Clinical Information Extraction: Lowering the Barrier

Angus Roberts

Submitted in Partial Fulfilment of the
Requirements for the Degree of
Doctor of Philosophy

The University of Sheffield
Sheffield S10 2TN
United Kingdom

Department of Computer Science

October 2012

Abstract

Electronic Patient Records have opened up the possibility of re-using the data collected for clinical practice, to support both clinical practice itself, and clinical research. In order to achieve this re-use, we have to address the issue that most Electronic Patient Records make heavy use of narrative text. This thesis reports an approach to automatically extract clinically significant information from the textual component of the medical record, in order to support re-use of that record. The cost of developing such information extraction systems is currently seen to be a barrier to their deployment. We explore ways of lowering this barrier, through the separation of the linguistic, medical and engineering knowledge and skills required for development.

We describe a rigorous methodology for the construction of a corpus of clinical texts semantically annotated by medical experts, and its use to automatically train a supervised machine learning-based information extraction system. We explore the re-use of existing medical knowledge in the form of terminologies, and present a way in which these terminologies can be coupled with supervised machine learning for information extraction. Finally, we consider the extent to which pre-existing software components can be used to construct a clinical IE system, and build a system that is capable of extracting clinical concepts, their properties, and the relationships between them.

The resulting system shows that it is possible to achieve separation of linguistic, medical and engineering knowledge in clinical information extraction. We find that existing software frameworks are capable of some aspects of information extraction with little additional engineering work, but that they are not mature enough for the construction of a full system by the non-expert. We also find that a new cost is introduced in separating domain and linguistic knowledge, that of manual annotation by domain experts.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

ACE	Automatic Content Extraction (research programme)
ADR	Adverse Drug Reaction
API	Application Programming Interface
BE	Begin / End (defining a region of text)
CIAA	Corrected Inter Annotator Agreement
CLEF	CLinical E-Science Framework
CRIS	Case Register Interactive Search tool
CUI	Concept Unique Identifier (UMLS concept identifier)
DICOM	Digital Imaging and Communications in Medicine
EHR	Electronic Health Record
EPR	Electronic Patient Record
F1	F-measure (evaluation metric)
FBC	Full Blood Count (laboratory test)
FN	False Negative
FP	False Positive
FSR	Finite State Recogniser
GATE	General Architecture for Text Engineering
GP	General Practitioner, UK Primary Care Physician
I2B2	Informatics for Integrating Biology and the Bedside
IAA	Inter Annotator Agreement
ICD	International Classification of Diseases
ID	Identifier
IE	Information Extraction
IR	Information Retrieval
LSP	Linguistic String Project
ML	Machine Learning
MREC	Multi-centre Research Ethics Committee
MUC	Message Understanding Conference
NER	Named Entity Recognition

NHS	UK National Health Service
NIST	National Institute of Standards and Technology
NLM	US National Library of Medicine
NLP	Natural Language Processing
OHNLP	Open Health Natural Language Processing (consortium)
PASTA	Protein Active Site Template Acquisition (project)
POS	Part-Of-Speech
P	Precision (evaluation metric)
RMH	Royal Marsden Hospital
R	Recall (evaluation metric)
SLAM	South London and Maudsley NHS Foundation Trust
TP	True Positive
UIMA	Unstructured Information Management Architecture
UMLSKS	UMLS Knowledge Source Server
UMLS	Unified Medical Language System
U&E	Urea and Electrolytes (laboratory test)
VA	Veterans Administration (US health organisation)
WHO	World Health Organisation
XML	eXtensible Markup Language

Preface

Chapters 2, 3, and 4 of this thesis consist of previously published research papers reproduced in full. Details of copyright, permission to reproduce, and the author's contribution to the research, is given in forewords to these chapters.

Funding

The author of this thesis was initially funded by a UK Medical Research Council (MRC) postgraduate studentship, and was later employed on the MRC Clinical E-Science Framework project (CLEF) (Rector et al., 2003). The research reported was carried out as part CLEF, which was funded by MRC grant reference GO300607, "CLEF: a Clinical E-Science Framework", and grant reference RB106367, "CLEF Services". The author also received a travel grant from the Association of Computational Linguistics (ACL), to participate in the ACL 2005 Student Research Workshop.

Confidentiality

The reported research extracts information from a large corpus of patient records available within the CLEF project. Approval to use this corpus for research purposes within CLEF was sought and obtained from the Thames Valley Multi-centre Research Ethics Committee (MREC). The corpus was handled in accordance with confidentiality guidelines laid down by the Medical Research Council (Medical Research Council, 2000), and those developed by the CLEF consortium (Kalra et al., 2003). No portions of the corpus and no personal data from the corpus are published within this thesis.

Acknowledgements

The author thanks his supervisor, Robert Gaizauskas, and the other members of his PhD panel, Mark Hepple and Mahesan Niranjan, for comments and suggestions; members of the University of Sheffield Natural Language Processing group for useful discussions; members of the University of Manchester Medical Informatics Group for inspiration; the Royal Marsden Hospital for providing the corpus used in the research, and staff at the University of Manchester and University College London for annotating the corpus; staff and clinicians at the Biomedical Research Centre at the South London and Maudsley NHS Trust (SLAM) for the opportunity to further develop the ideas in this thesis; other members of the University of Sheffield GATE Team working with the SLAM data; the numerous developers who have contributed to the GATE software used in the research; Mark Greenwood for the initial LaTeX 2_ε template used for typesetting this thesis; and Mandy, Ben and Bella for the time and space in which to complete the work.

Chapter 1

Clinical Information Extraction: Lowering the Barrier

1.1 Introduction

1.1.1 Motivation

Writing in 2002, Johann van der Lei imagined a virtuous circle in which medical records of the past would improve medical practice of the future:

> Each patient-physician encounter, each investigation, each laboratory test, and each treatment in medical practice constitutes, in principle, an experiment. Ideally, we learn from each experiment (van der Lei, 2002, page 54).

In common with many proponents of the Electronic Patient Record (EPR), Van der Lei argued that it is this electronic record of routine practice that gives us the potential to "close the loop" (van der Lei, 2002, page 54) between clinical practice and research. Holding the patient record electronically allows us to re-use the data for other purposes (van der Lei, 2002). Re-use is not, unfortunately, as simple as opening up the records and pouring out the data. A 2008 review of the structure and content of Electronic Health Records (EHRs) [1] concluded that "most EHRs are still primarily based on narrative text" (Hayrinen et al., 2008, page 300). Closing van der Lei's loop with the current EPR is dependent on us extracting the data from the narrative of the patient record. This is the motivation for the research reported in this thesis.

[1] EHRs are usually distinguished from EPRs in that they can be used across multiple health care institutions.

1.1.2 Problem statement

Natural Language Processing (NLP), the computerised processing of human language, is frequently suggested as a way to re-use the narrative of the patient record (Nadkarni et al., 2011). The idea has common currency in medicine: it is discussed in general medical journals (see, for example, Jha (2011)), and NLP technologies have found their way into the marketing material from major EPR vendors (eClinicalWorks, 2012b).

