27, May 2004.

[dCKH08] Paul A. de Clercq, K. Kaiser, and A. Hasman. Computer-Interpretable Guideline formalisms. *Studies in Health Technology and Informatics*, 139:22–43, 2008.

[Den08] K. Denecke. Semantic structuring of and information extraction from medical documents using the UMLS. *Methods of Information in Medicine*, 47(5):425–434, 2008.

[DTY+04] Zhi-Hong Deng, Shi-Wei Tang, Dong-Qing Yang, MingZhangLi-Yu Li, and Kun-Qing Xie. A comparative study on feature weight in text categorization. In JeffreyXu Yu, Xuemin Lin, Hongjun Lu, and Yanchun Zhang, editors, *Advanced Web Technologies and Applications*, volume 3007 of *Lecture Notes in Computer Science*, pages 588–597. Springer Berlin Heidelberg, 2004.

[EFC+11] Oren Etzioni, Anthony Fader, Janara Christensen, Stephen Soderland, and Mausam. Open Information Extraction: The Second Generation.

In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume One*, IJCAI'11, pages 3–10. AAAI Press, 2011.

[FDCBA14] Manuel Fernández-Delgado, Eva Cernadas, Senén Barro, and Dinani Amorim. Do we need hundreds of classifiers to solve real world classification problems? *Journal of Machine Learning Research*, 15:3133–3181, 2014.

[FH98] C. Friedman and G. Hripcsak. Evaluating natural language processors in the clinical domain. *Methods of Information in Medicine*, 37(4-5):334–344, Nov 1998.

[FL90] Marilyn J. Field and Kathleen N. Lohr. *Clinical Practice Guidelines: Directions for a New Program.* The National Academies Press, 1990.

[FL04] David Ferrucci and Adam Lally. UIMA: An Architectural Approach to Unstructured Information Processing in the Corporate Research Environment. *Nat. Lang. Eng.*, 10(3-4):327–348, September 2004.

[FLG+06] David Ferruci, Adam Lally, Daniel Gruhl, Edward Epstein, Marshall Schor, Marshall Schor, William Murdock, Andy Frenkiel, Eric Brown, Thomas Hampp, Yurdaer Doganata, Christopher Welty, Lisa Amini, Galina Kofman, Lev Kozakov, and Yosi Mass. Towards an interoperability standard for text and multi-modal analytics, 2006.

[FMSW01] Lisa Ferro, Inderjeet Mani, Beth Sundheim, and George Wilson. TIDES Temporal Annotation Guidelines, 2001.

[Fou15a] The Apache Software Foundation. Apache cTAKES, 2015. https://cwiki.apache.org/confluence/display/CTAKES/cTAKES+3.1+Component+Use+Guide (last accessed March 16, 2015).

[Fou15b] The Apache Software Foundation. UIMA Unstructured Information Management Architecture, 2015. https://uima.apache.org (last accessed March 16, 2015).

[GMF+03] John H Gennari, Mark A Musen, Ray W Fergerson, William E Grosso, Monica Crubézy, Henrik Eriksson, Natalya F Noy, and Samson W Tu. The evolution of Protégé: an environment for knowledge-based systems development. *International Journal of Human-Computer Studies*, 58(1):89 – 123, 2003.

[Goo12a] Philip G. Gooch. A lightweight, pattern-based approach to identification and formalisation of TimeML expressions in clinical narratives. In *The Sixth Informatics for Integrating Biology and the Bedside (i2b2) Natural Language Processing Challenge for Clinical Records*, 2012.

[Goo12b] Philip G. Gooch. A modular, open-source information extraction framework for identifying clinical concepts and processes of care in clinical narratives, December 2012. PhD thesis, Centre for Health Informatics, School of Informatics, City University London, London, UK.

[GR93] J. M. Grimshaw and I. T. Russell. Effect of clinical guidelines on medical practice: a systematic review of rigorous evaluations. *Lancet*, 342:1317–1322, Nov 1993.

[Gri12] Ralph Grishman. Information Extraction: Capabilities and Challenges, 2012. http://cs.nyu.edu/grishman/tarragona.pdf (last accessed: April 6, 2015).

[GSB05] G. Georg, B. Seroussi, and J. Bouaud. Extending the GEM model to support knowledge extraction from textual guidelines. *International Journal of Medical Informatics*, 74(2-4):79–87, Mar 2005.

[HKMS11] Negin Hajizadeh, Nitu Kashyap, George Michel, and Richard N Shiffman. GEM at 10: a decade's experience with the Guideline Elements Model. *AMIA Annual Symposium proceedings / AMIA Symposium.*, 2011:520–528, 2011.

[HR10] Jerry R. Hobbs and Ellen Riloff. Information extraction. In Nitin Indurkhya and Fred J. Damerau, editors, *Handbook of Natural Language Processing, Second Edition*. CRC Press, Taylor and Francis Group, Boca Raton, FL, 2010. ISBN 978-1420085921.

[HtTvHAM14] Zhisheng Huang, Annette ten Teije, Frank van Harmelen, and Salah Ait-Mokhtar. Semantic representation of evidence-based clinical guidelines. In *Knowledge Representation for Health Care - 6th International Workshop, KR4HC 2014*, pages 78–94, 2014.

[HYSS08] A. Hatsek, O. Young, E. Shalom, and Y. Shahar. DeGeL: a clinical-guidelines library and automated guideline-support tools. *Studies in Health Technology and Informatics*, 139:203–212, 2008.

[IM08] David Isern and Antonio Moreno. Computer-based execution of clinical guidelines: A review. *International Journal of Medical Informatics*, 77(12):787–808, 2008.

[Joa02] T. Joachims. *Learning to Classify Text Using Support Vector Machines – Methods, Theory, and Algorithms*. Kluwer/Springer, 2002.

[KAM07] Katharina Kaiser, Cem Akkaya, and Silvia Miksch. How can information extraction ease formalizing treatment processes in clinical practice guidelines? A method and its evaluation. *Artificial Intelligence in Medicine*, 39(2):151–163, 2007.

[Kat05] Graham Katz. Towards a Denotational Semantics for TimeML. In Frank Schilder, Graham Katz, and James Pustejovsky, editors, *Annotating, Extracting and Reasoning about Time and Events*, volume 4795 of *Lecture Notes in Computer Science*, pages 88–106. Springer, 2005.

[KCK08] J. A. Kim, I. Cho, and Y. Kim. Knowledge translation of SAGE-based guidelines for executing with knowledge engine. *AMIA Annual Symposium Proceedings*, page 1008, 2008.

[Kon14] Natalia Konstantinova. Review of relation extraction methods: What is new out there? In Dmitry I. Ignatov, Mikhail Yu. Khachay, Alexander Panchenko, Natalia Konstantinova, and Rostislav E. Yavorsky, editors, *Analysis of Images, Social Networks and Texts*, volume 436 of *Communications in Computer and Information Science*, pages 15–28. Springer International Publishing, 2014.

[Kot07] S. B. Kotsiantis. Supervised machine learning: A review of classification techniques. In *Proceedings of the 2007 Conference on Emerging Artificial Intelligence Applications in Computer Engineering: Real Word AI Systems with Applications in eHealth, HCI, Information Retrieval and Pervasive Technologies*, pages 3–24, Amsterdam, The Netherlands, 2007. IOS Press.

[KPS05] Graham Katz, James Pustejovsky, and Frank Schilder. 05151 Summary – Annotating, Extracting and Reasoning about Time and Events. In Graham Katz, James Pustejovsky, and Frank Schilder, editors, *Annotating, Extracting and Reasoning about Time and Events*, number 05151 in Dagstuhl Seminar Proceedings, Dagstuhl, Germany, 2005. Internationales Begegnungs- und Forschungszentrum für Informatik (IBFI), Schloss Dagstuhl, Germany.

[KS10] Katharina Kaiser and Andreas Seyfang. Modelling Clinical Guidelines and Protocols in MHB – A Tutorial. Technical report, Institute of Software Technology and Interactive Systems; Vienna University of Technology, March 2010. http://ieg.ifs.tuwien.ac.at/pub/Asgaard-TR-2010-1.pdf (last accessed: April 6, 2015).

[KSM10] Katharina Kaiser, Andreas Seyfang, and Silvia Miksch. Identifying actions described in clinical practice guidelines using semantic relations. In *Proc. of the KR4HC 2010 - 2nd International Workshop on Knowledge Representation for Health-Care in conjunction with the European Conference on Artificial Intelligence (ECAI 2010)*, August 17th 2010.