The use of NLP techniques to extract structured information from unstructured text is known as Information Extraction (IE). It has a long history of research and of use with medical records, as reviewed most recently by Meystre et al. (2008). Information Extraction, however, especially in the medical domain, is expensive. As Chapman et al. say:

> Currently, the perceived cost of applying NLP outweighs the perceived benefit. Deploying an NLP system typically requires a substantial amount of time from an expert NLP developer – normally applications do not generalize and must be rebuilt, retrained, enhanced and re-evaluated for each new task (Chapman et al., 2011, page 541)

Costs arise not just because we need to carry out some complicated software engineering task or system configuration. IE also requires large volumes of high quality, manually annotated example text (Meystre et al., 2008; Chapman et al., 2011; Xia and Yetisgen-Yildiz, 2012) [2]. This is used to clarify requirements, create gold standards with which to assess performance, and to provide resources with which to develop or train IE systems (Roberts et al., 2009; Xia and Yetisgen-Yildiz, 2012).

The high cost of annotation is exacerbated by the medical domain itself. The language used is rich with specialist terminology, telegraphese, abbreviations, and neologisms. Text style and terminology may also be quite local in nature, with individual units evolving their own forms of expression. Consequently, good manual annotation requires the skills of the very people writing the records in the first place – clinicians (Chapman et al., 2008; Xia and Yetisgen-Yildiz, 2012; Scott et al., 2012) [3].

This, then is the problem tackled by this thesis: closing van der Lei's loop with IE is expensive, because it requires tailoring of software, and skilled annotation of large numbers of examples. Can this barrier be lowered?

[2] By *annotated text*, we mean text in which examples of the phenomenon being considered are marked or highlighted in some way.

[3] Although this has recently been disputed in a heated debate on the BioNLP mailing list (BioNLP mailing list, 2012).

1.1.3 Aims and objectives

There are three major costs in building an IE system: the cost of encoding linguistic knowledge required by the application, such as knowledge about syntax and grammatical structure; the cost of encoding domain knowledge, such as knowledge about the terminology and facts of the domain; and the cost of the software engineering effort required to build the system. These three costs are not always mutually exclusive. For example, the encoding of domain knowledge has not always been cleanly separated from the software engineering of an IE system, and recognising this, some effort has gone into making components of an IE system easy to port between domains (Cowie and Lehnert, 1996; Grishman and Sundheim, 1996). Similarly, the distinction between domain knowledge and linguistic knowledge is not always clear-cut.

The aim of the research reported in this thesis is to lower the barrier to building clinical IE systems (that is, IE systems that operate over the text of the medical record). Our objectives examine potential ways of lowering costs, through the separation of linguistic, domain and engineering knowledge, and through the maximal re-use of pre-existing linguistic, domain and software resources. There are three specific objectives.

1. To adopt a supervised machine learning (ML) [4] approach to clinical text, in which models of the text are trained from human annotated example documents, in order that these models may then be applied to unseen texts. The objective is to use domain experts to provide a corpus of examples that capture the semantics of medical language: the meaning of medical terms, and the relationship between these terms in the text. In meeting this objective, a methodology for creating such a corpus, and metrics for assessing its quality will be developed. The application of supervised ML to a full clinical IE system is novel.

2. To examine the use of pre-existing medical terminologies and knowledge resources in clinical IE. Medicine has a large and rich collection of such resources, due in part to the depth and breadth of medical knowledge, and in part the compositional nature of medical terminology. These resources encode knowledge of the domain. Our research question is: can these resources be successfully coupled with supervised ML, to enhance its performance?

3. To build a clinical IE system using "off the shelf" NLP and ML frameworks. These frameworks are intended to ease the task of system development, by providing reusable infrastructures, and by delivering linguistic knowledge in the shape of ready

[4] A *machine learning* algorithm learns a model of some phenomenon from a set of examples. This model can consequently be applied to unseen examples, in order to predict some unknown characteristics.

to use components – for example knowledge of grammatical structure and parts of speech. We ask if such frameworks are sufficiently advanced that the construction of a clinical IE system can become a software engineering task, or even an end-user task.

It is our aim that by separating out the different types of knowledge and skills required to built a clinical IE system, we will maximise the effectiveness of domain experts, linguists and software engineers, and also maximise the re-use of pre-existing components. It is the hope that this will lower the barrier to creating a clinical IE system, as well as increasing the robustness and portability of the system.

1.1.4 Structure of the thesis

This introductory chapter gives background and context to the research, summarises the three research papers that form the body of the thesis, and draws conclusions. Section 1.2 looks at the prevalence of text in the medical record, explores questions of why text is used, and asks whether data held electronically in medical records can be reused. Next, in Section 1.3, IE, the technology used in the research to extract information from the textual portion of the EPR is introduced, giving a history, and a descriptive landscape of work in the area.

The specialisation of information extraction for medical record text is discussed in Section 1.4. A major theme in current research, and one explored by this thesis, is the construction of gold standard annotated corpora of clinical text. These are used to help clarify and focus requirements, to learn statistical models of the text for IE, and to evaluate these models. Recent work on clinical text corpora is reviewed in Section 1.5.

Section 1.6 introduces the Clinical E-Science Framework project (CLEF) (Rector et al., 2003). A gold standard corpus was constructed within CLEF, and the corpus used to drive the construction of an IE system. It is these that provide the research material for this thesis. The section describes the historical background and evolution of CLEF IE system, and places the research into the landscape introduced in previous sections. Finally, Section 1.7 of this introduction states the contributions made by this research.

Following the introduction, Chapters 2, 3 and 4 present the construction of the CLEF gold standard, and experiments on its use in developing the CLEF IE system. Chapter 2 (previously published as Roberts et al. (2009)), details the CLEF corpus, the development of annotation requirements, and the methodology used to create a gold standard for clinical IE. The chapter gives measures of gold standard quality and annotator performance. A mock example of a letter from the corpus is given in Appendix A. The annotation guidelines used in creating the gold standard are given in Appendices B and C. Chapter 3

(previously published as Roberts et al. (2008c)) looks at extraction of *entities* – occurrences of basic clinical concepts from text, such as diseases and anatomical locations. This forms the basis of the CLEF IE system. The chapter examines the use of dictionary look-up of terms, and the use of ML methods, for entity extraction, making use of standard components in an ML and IE framework. Entity extraction is evaluated against the gold standard. Chapter 4 (previously published as Roberts et al. (2008d)) uses the same ML and IE framework, to learn relations between the entities, such as the presence of a disease at a particular anatomical location. The utilities of various types of linguistic knowledge for this task are examined. Finally, in Chapter 5, we discuss the objectives of this thesis in the light of the research presented, and in the light of ongoing work.

1.2 Text in the medical record

Clinical text is the textual portion of a medical record. A medical record typically consists of both structured data (such as laboratory test results, drug prescriptions), and unstructured natural language text. In the UK hospital setting (from where all of the data used for this thesis was derived), the textual portion consists mainly of letters and discharge summaries sent from hospital physicians to primary care physicians, and investigation reports. These texts are often dictated by a clinician, and typed by a medical secretary. Table 1.1 analyses the types of letters from a sample taken from a large oncology hospital, and as analysed further in Chapters 2, 3 and 4. A mock example of such a letter is given in Appendix A.