[KSM11] Katharina Kaiser, Andreas Seyfang, and Silvia Miksch. Identifying treatment activities for modelling computer-interpretable clinical practice guidelines. In David Riaño, Annette Teije, Silvia Miksch, and Mor

Peleg, editors, *Knowledge Representation for Health-Care*, volume 6512 of *Lecture Notes in Computer Science*, pages 114–125. Springer Berlin Heidelberg, 2011.

[LBTvdHH10] A. Latoszek-Berendsen, H. Tange, H. J. van den Herik, and A. Hasman. From clinical practice guidelines to computer-interpretable guidelines. A literature overview. *Methods of Information in Medicine*, 49(6):550–570, 2010.

[LCC+10] Bin Liu, Laura Chiticariu, Vivian Chu, H. V. Jagadish, and Frederick R. Reiss. Automatic rule refinement for information extraction. *Proc. VLDB Endow.*, 3(1-2):588–597, September 2010.

[LHM93] D. Lindberg, B. Humphreys, and A. McCray. The Unified Medical Language System. *Methods of Information in Medicine*, 32(4):281–291, 1993.

[LJLW11] Zhihui Luo, Stephen B. Johnson, Albert M. Lai, and Chunhua Weng. Extracting temporal constraints from clinical research eligibility criteria using conditional random fields. *AMIA Annual Symposium proceedings / AMIA Symposium*, 2011:843–852, 2011.

[Mad07] Nitin Madnani. Getting started on natural language processing with python. *Crossroads*, 13(4):5–5, September 2007.

[MBB01] Alexa T McCray, A Burgun, and O Bodenreider. Aggregating UMLS semantic types for reducing conceptual complexity. *Proceedings of Medinfo*, 10(pt 1):216–20, 2001.

[MBD+13] Timothy Miller, Steven Bethard, Dmitriy Dligach, Sameer Pradhan, Chen Lin, and Guergana Savova. Discovering temporal narrative containers in clinical text. In *Proceedings of the 2013 Workshop on Biomedical Natural Language Processing*, Sofia, Bulgaria, August 2013. Association for Computational Linguistics.

[MH13] Beatrice Moreno and Eric Harder. Neue Wege der IT-gestuetzten Implementierung klinischer Leitlinien. *Professional Process*, 2013.

[Mik99] Silvia Miksch. Plan management in the medical domain. *AI Commun.*, 12(4):209–235, 1999.

[MK13] Anne-Lyse Minard and Katharina Kaiser. Supporting computer-interpretable guidelines' modeling by automatically classifying clinical actions. In David Riaño, Richard Lenz, Silvia Miksch, Mor Peleg, Manfred Reichert, and Annette ten Teije, editors, *KR4HC/ProHealth*, volume 8268 of *Lecture Notes in Computer Science*, pages 39–52. Springer, 2013.

[MKSJ98] Silvia Miksch, Robert Kosara, Yuval Shahar, and Peter D. Johnson. AsbruView: Visualization of Time-Oriented, Skeletal Plans. In Reid G. Simmons, Manuela M. Veloso, and Stephen F. Smith, editors, *AIPS*, pages 11–18. AAAI, 1998.

[MM05] Monika Moser and Silvia Miksch. Improving clinical guideline implementation through prototypical design patterns. In Silvia Miksch, Jim Hunter, and Elpida T. Keravnou, editors, *Artificial Intelligence in Medicine*, volume 3581 of *Lecture Notes in Computer Science*, pages 126–130. Springer Berlin Heidelberg, 2005.

[MnFD+03] Elena Montañés, Javier Fernández, Irene Díaz, Elías F. Combarro, and José Ranilla. Measures of rule quality for feature selection in text categorization. In Michael R. Berthold, Hans-Joachim Lenz, Elizabeth Bradley, Rudolf Kruse, and Christian Borgelt, editors, *IDA*, volume 2810 of *Lecture Notes in Computer Science*, pages 589–598. Springer, 2003.

[MR02] Shaul Markovitch and Dan Rosenstein. Feature Generation Using General Constructor Functions. *Machine Learning Journal*, 49(1):59–98, October 2002.

[MRS08] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, New York, USA, 2008.

[MRvdA10] J. Mendling, H. A. Reijers, and W. M. P. van der Aalst. Seven Process Modeling Guidelines (7PMG). *Information and Software Technology*, 52(2):127–136, February 2010.

[MS99] Christopher D. Manning and Hinrich Schütze. *Foundations of statistical natural language processing*. MIT Press, Cambridge, MA, USA, 1999.

[MTDS96] M. A. Musen, S. W. Tu, A. K. Das, and Y. Shahar. EON: a component-based approach to automation of protocol-directed therapy. *Journal of the American Medical Informatics Association*, 3(6):367–388, 1996.

[MvdAP07] Nataliya Mulyar, Wil M. P. van der Aalst, and Mor Peleg. A Pattern-based Analysis of Clinical Computer-interpretable Guideline Modeling Languages. *Journal of the American Medical Informatics Association*, 14(6):781 – 787, 2007.

[NS07] David Nadeau and Satoshi Sekine. A survey of named entity recognition and classification. *Linguisticae Investigationes*, 30(1):3–26, January 2007. Publisher: John Benjamins Publishing Company.

[OWB08] P. V. Ogren, P. G. Wetzler, and S. J. Bethard. ClearTK: A UIMA Toolkit for Statistical Natural Language Processing. In *LREC*, 2008.

[Pav11] Hemdev Pavan. *Information Extraction: A Smart Calendar Application.* VDM Verlag, Saarbrücken, Germany, 2011.

[PBO+00] M. Peleg, A. A. Boxwala, O. Ogunyemi, Q. Zeng, S. Tu, R. Lacson, E. Bernstam, N. Ash, P. Mork, L. Ohno-Machado, E. H. Shortliffe, and R. A. Greenes. GLIF3: the evolution of a guideline representation format. *Proceedings of the AMIA Symposium*, pages 645–649, 2000.

[PCnI+03] James Pustejovsky, José M. Castaño, Robert Ingria, Roser Saurí, Robert J. Gaizauskas, Andrea Setzer, Graham Katz, and Dragomir R. Radev. TimeML: Robust Specification of Event and Temporal Expressions in Text. In *New Directions in Question Answering*, pages 28–34. AAAI Press, 2003.

[Pel13] M. Peleg. Computer-interpretable clinical guidelines: a methodological review. *Journal of Biomedical Informatics*, 46(4):744–763, Aug 2013.

[PLBR10] James Pustejovsky, Kiyong Lee, Harry Bunt, and Laurent Romary. ISO-TimeML: An International Standard for Semantic Annotation. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, May 2010. European Language Resources Association (ELRA).

[PS11] James Pustejovsky and Amber Stubbs. Increasing informativeness in temporal annotation. In *Proceedings of the Fifth Linguistic Annotation Workshop, LAW 2011*, pages 152–160. Association for Computational Linguistics, 2011.

[PTB+03] Mor Peleg, Samson Tu, J Bury, Paolo Ciccarese, John Fox, R Greenes, Richard Hall, Peter D Johnson, N. Jones, Anand Kumar, Silvia Miksch, Silvana Quaglini, Andreas Seyfang, Edward H Shortliffe, and Mario Stefanelli. Comparing Computer-Interpretable Guideline Models: A Case-Study Approach. *JAMIA*, 10(1):52–68, Jan. - Feb. 2003.

[Pus12] James Pustejovsky. Reasoning about Temporal and Event Information in Clinical Texts, 2012.

[PY13] Jakub Piskorski and Roman Yangarber. Information extraction: Past, present and future. In Thierry Poibeau, Horacio Saggion, Jakub Piskorski, and Roman Yangarber, editors, *Multi-source, Multilingual Information Extraction and Summarization*, Theory and Applications of Natural Language Processing, pages 23–49. Springer Berlin Heidelberg, 2013.

[RBT+04] P. Ram, D. Berg, S. Tu, G. Mansfield, Q. Ye, R. Abarbanel, and N. Beard. Executing clinical practice guidelines using the SAGE execution engine. *Studies in Health Technology and Informatics*, 107(Pt 1):251–255, 2004.

[Rem10] Remine. Documentation of formalized guidelines, 2010. http://www.remine-project.eu/ (last accessed March 4, 2015).

[RFLL12] Preethi Raghavan, Eric Fosler-Lussier, and Albert M. Lai. Temporal Classification of Medical Events. In *Proceedings of the 2012 Workshop on Biomedical Natural Language Processing*, BioNLP '12, pages 29–37, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics.

[RGH+07] Angus Roberts, Robert Gaizauskas, Mark Hepple, Neil Davis, George Demetriou, Yikun Guo, Jay (Subbarao) Kola, Ian Roberts, Andrea Setzer, Archana Tapuria, and Bill Wheeldin. The CLEF Corpus: Semantic Annotation of Clinical Text. *American Medical Informatics Association*, pages 625–629, 2007.