Type	**Count**	**%**	**Notes**
GP Letter	118002	48.6	Letter to primary physician
Discharge Summary	41275	17.0	Sent to primary physician on discharge
Case Note Only	35771	14.7	Unspecified note
Other Letter	16722	6.9	Unspecified letter
Other Consultant	14011	5.8	Letter to some other consultant physician
Letter to referring Doctor	8642	3.6	Letter to the doctor referring the patient
Patient Letter	6329	2.6	Letter to the patient
Medical Report	1758	0.7	To external body, such as insurance company
Audit Meeting	512	0.2	Report of internal audit
Total	243022		

Table 1.1: Analysis of medical record letters at the Royal Marsden Hospital, by type of letter, using the hospital's own classification

Letters and discharge summaries sent to the General Practitioner (GP, the primary care physician) predominate, accounting for more than 65% of documents. This is typical: there is nothing unusual about the hospital from which this sample was taken. The text of these letters is often considered by clinicians to be the primary record. In the UK, it is the record that will go into the patient's notes kept with their primary physician, and it is

the record that would be used in court. Many clinicians will refer to the last letter at the beginning of a patient consultation, expecting it to summarise the case so far. Hospital physicians therefore have a vested interest in maintaining this textual record.

1.2.1 Text in the electronic patient record

It is unusual to find a hospital ward or outpatient consulting room in the UK without a computer, invariably with access to EPR software with which the clinician can record details of the patient. Many developed countries have invested large amounts into Health IT infrastructure over the past decade, and these infrastructures depend on electronic recording of patient information. In the UK, the most recent work on this is the NHS National Programme for IT (NHS, 2012), the costs of which are disputed but are probably in the region of £20 billion (Carvel, 2006). In the USA, the American Recovery and Reinvestment Act of 2009 (United States Government, 2009) is putting $25.8 billion into Health Information Technology, to "modernize the health care system by promoting and expanding the adoption of health information technology by 2014" (Savel and Foldy, 2012, page 22).

It might be expected that this investment would have led to greater recording of the EPR as structured data. Full text letters to the primary care physician, however, remain the major part of the UK medical record. For example, the figures given in Table 1.1 were taken from a hospital with a strong history of EPR use. Additionally, other forms of text are increasingly found, such as free-text notes of patient-clinician encounters, and free text fields on data entry forms. Unlike the dictated and transcribed letters, these texts are usually entered directly by the clinician.

In the USA, a 2008 review of literature on EHRs concludes that "Most EHRs are still primarily based on narrative text" (Hayrinen et al., 2008, page 300). Many EHRs contain facilities for both structured and unstructured data entry. While no study has compared what is entered into the text to what is entered into the structured record, some do suggest that the use of unstructured text is more popular than structured, coded lists. Simon et al. (2009) for example, in a study of 234 clinicians in Massachusetts, found that 78% of physicians use visit notes (a typically free text part of EPRs) extensively, while 57% use problem lists (a typically coded data part of EPRs) extensively. Menachemi et al. (2006) found similar results. The largest provider of EPR software in the United States is eClinicalWorks. Their main product, EMR, is text-centric, with text-based data entry via voice recognition at its core (eClinicalWorks, 2012a).

1.2.2 Why do clinicians prefer text?

Meystre et al. (2008) note that free text is "convenient to express concepts and events" (Meystre et al., 2008, page 128), but that it is difficult for re-use in other applications, and difficult for statistical analysis. Rosenbloom et al. (2011) have reviewed the few studies that look at the expressivity of structured clinical documentation systems compared to natural prose notes, and report that prose is more accurate, reliable and understandable. Powsner et al. (1998, page 1618) refer to structured data as freezing clinical language, and restricting what may be said. Greenhalgh et al. (2009) referring to the free text of the paper record, say that it is tolerant of ambiguity, which supports the complexity of clinical practice. Much of medical language is hedged with ambiguity and probability, which is difficult to represent as structured data. Scott et al. (2012) examines the uncertainty expressed by this hedging, giving examples from radiology reports.

Other authors refer not to the technical challenges of text, but rather to a difficulty in changing the behaviour of clinicians, away from free text recording to structured data entry (Schleyer, 2008). This is perhaps a manifestation of the wider organisational problem of introducing any EPR system into healthcare organisations (Greenhalgh et al., 2009, page 751).

There is therefore a tension between data re-use, the entry of data in the structured record, and dictation into unstructured text. If we wish to benefit from automatic patient records for audit, research, and for decision support, then we need to maximise the structured data in the medical record. On the other hand, clinicians prefer to use unstructured text (Powsner et al., 1998).

This tension has been presented as a dichotomy between structured medical records and NLP. Sager et al. (1994, page 42) refers to it as a "controversy between free text and preset categories", and it has been the subject of a brief debate in the pages of the Journal of the American Medical Informatics Association (Schleyer, 2008; Kohane and Uzuner, 2008). The debate centred on an issue of the Journal that contained several papers from a clinical NLP community research challenge (the first i2b2 challenge (Uzuner et al., 2008)) on extracting smoking status from the EPR. Schleyer, commenting on the papers, hypothesised that the general public would be puzzled as to why so much effort was being put into extracting smoking status from text, rather than into ensuring the structured data entry (SDE) of such an important piece of medical history. He went on to say that partial structured recording of this information would be far superior to any available extraction algorithms.

In reply, Kohane and Uzuner countered that data entry using current structured data interfaces is too time consuming, and does not support "the full richness of patient state captured by natural language" (Kohane and Uzuner, 2008, page 708). They pointed out

that the volume of text in the medical record is growing, and is likely to continue doing so.

1.2.3 Knowledge representation, the structured record and natural language

The debate as presented by both Schleyer and by Kohane and Uzuner, is seen as a choice: either NLP, or the structured record. That this is a simplification is not apparent from the literature on NLP and the EPR, which generally ignores the question, and assumes NLP is necessary. It is, however, apparent from the literature on knowledge representation and clinical terminologies, and the literature on the structured entry of data into the EPR, both of which discuss the textual record.

We will turn first to the literature on clinical terminologies, which Rector defines as "concern[ing] the meaning, expression and use of concepts and statements in the [structured] medical record"(Rector, 1999). There is a significant body of research in this area. This is used to underpin many approaches to a structured EPR, on which the following discussion is based.

Rosenbloom et al. (2006) describe interface terminologies as providing a layer between clinicians' natural language descriptions of patients, and the structured data required for re-use by health care applications. In reviewing interface terminologies, Rosenbloom et al. (2006) raises several points that can inform our understanding of the relationship between natural language processing and the structured record. Drawing on several examples, Rosenbloom et al. (2006) suggest that a balance needs to be struck between usability and domain coverage, and that developers should limit the scope of terminologies to specific, constrained use cases. Rosenbloom also points out that rather than terminologies supplanting NLP of clinical text, they assist in the task.

Rector, drawing lessons from the PEN&PAD project (Nowlan et al., 1991; Rector et al., 1995) and the GALEN project (Rector and Rogers, 2006; Rector et al., 1995), considers that clinical terminologies "bridge the gap between language, medicine and software" (Rector, 1998), and considers several aspects of language. These include (Rector, 1998, 1999, 2010):

- the importance of separating terminological models from the language labels describing concepts within these models;
- the computational linguistics behind this separation, in particular how concepts can be expressed in natural language;
- the way in which clinical language often defies literal logical interpretation;

- the way in which language is situated within the dialogue between user and information system, i.e. interaction between system and user in the context of use;

These aspects are primarily concerned with the relationship between things in the world, abstraction from those things to concepts in the mind, and reference to these abstractions and things by signs. Rector extends the usual description of these relationships, Ogden and Richards' semantic triangle, by splitting signs between natural language and formal symbol systems (Rector, 1998, 2010).