[ROM+13] Ruth M. Reeves, Ferdo R. Ong, Michael E. Matheny, Joshua C. Denny, Dominik Aronsky, Glenn T. Gobbel, Diane Montella, Theodore Speroff, and Steven H. Brown. Detecting temporal expressions in medical narratives. *International Journal of Medical Informatics*, 82(2):118–127, 2013.

[RP06] Romesh Ranawana and Vasile Palade. Multi-Classifier Systems: Review and a Roadmap for Developers. *International Journal of Hybrid Intelligent Systems*, 3(1):35–61, January 2006.

[SBF+14] William F. Styler, Steven Bethard, Sean Finan, Martha Palmer, Sameer Pradhan, Piet C. de Groen, Brad Erickson, Timothy Miller, Chen Lin, Guergana Savova, and James Pustejovsky. Temporal Annotation in the Clinical Domain, 2014.

[SBS+09] Guergana Savova, Steven Bethard, Will Styler, James Martin, Martha Palmer, James Masanz, and Wayne Ward. Towards temporal relation discovery from the clinical narrative. *AMIA Annual Symposium proceedings*, 2009:568–572, 2009.

[Sco11] Scottish Intercollegiate Guidelines Network (SIGN). British Guideline on the Management of Asthma. A Clinical National Guideline SIGN 63, Scottish Intercollegiate Guidelines Network (SIGN), Edinburgh (Scotland), May 2011.

[Set01] Andrea Setzer. Temporal Information in Newswire Articles: An Annotation Scheme and Corpus Study., 2001.

[SF03] David R. Sutton and John Fox. The syntax and semantics of the PRO*forma* guideline modeling language. *Journal of the American Medical Informatics Association*, 10(5):433 – 443, 2003.

[SGL04] S. Skonetzki, H. J. Gausepohl, and O. Linderkamp. Evolution of the HELEN Representation for Managing Clinical Practice Guidelines, November 2004.

[SGvdH+04] S. Skonetzki, H. J. Gausepohl, M. van der Haak, S. Knaebel, O. Linderkamp, and T. Wetter. HELEN, a modular framework for representing and implementing clinical practice guidelines. *Methods of Information in Medicine*, 43(4):413–426, 2004.

[SH06] F. A. Sonnenberg and C. G. Hagerty. Computer-interpretable clinical practice guidelines. Where are we and where are we going ? *Yearbook of Medical Informatics*, pages 145–158, 2006.

[SH10] Amber Stubbs and Benjamin Harshfield. Applying the TARSQI Toolkit to Augment Text Mining of EHRs. In *Proceedings of the 2010 Workshop on Biomedical Natural Language Processing*, BioNLP '10, pages 141–143, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[SHWA03] Mihai Surdeanu, Sanda Harabagiu, John Williams, and Paul Aarseth. Using predicate-argument structures for information extraction. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*, ACL '03, pages 8–15, Stroudsburg, PA, USA, 2003. Association for Computational Linguistics.

[SK11] Andreas Seyfang and Katharina Kaiser. MHB-F specification, 2011.

[SKA+00] R. N. Shiffman, B. T. Karras, A. Agrawal, R. Chen, L. Marenco, and S. Nath. GEM: a proposal for a more comprehensive guideline document model using XML. *Journal of the American Medical Informatics Association*, 7(5):488–498, 2000.

[SLG+06] Roser Saurí, Jessica Littman, Robert Gaizauskas, Andrea Setzer, and James Pustejovsky. TimeML Annotation Guidelines, Version 1.2.1, 2006.

[SM10] Charles Sutton and Andrew McCallum. An Introduction to Conditional Random Fields, November 2010.

[SMET04] R. N. Shiffman, G. Michel, A. Essaihi, and E. Thornquist. Bridging the guideline implementation gap: a systematic, document-centered approach to guideline implementation. *Journal of the American Medical Informatics Association*, 11(5):418–426, 2004.

[SMJ98] Yuval Shahar, Silvia Miksch, and Peter D. Johnson. The Asgaard project: a task-specific framework for the application and critiquing of time-oriented clinical guidelines. *Artificial Intelligence in Medicine*, 14(1-2):29–51, 1998.