In addition to this focus on the relationship between concepts and their natural language representation, Rector makes several comments on the use of natural language in health information systems. Rector is pragmatic, recognising that "there is a fundamental conflict between the needs of software and the needs of human users" (Rector, 1999), and comments that the desire to separate concept representation and natural language should not be taken to mean that all of the medical record can be represented in formally specified terminologies:

> The clinical notes expressed in natural language will, for the foreseeable future, be richer in content and context than any formal representation of them (Rector, 1999)

Rector also argues that one of the functions of formal clinical terminologies is to support natural language processing, aiding in the understanding of concepts described in natural language (Rector, 1998)

A different perspective, but similar conclusions, are provided by researchers in the Structured Data Entry (SDE) and computer-based documentation (CBD) communities. As a recent and currently in-use example, OpenSDE (Los et al., 2005; Bleeker et al., 2006) is intended to structure the narrative found in the medical record, delivering data for routine patient care and for re-use. Describing the goals of OpenSDE, Los et al. (2005) point out that by preference, clinicians enter data as free text, but for research, coded data is preferred. Los et al. (2005) argue that their goal requires a much finer level of granularity than that achievable with the type of terminologies described above. Even so, it has a fall back of free text data entry, for those cases where the SDE model cannot represent the clinical reality.

In reviewing and comparing CBD and the post-processing of free text records, Rosenbloom et al. (2011) describe text processing as a viable alternative to CBD, and argue that the choice between such text processing and CBD should be based on the needs of individual healthcare providers, and not on the idea of a single best method. Rosenbloom et al. (2011) also discuss systems that post-process loosely defined CBD models to produce structured records, attempting to play to the strengths of both approaches.

1.2.4 Can data from the EPR be re-used?

An in-depth analysis of the EPR and data re-use, and an alternative approach, has been given by Greenhalgh et al. (2009), who examined the meta-narratives (over-arching story lines) of twenty-four reviews of the EPR, encompassing several hundred primary EPR studies. Two of their findings are relevant to the discussion of IE and the EPR. First, they find that the research work in the tradition of computer-supported co-operative work (CSCW) has found that use of the structured EPR can have a negative impact on the clinical encounter, but a positive impact on data re-use. This might seem to strengthen the argument that we should look to analyse the unstructured, textual, record.

The second relevant finding by Greenhalgh et al. (2009), however, is to report a dispute on "the extent to which information in the EPR can be extracted from its context and transferred to a different context while still retaining its meaning" (Greenhalgh et al., 2009, page 763). In the case of information extracted from clinical text, we may ask if some fact extracted from the text retains its original meaning, when transferred to some other system for re-use, perhaps in a way not envisaged by the original author. This might be expected to be especially true of written text as opposed to the structured record, given the rich way in which natural language can be used to qualify facts in text with probability, situate them in time, and alter their strength from the negative to the highly important.

We can mitigate for this argument in two ways. First, we can attempt to extract more context. For example, when extracting medications from text, we could consider not just the medication itself, but also the event with which it is associated. Is this a new prescription, a drug being stopped, or a dose being changed? A good example of this approach is provided by the prevalence of negation detection algorithms for clinical IE (see for example Chapman et al. (2001)). Second, we can limit IE to extracting phenomenon that are less subject to context. For example, reporting of blood pressure is less likely to be negated or qualified with a statement of its likelihood.

In recognition of the fact that some meaning can be retained when information is extracted from its context, Greenhalgh et al. (2009) give a qualified conclusion that there can be efficiency gains from data re-use, arguing that:

> Rather than promising that the EPR will "save time" or "make clinical care more efficient", a more honest method would be that creating accurate and complete clinical records requires the sacrifice of time and effort by front-line clinical and administrative staff but that this is (sometimes) justified by more benefits for business processes (e.g., billing), governance, and research (Greenhalgh et al., 2009, page 755).

In concluding this examination of the NLP vs. structured record debate, we can see

that information extraction from the unstructured text of the EPR is not in direct competition with structured data entry. The use of IE does not deny the utility of structured data entry where it can be deployed. Rather it is an attempt to use a different technology to side step a state of affairs that has deep social and cultural roots, and that cannot always be overcome. As Sager et al. (1994) say:

> The need for standards pushes toward preset categories and controlled vocabularies, while the need for expressive power, so as not to distort the patient data, speaks for allowing some amount of free text reporting. A compromise that is not compromising is called for (Sager et al., 1994, page 142).

1.3 Information Extraction

1.3.1 Definition

The research presented in Chapters 2, 3 and 4 uses IE to extract structured information from the unstructured textual portion of medical records. Information Extraction is a sub-field of NLP. NLP can be defined as:

> a theoretically motivated range of computational techniques for analyzing and representing naturally occurring texts at one or more levels of linguistic analysis for the purpose of achieving human-like language processing for a range of tasks or applications (Liddy, 2003, page 137).

IE uses NLP techniques to extract information from relatively unstructured text, and to output it in some structured format, such as fields in database tables. The aim is generally to represent the information from text in a machine-readable, reusable form. Cunningham (2005) defines IE as:

> the process of deriving disambiguated quantifiable data from natural language texts in service of some pre-specified precise information need (Cunningham, 2005, page 665).

Authors often define IE by partial exclusion or comparison, differentiating it from Information Retrieval (IR) (see for example Gaizauskas and Wilks (1998); Cowie and Lehnert (1996)). In this form of definition, IR is said to be the retrieval of relevant texts from a collection, in response to a user query. IE on the other hand analyses the texts, and retrieves specific facts from the texts.

On unpacking Cunningham's definition, several terms and phrases need to be elaborated, as below:

- **Disambiguated data** implies that IE attempts to resolve ambiguities in the meaning of the text being processed, for example resolving temporal expressions such as *tomorrow* into actual dates, and distinguishing between the senses of polysemous words;
- IE extracts **quantifiable data**, i.e. data that can be stored and manipulated by machine;
- IE is targeted at **pre-specified information needs**, and is not an attempt at general language understanding – historically, this was an important limiting step for the field to take;
- IE is also targeted at **precise information needs**, in that what is to be extracted is well defined and tightly constrained.

1.3.2 Background

IE has its roots in research and commercial systems from the 1970s and 1980s, several of which are described by Cowie and Lehnert (1996). One of the main strands of work to influence early IE was the Linguistic String Project (LSP), the history, design and results of which are summarised by Sager et al. (1994). The LSP is especially relevant to the subject of this thesis, as the main domain of application was clinical text, in the form of radiology reports. The LSP stemmed from basic research in the 1960s, with the aim that NLP would "bridge between users and the stored knowledge they need" (Sager et al., 1994, page 143). The first output of the project was a general English parser in 1967 (Sager, 1978a). Sager used this as an initial step in processing language from restricted domains (Kittredge and Lehrberger, 1982). Sager drew on work by Harris (1982) on these so called domain sublanguages, looking for recurrent patterns of word co-occurrence that characterised them, and using these patterns to form a bridge between structure, syntax and the information content of the text. These patterns were used to drive *information formatting*, the conversion of text to a formal, database like structure (Sager, 1978b). The first implemented application of information formatting, in 1976, was to radiology reports (Hirschman et al., 1976).

The greatest impetus to IE research since the late 1980s has been from the Message Understanding Conferences (MUCs) held through to the late 1990s. The MUCs (NIST, a,b,c,d), were largely funded by the US Defense Advanced Research Projects Agency (DARPA), and were started to encourage and evaluate research on automated analysis of naval messages. Later MUCs examined newswire, with the complexity of analysis increasing through the programme (Grishman and Sundheim, 1996). The key feature of

the MUCs were evaluations, in which all participants took part. Each participant developed a system based on sample texts and templates, and then ran this system over unseen texts. The system output (the *response*) was then evaluated against a manually prepared set of filled templates (the *key*). These shared IE tasks and evaluations have shaped IE research (Grishman and Sundheim, 1996; Cowie and Lehnert, 1996), and discussions of and references to MUC are still a regular feature of IE research papers.

1.3.3 IE tasks

Over ten years, the MUCs crystallised five general extraction tasks. The tasks were centred around extracting information into relational records, known as templates. The tasks are given below, adapted from the MUC-7 task definitions (Chinchor and Marsh, 1998).

- **Entity extraction**, originally called named entity extraction, meaning extraction of entities referred to by proper nouns. Originally focussed on finding mentions of things such as organisations and people in the text, the task has been greatly expanded to cover both concrete and abstract things in text. In the clinical domain, this might include entities such as *DISEASE* and *DRUG*.

- **Coreference resolution** finds identity relations between references to entities from the previous task, grouping these references into all of those that refer to the same thing. For example, mentions of *Obama* and *the President* may be determined to both be mentions of the same entity.

- The **Template element task** aggregates the information from the previous two tasks, creating a template record for each entity in the document, and adding descriptive attributes to this. A *PERSON* template, for example, might have attributes for gender and title.

- The **Template relation task** finds relationships between the template elements, for example an *EMPLOYEE_OF* relationship between *PERSON* and *ORGANISATION* template elements.

- The **Scenario (event) template task** extracts events involving the template elements, for example, the event of one person being succeeded by another in some job at some organisation.

Current IE systems do not generally extract MUC-style templates. In the Automatic Content Extraction programme (ACE), a successor to MUC, tasks are conflated into one task for each of entities, relations and events (Doddington et al., 2004). Additionally, the

increasing use of ontologies and linked data in IE, has led to entities being resolved against these resources. The term *semantic annotation* is sometimes used to imply annotation of a text with semantics defined in some external resource, such as an ontology, to form the basis of *ontology-based information extraction* (Bontcheva et al., 2004).

1.3.4 The balance of skills and effort in building an IE system

There are three themes of IE research which, as well as having theoretical ramifications (dealt with by Cowie and Lehnert (1996) and by Gaizauskas and Wilks (1998), for example), have a bearing on the way in which linguistic, domain and software engineering skills and knowledge are deployed, and therefore have implications for the effort required to build an IE system. These are: the use shallow processing as opposed to deep processing; the automatic acquisition of IE rules; and the impact of software engineering. These are discussed below.

1.3.4.1 Shallow processing

There has been a tendency for IE system developers to move away from deep techniques towards the use of shallow techniques (Grishman and Sundheim, 1996). By *deep* processing, we mean processing that is based on some theoretical framework, generally of the grammatical structure of language, and of the semantic content of language (Gaizauskas and Wilks, 1998). By *shallow* processing, we mean processing that is based on simple lexico-syntactic techniques (such as tokenisation) augmented with the recognition of patterns specific to the extraction task (Grishman and Sundheim, 1996). Deep processing systems have been characterised by full syntactic analysis of the text, and construction of a model of the discourse using some formal representation, unifying this with a knowledge base pertaining to the domain (see for example Gaizauskas et al., 1995). Consequently, significant effort is required to model the syntax and semantics of general language. It is held, however, that this effort and modelling is re-usable across domains. Shallow processing, on the other hand, requires not general models of language, but sets of rules (often instantiated as finite state automata) that model simple patterns of language use specific to the domain – for example, rules that define how a diagnosis is expressed in text.

A general outcome of the MUC era was that while shallow techniques may not give a high level of task-independent *language understanding*, they can produce useful results for the easier task of *information extraction*, in less time and with less effort than deep techniques (Cowie and Lehnert, 1996; Gaizauskas and Wilks, 1998).

1.3.4.2 Automatic acquisition of IE rules and models

The second theme that affects the balance of skills in building an IE system, is the automatic acquisition of IE rules or models, through the training of ML systems. This was part of a wider shift in NLP, towards statistical methods, which itself reflects a shift in linguistics, as outlined by McEnery and Wilson (1996). Prior to the advent of statistical modelling of corpora, NLP was focused on the construction of language models in the shape of hand-written grammars and rules. Such an approach will always come up against the fact that language can combine words in seemingly endless ways, embracing ambiguity and always breaking the rules. By focusing on the words and their syntactic relationship, these grammars ignore semantics. Incorporating semantics into the grammar invariably leads to an explosion in the number of rules for all but the most restricted language (Nadkarni et al., 2011). In the face of these problems, grammar construction depends on the model builder – the computational linguist – deciding what should and should not be modelled, and how it should be modelled. Statistical NLP, on the other hand, relies on the increasing availability of corpora of electronic texts through the 1980s and 1990s, and the increase in computational power. It became possible to examine every inch of large corpora, and to build models of language phenomenon that were based on the empirical data in those corpora. A statistical approach will generalise from the corpus, giving the most probable answer. It does this with a rigour not possible in a hand-built grammar, having examined the entire corpus in order to build the model that informs its choices.

Statistical approaches were initially applied to lexico-syntactic tasks, such as part-of-speech tagging (Gaizauskas and Wilks, 1998), but by the time of the ACE program (Doddington et al., 2004), they were in common use across all IE tasks, learning rules and models from annotated training data (Cunningham, 2005). This has had the effect of moving effort away from language engineers skilled in the representation of IE requirements as extraction rules, and on to annotators with sufficient domain knowledge to find and mark the required phenomenon in example texts (Cowie and Lehnert, 1996). In the case of a typical MUC task, for example extracting people, organisations and the relationships between them, this manual annotation of examples might be expected to be a straightforward task. It may not be so straightforward for a technical domain, such as medical records.

Despite the shift in skills, this *supervised* ML, in which the learning algorithm is presented with manually labelled examples, may still have a high cost, resulting from the need to provide large numbers of training examples. Several strands of research have attempted to overcome this, by attempting to cheaply increase the number of training examples through semi-automatic means, generally using a set of seeds to learn some

pattern with which further training examples can be generated. These *semi-supervised* approaches to ML are detailed in Abney (2007).

Other research has looked to decrease cost through *mixed initiative*, or *active*, learning. In this, the algorithm is trained on an initial set of examples. It then suggests further annotations to the user, who either accepts them as correct, or corrects them. These corrections are then fed back to into the pool of training examples, the algorithm re-trained, and further examples proposed for correction. For a medical example, see Patrick and Sabbagh (2011).

Finally, in *unsupervised* learning, there has been an attempt to learn clusters of semantically related entities and relations (see, for example, Grefenstette (1994); Finkelstein-Landau and Morin (1999); Reinberger et al. (2003)).

1.3.4.3 Software engineering and NLP frameworks

Software engineering has also impacted the balance of skills in building an IE system. Many researchers, such as Cowie and Lehnert (1996), have recognised the software engineering cost in creating a large IE systems. This has spurred on the development of several NLP software frameworks and architectures (for example Apache UIMA, 2012; University of Sheffield, 2012; Apache OpenNLP Development Community, 2011; Carpenter and Baldwin, 2011; Bird et al., 2009). These are intended to ease the task of NLP and IE system development, by providing re-usable infrastructures and well-defined interfaces [5]. Additionally, the communities that have formed around these frameworks, have provided linguistic knowledge in the shape of NLP tools, which may be re-used within the frameworks. For example, a typical framework might include ready-to-use tools for tagging of words in a document with their grammatical part-of-speech, constructing syntactic parses of phrases, lookup of terms in domain dictionaries, and so on. A similar trend of framework and toolkit development has occurred in the ML community (see Witten and Frank (2005) for example), and many of the NLP toolkits listed above are integrated with ML tools and algorithms.