[SMM+06] Andreas Seyfang, Silvia Miksch, Mar Marcos, Jolanda Wittenberg, Cristina Polo-Conde, and Kitty Rosenbrand. Bridging the gap between informal and formal guideline representations. In *European Conference on Artificial Intelligence (ECAI-2006)*, volume 141, pages 447–451, Riva del Garda, Italy, 2006. IOS Press.

[SMO+10] Guergana K. Savova, James J. Masanz, Philip V. Ogren, Jiaping Zheng, Sunghwan Sohn, Karin C. Kipper-Schuler, and Christopher G. Chute. Mayo clinical Text Analysis and Knowledge Extraction System (cTAKES): architecture, component evaluation and applications. *Journal of the American Medical Informatics Association : JAMIA*, 17(5):507–513, September 2010.

[SMPC+05] Andreas Seyfang, Silvia Miksch, Cristina Polo-Conde, Jolanda Wittenberg, Mar Marcos, and Kitty Rosenbrand. MHB–A Many-Headed Bridge Between Informal and Formal Guideline Representations. In Silvia Miksch, Jim Hunter, and Elpida Keravnou, editors, *Proceedings of the 10th Conference on Artificial Intelligence in Medicine (AIME 2005)*, page 146–150. Springer, 2005.

[SMV04] Andreas Seyfang, Silvia Miksch, and Peter Votruba. Specification of Formats of Intermediate, Asbru and KIV Representations, November 2004. http://www.keg.uji.es/deliverables/D22a-specification-of-formatsR1.pdf (last accessed: February 22, 2015).

[SOBG03] M. Sordo, O. Ogunyemi, A. A. Boxwala, and R. A. Greenes. GELLO: an object-oriented query and expression language for clinical decision support. *AMIA Annual Symposium Proceedings*, page 1012, 2003.

[SRU13] Weiyi Sun, Anna Rumshisky, and Ozlem Uzuner. Temporal reasoning over clinical text: the state of the art. *JAMIA*, 20(5):814–819, September 2013.

[SSTM+08] Erez Shalom, Yuval Shahar, Meirav Taieb-Maimon, Guy Bar, Avi Yarkoni, Ohad Young, Susana B. Martins, Laszlo Vaszar, Mary K. Goldstein, Yair Liel, Akiva Leibowitz, Tal Marom, and Eitan Lunenfeld. A Quantitative Assessment of a Methodology for Collaborative Specification and Evaluation of Clinical Guidelines. *J. of Biomedical Informatics*, 41(6):889–903, December 2008.

[StTvH+07] Radu Serban, Annette ten Teije, Frank van Harmelen, Mar Marcos, and Cristina Polo-Conde. Extraction and use of linguistic patterns for modelling medical guidelines. *Artificial Intelligence in Medicine*, 39(2):137–149, February 2007.

[SVP06] Roser Saurí, Marc Verhagen, and James Pustejovsky. SlinkET: A partial modal parser for events. In *Language Resources and Evaluation Conference*, 2006.

[SWL+13] Sunghwan Sohn, Kavishwar B. Wagholikar, Dingcheng Li, Siddhartha R. Jonnalagadda, Cui Tao, Ravikumar Komandur Elayavilli, and Hongfang Liu. Comprehensive temporal information detection from clinical text: medical events, time, and TLINK identification. *Journal of the American Medical Informatics Association*, 20(5):836–842, April 2013.

[TCG+07] S. W. Tu, J. R. Campbell, J. Glasgow, M. A. Nyman, R. McClure, J. McClay, C. Parker, K. M. Hrabak, D. Berg, T. Weida, J. G. Mansfield, M. A. Musen, and R. M. Abarbanel. The SAGE Guideline Model: achievements and overview. *Journal of the American Medical Informatics Association*, 14(5):589–598, 2007.

[TCM+13] Camilo Thorne, Elena Cardillo, Marco Montali, Claudio Eccher, and Diego Calvanese. Process fragment recognition in clinical documents. In *Proceedings of the 13th Conference of the Italian Association for Artificial Intelligence (AI*IA 2013)*, 2013.

[TGS08] P. Terenziani, E. German, and Y. Shahar. The temporal aspects of clinical guidelines. *Studies in Health Technology and Informatics*, 139:81–100, 2008.

[TMB+04] P. Terenziani, S. Montani, A. Bottrighi, M. Torchio, G. Molino, and G. Correndo. The GLARE approach to clinical guidelines: main features. *Studies in Health Technology and Informatics*, 101:162–166, 2004.