The work reported in this thesis is based on shallow techniques, supervised ML, and maximises the use of NLP and ML frameworks. This was a conscious attempt to shift the balance of system development away from the hand-crafting of template extraction rules, towards the annotation of example corpora by domain experts. This is discussed further in Section 1.6.

[5] By *interface*, we mean the point of connection between software components. If this is well-defined, then software from different vendors can be used together, in much the same way that you can plug all of you electrical appliances into the same sockets.

1.4 Clinical Information Extraction

Having defined and described Information Extraction, we now turn to clinical information extraction. We define clinical information extraction as information extraction carried out on clinical text. By clinical text we mean the unstructured, textual portion of the medical record. Further, we limit our definition to text that is already held in electronic form, and therefore will concern ourselves with neither the transcription of dictated text, nor the optical recognition of handwritten texts.

In considering clinical information extraction to be a form of *information extraction*, we consider it to be exclude other applications of NLP over the medical record, such as automatic coding (Stanfill et al., 2010), decision support (Demner-Fushman et al., 2009), or de-identification (Uzuner et al., 2007).

In considering clinical information extraction to be over *clinical* texts, we exclude other forms of medical text, such as medical journal text, text books, clinical trials, public health information, blogs and social media.

We also exclude the more general and widely used term *biomedical text*, and IE from biomedical text, which we take to refer to any text in the life science literature, i.e. medicine and biology. This is in line with Meystre et al. (2008). Processing of general biomedical text is discussed by Ananiadou and Mcnaught (2005); Zweigenbaum et al. (2007); Aggarwal and Zhai (2012).

1.4.1 Why is Clinical IE different?

Clinical NLP, and by extension clinical IE, is often treated as a distinct sub-speciality of general NLP. One reason for this may be social. Patient data is often considered personal and access is therefore restricted. Additionally, technologies that assist medicine are considered worthy by society as a whole. Some informatics and computing research groups have therefore formed within healthcare establishments, where access to records is more practical, and where they concentrate on this data.

There is also a technical reason. The language of medicine is widely considered to present specific difficulties to NLP, and we have touched on this in Section 1.2.2, where we considered the way in which clinicians value the expressivity of natural language, in particular the way that it can deal with ambiguity and uncertainty.

Nadkarni et al. (2011) look at some of the sub-problems of NLP, and how they apply to clinical text. We have listed these below, together with additional problems listed by Spyns (1996), by Meystre et al. (2008), other references to studies of specific problems, and problems found in the course of the work reported in this thesis.

- Sentence boundary detection may be complicated by the large number of abbreviations, medical titles, and lists.
- Tokenisation of biomedical text may be complicated by the heavy use of non-alphanumeric characters in e.g. drug doses, chemical names and drug names.
- There is a wide range of texts, from the prose-like nature of letters, to highly telegraphic and terse imaging reports.
- Misspellings are common.
- Both shallow and deep parsing may face problems with the ungrammatical, telegraphic nature of some texts (especially reports), together with the frequent use of lists, proformas and other text structures.
- Entity recognition may suffer because of a high degree of word order variation in named entities (for example, *perforated duodenal ulcer* as opposed to *duodenal ulcer, perforated*).
- There is a high degree of synonymy in medical language, also impacting on entity recognition.
- There is a high degree of polysemy, e.g. *haemoglobin* can refer to a substance, a laboratory test, or the result of that test. This requires a high degree of word sense disambiguation (Gangemi et al., 2000; Pisanelli et al., 2004).
- Abbreviations and acronyms are widely used, and may even be specific to a particular institution (Xu et al., 2009).
- Abbreviations and acronyms may themselves be polysemous (Liu et al., 2001).
- Phrases in medical text are frequently qualified with negation and uncertainty modifiers, with use of words such as *suggestive of*, *probably*, *less likely*.
- Temporal information is presented in a domain-specific way. For example, *I have arranged an x-ray*, the verb tense does not tell us that the x-ray took place before the date of the document in which this phrase occurs (Gaizauskas et al., 2006).
- Training and evaluation data is often anonymised to overcome ethical constraints. This is often carried out by replacing the patient name with a string of meaningless characters, and gender also obscured. This means that co-references between the patient, derivations of their name and pronouns may be difficult to spot.

- Terminology can cause other problems. In addition to the large number of technical terms, general language terms can take on a specific meaning. Neologisms (new terms, not previously seen) are also common (Fisk et al., 2003).

The following two sections gives a brief description of trends in clinical IE and in the construction of corpora for clinical IE, examining some of the major clinical IE systems and corpora of the past two decades. The position of the work described in this thesis relative to these trends is delineated in Section 1.6.

1.4.2 Trends in clinical information extraction

Historic contributions to clinical IE are given in Spyns's broad review of NLP in medicine (Spyns, 1996). A more recent review focused specifically on clinical IE can be found in Meystre et al. (2008). Reviews of related applications of NLP to clinical text include Demner-Fushman et al. (2009) on the use of NLP to support clinical decision support, which includes a general review of current clinical NLP systems, and Stanfill et al. (2010) on the related field of clinical coding, i.e. the assignment of codes from standard terminologies to narrative text.

Most discussion of clinical IE, and of IE in general, considers work on the LSP to be formative. As discussed in Section 1.3, the LSP developed a technique called information formatting, in which information was extracted from text into a tabular form for further analysis (Hirschman et al., 1976; Sager et al., 1994). This format was a precursor of the template structures used in the MUCs. From this early start, clinical IE has developed along same lines as much of general NLP and IE, with an important delay in the introduction of some ideas, that reflects the poor availability of data outside of a few major centres, which in turn reflects the ethical issues associated with the re-use and sharing of medical records (Meystre et al., 2008, page 131). While general IE in the shape of the MUCs was able to push forward with shared data and collaborative evaluations, clinical IE lagged behind somewhat. The main trends are outlined in the following sections.

1.4.2.1 Towards shallow understanding and machine learning

Of the eleven full systems reviewed by Spyns (1996), seven created models of the text based on conceptual graphs (a formal logic used for knowledge representation (Sowa, 2000)), and four created other models of domain semantics. LSP (Sager et al., 1994) and Specialist (McCray, 1991) created models based on less formal representations. In the case of Specialist, this consisted of a combined syntactic and semantic parse. In the case of LSP, it consisted of a syntactic parse followed by selection of a semantically correct parse based on medical word-class co-occurrence patterns. The widespread use of deep

understanding may reflect some bias in Spyns's selection, but it is similar to the situation in the MUCs a few years earlier.

Typical examples of the early deep understanding clinical IE systems include MENELAS (Zweigenbaum, 1994; Zweigenbaum et al., 1995), medsynDikate (Hahn et al., 2002), and the Geneva Hospitals system (Baud et al., 1992), all of which create some discourse model of the text, linked to and resolved against a model of the domain. Importantly, Zweigenbaum (1994) reports that knowledge development is time consuming and error prone. Hahn and Schulz (2003) approached this by building domain models from the knowledge present in the Unified Medical Language System (UMLS – a set of lexical and semantic repositories combined into a common format, see below, this section). They reported, however, that it contained inconsistencies and lacked a formal framework.