[TMC+13] Camilo Thorne, Marco Montali, Diego Calvanese, Elena Cardillo, and Claudio Eccher. The VeriCliG Project: Extraction of Computer Interpretable Guidelines via Syntactic and Semantic Annotation. In *Proc. of the Workshop on Computational Semantics in Clinical Text (CSCT 2013)*, pages 54–58, 2013.

[TMKS09] N. Tran, G. Michel, M. Krauthammer, and R. N. Shiffman. Embedding the guideline elements model in web ontology language. *AMIA Annual Symposium Proceedings*, 2009:645–649, 2009.

[TMRn+10] María Taboada, María Meizoso, David Riaño, Albert Alonso, and Diego Martínez. From natural language descriptions in clinical guidelines to relationships in an ontology. In *Proceedings of the 2009 AIME international conference on Knowledge Representation for Health-Care: Data, Processes and Guidelines*, KR4HC'09, pages 26–37, Berlin, Heidelberg, 2010. Springer-Verlag.

[TMT01] P. Terenziani, G. Molino, and M. Torchio. A modular approach for representing and executing clinical guidelines. *Artificial Intelligence in Medicine*, 23(3):249–276, Nov 2001.

[TPC+11] S. W. Tu, M. Peleg, S. Carini, M. Bobak, J. Ross, D. Rubin, and I. Sim. A practical method for transforming free-text eligibility criteria into computable criteria. *Journal of Biomedical Informatics*, 44(2):239–250, Apr 2011.

[UA10a] Naushad UzZaman and James Allen. TRIOS TimeBank Corpus: Extended TimeBank Corpus with Help of Deep Understanding of Text. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, May 2010. European Language Resources Association (ELRA).

[UA10b] Naushad UzZaman and James F. Allen. Extracting events and temporal expressions from text. In *ICSC*, pages 1–8. IEEE, 2010.

[UGLK08] O. Uzuner, I. Goldstein, Y. Luo, and I. Kohane. Identifying patient smoking status from medical discharge records. *Journal of the American Medical Informatics Association*, 15(1):14–24, 2008.

[Uzu09] O. Uzuner. Recognizing obesity and comorbidities in sparse data. *Journal of the American Medical Informatics Association*, 16(4):561–570, 2009.

[Ver12] Marc Verhagen. The Tarsqi Toolkit Temporal Modeling for the news domain and the medical domain, October 2012. Presentation to the AMIA NLP Working Group, Washington.

[VMK04] Peter Votruba, Silvia Miksch, and Robert Kosara. Tracing the Formalization Steps of Textual Guidelines. In *Computer-based Support for Clinical Guidelines and Protocols. Proceedings of the Symposium on Computerized Guidelines and Protocols (CGP 2004)*, volume 101 of *Studies in Health Technology and Informatics*, page 172–176. IOS Press, 2004.

[VP08] Marc Verhagen and James Pustejovsky. Temporal Processing with the TARSQI Toolkit. In *COLING (Demos)*, pages 189–192, 2008.

[VP12] Marc Verhagen and James Pustejovsky. The TARSQI Toolkit. In *LREC*, pages 2043–2048. European Language Resources Association (ELRA), 2012.

[VSS+08] Celine Vens, Jan Struyf, Leander Schietgat, Sašo Džeroski, and Hendrik Blockeel. Decision trees for hierarchical multi-label classification. *Journal of Machine Learning Research*, 73(2):185–214, November 2008.

[WK13] Reinhardt Wenzina and Katharina Kaiser. Identifying Condition-Action Sentences Using a Heuristic-based Information Extraction Method. In *Proceedings of the Joint International Workshop: KR4HC'13+ProHealth'13*, pages 17–29, 2013.

[WK14a] Reinhardt Wenzina and Katharina Kaiser. Towards the Application of TimeML in Clinical Guidelines. In Michael Heß and Hannes Schlieter, editors, *Modellierung im Gesundheitswesen. Tagungsband des Workshops im Rahmen der Modellierung 2014*, pages 37–48, 57, 2014. ICB-Research Report.

[WK14b] Reinhardt Wenzina and Katharina Kaiser. Using TimeML to Support the Modeling of Computerized Clinical Guidelines. *Studies in Health Technology and Informatics*, 205:8–12, 2014.

[WK15] Reinhardt Wenzina and Katharina Kaiser. GOALS - Modeling Clinical Guidelines Based on TimeML Concepts, May 2015. Accepted at the 5th International Conference on Digital Health 2015. http://dx.doi.org/10.1145/2750511.2750520.

[WKRQ+08] Xindong Wu, Vipin Kumar, J. Ross Quinlan, Joydeep Ghosh, Qiang Yang, Hiroshi Motoda, GeoffreyJ McLachlan, Angus Ng, Bing Liu, PhilipS Yu, Zhi-Hua Zhou, Michael Steinbach, DavidJ Hand, and Dan Steinberg. Top 10 algorithms in data mining. *Knowledge and Information Systems*, 14(1):1–37, January 2008.

[WLCF+94] W. W. Lehnert, C. Cardie, D. Fisher, J. McCarthy, E. Riloff, and S. Soderland. Evaluating an Information Extraction System. *Integrated Computer-Aided Engineering*, 1(6):453–472, 1194.

[WS02] D. Wang and E. H. Shortliffe. GLEE–a model-driven execution system for computer-based implementation of clinical practice guidelines. *Proceedings of the AMIA Symposium*, pages 855–859, 2002.

[YL04] Lei Yu and Huan Liu. Efficient Feature Selection via Analysis of Relevance and Redundancy. *Journal of Machine Learning Research*, 5:1205–1224, December 2004.

[YSL+07] O. Young, Y. Shahar, Y. Liel, E. Lunenfeld, G. Bar, E. Shalom, S. B. Martins, L. T. Vaszar, T. Marom, and M. K. Goldstein. Runtime application of Hybrid-Asbru clinical guidelines. *Journal of Biomedical Informatics*, 40(5):507–526, Oct 2007.

[ZPH08] Li Zhou, Simon Parsons, and George Hripcsak. Research Paper: The Evaluation of a Temporal Reasoning System in Processing Clinical Discharge Summaries. *JAMIA*, 15(1):99–106, 2008.

[ZPJH10] Kai Zheng, Rema Padman, Michael P. Johnson, and Sharique Hasan. Guideline Representation Ontologies for Evidence-Based Medicine Practice. In Khalil Khoumbati, Yogesh K. Dwivedi, Aradhana Srivastava, and Banita Lal Lal, editors, *Handbook of Research on Advances in Health Informatics and Electronic Healthcare Applications: Global Adoption and Impact of Information Communication Technologies*, pages 234–262. IGI Global, 2010.

Curriculum Vitae

Address	Reinhardt Wenzina Wilhelm Mantlergasse 608 A-3571 Gars am Kamp, Austria
Date of Birth	May 6th, 1964
Education	2011 to 2015 PhD Studies in Information Science at the Vienna University of Technology 2003 to 2005 Diploma Studies in Economics (post-graduate) at the University of Applied Science Hamburg 1982 to 1987 Diploma Studies in Information Science at the Vienna University of Technology
Job Experience	since 2009 Lecturer at the New Design University St. Pölten – Austria Joint development of M.Sc. programs for - Information Science & Big Data Analytics in 2015 - Intellectual Property Rights & Innovations in 2011 since 2005 Teacher at the Department of Information Technology HTL Krems, Lower Austria 2009 to 2010 Research assistant at Krems Research Forschungs-GmbH Krems/Donau, Austria

Job Experience	2000 to 2005 Head of Department PXP Waidhofen PXP Austria GmbH 1988 to 2000 CIO and Head of Controlling Buhl Bauunternehmung GmbH Gars/Kamp, Austria
Professional Memberships	since 2011 Member of the Academic Association of Applied Research
Publications	Reinhardt Wenzina and Katharina Kaiser. Identifying Condition-Action Sentences Using a Heuristic-based Information Extraction Method. In *Proceedings of the Joint International Workshop: KR4HC'13+ProHealth'13*, pages 17–29, 2013. Reinhardt Wenzina and Katharina Kaiser. Towards the Application of TimeML in Clinical Guidelines. In *Modellierung im Gesundheitswesen*. Tagungsband des Workshops im Rahmen der Modellierung 2014, pages 37–48, 2014. ICB-Research Report. Reinhardt Wenzina and Katharina Kaiser. Using TimeML to Support the Modeling of Computerized Clinical Guidelines. *Studies in Health Technology and Informatics*, 205:8–12, 2014. Reinhardt Wenzina and Katharina Kaiser. GOALS - Modeling Clinical Guidelines Based on TimeML Concepts, May 2015. Accepted at the 5th International Conference on Digital Health 2015. http://dx.doi.org/10.1145/2750511.2750520.