Friedman et al. (1994, pages 162 to 163) makes the distinction between semantically driven clinical IE systems, and those that combine syntactic and semantic analysis. All of the aforementioned systems combine syntactic and semantic analysis. Typical semantics-only systems include MedLEE Friedman et al. (1994) and SPRUS (Ranum, 1989). In SPRUS, Ranum (1989) tackled the problem of the cost of knowledge engineering by re-using an expert system knowledge base, compiling a semantic grammar from that knowledge base. SPRUS was in operation at LDS Hospital, Salt Lake City, where it was later succeeded by the SymTEXT system that followed a syntactic parse with a semantic analysis (Haug et al., 1994). This has itself been succeeded by MPLUS, which interleaves syntactic and semantic parses (Christensen et al., 2002). Both SymTEXT and MPLUS use a model of semantics based on a network with probabilities assigned to arcs. The construction of these is described as the most time consuming task by Christensen et al. (2002). MedLEE, which is in production use in two hospitals (Meystre et al., 2008), has a semantic model based on that of LSP. Semantic classes include *disease*, *region*, *and device*. Rules specify patterns involving these classes to be instantiated from the text, and information structures to which they should be mapped. These information structures are the domain model (Friedman et al., 1994, page 163). Friedman et al. (1994) does not discuss the time taken to create the semantic model, but given that MedLEE is one of the longest standing clinical IE systems, significant time is likely to have been spent on this.

In contrast to the systems listed by Spyns (1996) and those described above, more recent reviews and commentaries (for example Demner-Fushman et al. (2009), Chapman et al. (2011), Savova et al. (2011)) show a shift to much shallower systems. This is not to say that deeper understanding systems are no longer used, and several of those listed above are still in use. A typical example of such a shallow system is the SPIN IE system (Liu et al., 2005), which includes assignment of semantic classes by lookup of terms in UMLS, regular expression based negation detection, and pattern matching

rules to determine the attributes of concepts in histopathology reports. SPIN formed the basis of caTIES, a system in use to extract information from the descriptions associated with tissue samples, to aid in retrieval of those samples. caTIES uses nearest neighbour techniques to derive a topology of concepts (Crowley et al., 2010, page 257). HITEx also uses UMLS term lookup and regular expression based negation detection, together with noun phrase chunking, and regular expression based entity extraction (Zeng et al., 2006). cTAKES (Savova et al., 2008, 2010) and MedKAT/P (previously MedTAS/P) (Coden et al., 2009) are architecturally related systems, and both make use of shallow parsing and regular expression based rules for detection of negation and other useful constructs.

In addition to the move towards shallower understanding, many systems make some use of ML. HITEx, for example, uses Support Vector Machine (SVM) classifiers to determine if a sentence refers to patient smoking, and cTAKES uses Naive Bayes classifiers for named entity recognition. It is also noticeable that in recent clinical IE challenges, similar in nature to the MUCs, successful systems are shallow, ML based systems. Of the ten top performing concept extraction systems presented in the 2010 i2b2/VA workshop, five used supervised ML, two used semi-supervised ML, and three used a hybrid ML and shallow rule based system. For the top ten relation extraction systems, eight used supervised ML, one used semi-supervised ML, and one used a hybrid ML and shallow rule based system (Uzuner et al., 2011).

1.4.2.2 Knowledge resources

Medicine is rich in knowledge resources: terminologies, vocabularies, taxonomies and ontologies. Almost all of the above systems, whether deep or shallow, make use of these to some extent. By far the most commonly used resource is the UMLS (Lindberg et al., 1993), which has been used for the work in this thesis. Begun by the US National Library of Medicine (NLM) in 1986, the UMLS was conceived of from a publishing, retrieval and librarianship perspective, focused on machine-readable biomedical information to integrate literature, patient observations, and educational material (Humphreys et al., 1998a). The UMLS does not attempt to build yet another biomedical terminology, but contains an integrated collection of those already in use. The UMLS aims to tackles the variety of ways that concepts are expressed in different machine readable sources, and the relationship between these sources and user's retrieval questions (Lindberg et al., 1993; Humphreys et al., 1998a).

The terms within the UMLS source terminologies do not necessarily correspond to the terms used in natural text. For example, classifications used within epidemiology may contain "catch all" terms such as *Heart Disease Not Otherwise Specified*. This term is only understandable when the reader knows the complete set of heart diseases that *were*

specified in the classification. Other terms may only be understandable when seen in the context of their parents in the classification of origin. For example, the term *complications* within the obstetrics branch of a classification. The UMLS was not originally designed with language processing in mind, and neither were the source vocabularies: they are conceptual in nature, not lexical. The inclusion of conceptual structure makes the UMLS the largest source of knowledge on the semantics of medical terminology. This has been widely used, as detailed by Selden and Humphreys (1997). A current search of PubMed for UMLS in titles, abstracts, and in the PubMed subject index returns over 1000 results (NLM, 2012).

The UMLS consists of three separate resources. The *Metathesaurus* is seen as a classical thesaursus, i.e. translating ideas into words, rather than words into ideas, and contains terms and concepts from a range of source terminologies, linked in a single structure. Linkage makes use of the structural knowledge (both lexical and conceptual) within the terminologies themselves, and lexical matching techniques (Lindberg et al., 1993). The basic unit of the metathesaurus is the concept, each of which is identified by a Concept Unique Identifier (CUI). The *Semantic Network* provides a consistent high level categorisation of concepts in the metathesaurus, and links them through a set of relationships (Lindberg et al., 1993). It provides a basic ontology of biomedicine (McCray and Nelson, 1995), currently containing 135 semantic types and 54 relationships. The *UMLS lexicon* was added in the context of experiments on using UMLS to improve the parsing of biomedical text (McCray et al., 1994). It consists of lexical frames recording syntactic, morphological, and orthographic information. It also records spelling variants. Lexical items are drawn from the most common words in general language dictionaries and from a collection of MEDLINE abstracts (McCray et al., 1994).

In addition to developing the UMLS, the NLM has also developed several tools that make use of it, including the Specialist language system (McCray, 1991) (described above in Section 1.4.2.1), and MetaMap (Aronson, 1996, 2001). MetaMap uses a syntactic parse, a look-up in the UMLS knowledge sources, and a set of heuristics, to map free text phrases to Metathesurus concepts and semantic types (Aronson, 2001). MetaMap is used by several of the IE systems described in the previous section.

Within the research reported in this thesis, terms are recognised using the Termino system (Harkema et al., 2004b). Termino allows loading of terms from heterogeneous resources, including UMLS, into a database. These are then compiled into finite state recognisers, with which spans of texts matching these terms can be annotated and associated with identifiers from the resource of origin.

1.4.2.3 Modularisation and the use of NLP frameworks

Another trend apparent in the systems described in Section 1.4.2.1 is the extent to which NLP architectures and frameworks are now being used within clinical IE. None of the deep understanding systems described have re-use of particular processing components, or even the whole, as a stated design aim, although some have been adapted to multiple uses at their site of origin, and others have been ported to sites other than that of their origin (notably MedLEE (Hripcsak et al., 1998)). All of the shallow systems described in Section 1.4.2.1, however, are modular, and have been constructed from general NLP frameworks and toolkits, the most popular being the Unstructured Information Management Architecture (UIMA, (Apache UIMA, 2012)), used by MedKAT/P and cTAKES, and the General Architecture for Text Engineering (GATE (Cunningham et al., 2002)), used by SPIN, caTIES and HiTEX.

The advantage of this modularisation, is that alternative sub-processes may be easily tested; individual sub-processes and their effect on the overall system may be isolated from the whole; and as discussed in Section 1.3.4.3, the software engineering cost of the system can be reduced. Such frameworks encourage re-use of tools and sharing – it becomes easy to use another person's best of breed named entity recogniser, if you can take it and plug it into your framework with minimal effort. This has led to specialisations of these NLP frameworks specifically for the clinical domain, and the establishment of open source efforts for sharing IE systems and tools. The Open Health Natural Language Processing (OHNLP) Consortium (OHNLP, 2012a) was established to promote the development of open source health NLP, focused on UIMA-based tools and annotated data (Apache OpenNLP Development Community, 2011). Two IE systems have been made available through OHNLP, released in a joint initiative by IBM and the Mayo Cinic (IBM, 2009). These are cTAKES (OHNLP, 2012b; Savova et al., 2008, 2010) and MedKAT/P (OHNLP, 2012c; Coden et al., 2009). Other developers have begun to join the initiative, with for example plans to integrate ODIE (an ontology and IE toolkit) with cTAKES (Crowley, 2010).

An interesting question raised by Nadkarni et al. (2011), is whether NLP software is likely to become a commodity? By commodity software, they mean software that can be easily set up to perform some task, without any programming skills, and without specialist skills, such as IE skills in our case. The implication of commoditised NLP software, would be the ability of non-NLP specialists, perhaps clinicians or hospital IT departments, to set up and run IE for specific purposes. Nadkarni et al. (2011) argue that current NLP toolkits are still oriented to the advanced programmer, rather than the commodity market, and that NLP has not yet reached the stage of commoditisation found in statistical packages and data mining.

1.5 Corpora and annotation

As with general IE, clinical IE requires corpora of example texts annotated with the phenomenon to be extracted, for training and development of systems. Although unsupervised and semi-supervised methods of ML do have some currency, their use is not yet widespread (Uzuner et al., 2011). Annotated data is also needed by all systems for evaluation, and to assist with gathering requirements.

Publicly available annotated corpora in the broader area biomedical NLP area are relatively common, mostly consisting of annotated journal abstracts and articles – a short review is given in Section 2.3. Other prominent biomedical corpora not reviewed in that chapter include the CALBC corpus, a harmonisation of automatic annotation from multiple systems (Rebholz-Schuhmann et al., 2010), and the CRAFT corpus, a corpus of full-text journal articles richly annotated with several thousand entity classes (Bada et al., 2012).

Despite clinical IE being one of the major applications of IE, and being formative in the history of IE, it was twenty years from the release of the first MUC shared task corpus to the release of the first clinical IE corpus. It has proved very difficult to gain acceptance for the release of even anonymised clinical text, due to concerns over privacy (Chapman et al., 2011; Meystre et al., 2008). The last few years, however, have seen an increasing number of such corpora released, and used for MUC style competitive challenges. The main challenges in clinical IE have been the i2b2 challenges (I2B2, 2007) and the TREC medical records track challenges (TREC, 2011, 2012). A partial list of publicly available clinical text corpora is given in Chapter 2. As far as we are aware, the following is a current and complete list of all challenges with shared corpora of clinical text and all other publicly available clinical text corpora:

- i2b2 2006 Deidentification (Uzuner et al., 2007) and Smoking Challenges (Uzuner et al., 2008)
- i2b2 2008 Obesity Challenge (Uzuner, 2009)
- i2b2 2009 Medication Challenge (Uzuner et al., 2010c,b)
- i2b2/VA 2010 Relations Challenge (Uzuner et al., 2011)
- i2b2/VA 2011 Coreference Challenge (Uzuner et al., 2012)
- i2b2 2012 Temporal Relations Challenge (I2B2, 2012)
- University of Cincinnati Computational Medicine Center classification of radiology reports (Pestian et al., 2007)

- University of Cincinnati Computational Medicine Center sentiment analysis of suicide notes (Pestian et al., 2012)
- The University of Pittsburgh NLP Repository (University of Pittsburgh Department of Biomedical Informatics, 2012), a repository of one month of de-identified clinical reports from multiple hospitals. Users annotating the documents must contribute their annotations back to the repository for the community.
- The first TREC medical recrods track, using data from The University of Pittsburgh NLP Repository (TREC, 2011)
- The second TREC medical records track is ongoing at the time of writing, and also uses data from The University of Pittsburgh NLP Repository (TREC, 2012)
- The ODIE corpus of co-reference, which includes texts from The University of Pittsburgh NLP Repository and the Mayo Clinic (Chapman et al., 2012; Savova et al., 2011)
- The Adverse Drug Reaction (ADR) corpus is based on case reports rather than medical records, and uses a methodology developed from the one reported in this thesis (Gurulingappa et al., 2012).
- The ImageCLEFmed challenge evaluations used case descriptions associated with medical images (Hersh et al., 2006; Müller et al., 2007).

In addition to these publicly available corpora, many of the earlier clinical IE systems reported in the literature have been developed with corpora built specifically for the development of that system (for example, Sager et al. (1994); Christensen et al. (2002); Coden et al. (2009); Zeng et al. (2006)). For other systems, very limited information is given about the corpora used in evaluation (for example Haug et al. (1994)). As recognition of the importance of annotation and corpora has grown, however, more of this corpus work is described and published in its own right. For example, Pakhomov et al. (2006) report on a corpus of clinical notes manually annotated for part of speech, for use in training part of speech taggers for clinical NLP systems; Chapman and Dowling (2006) reports on the creation of a schema for emergency department reports, and on manual annotation using this schema (Chapman et al., 2008); and Roberts et al. (2009) give a full description of the corpus used in this thesis.

A greater emphasis on corpora has in turn led to the examination of the methods used in corpus creation. Such examination is perhaps more scarce in the general IE literature, but includes Boisen et al. (2000). Cohen et al. (2005) reviews six general biomedical

corpora and their usage, showing that the uptake of a corpus depends on the quality of that corpus. They consider dimensions of quality such as size, the effect of structural and linguistic annotation, distribution format, and level of semantic annotation. The strongest predictors of re-use were format and the extent of basic structural and linguistic annotation.

Xia and Yetisgen-Yildiz (2012) examine three manual annotation tasks of their own, using physicians as annotators. They examine the level of guidelines and training needed, and the problem of time commitment from busy physicians. They conclude that domain expertise is required for medical entity annotation, but that medical training alone is not sufficient to guarantee high quality annotation (Xia and Yetisgen-Yildiz, 2012). Scott et al. (2012) have also examined the use of medical experts when annotating medical language, and conclude from a rigorous study that their use is essential. Hripcsak et al. (1999) look at the number of annotators required for a given IE task, to create annotations to a given standard. Again, Hripcsak et al. (1999) use clinical experts. Snow et al. (2008), looking not at clinical annotation but the annotation of linguistic phenomenon, has compared the expensive use of linguists to the potentially cheaper use of non-expert volunteers, and concludes that to reach sufficient quality of annotation for an easy-to-explain task, multiple annotation by four non experts may be cheaper than annotation by a single expert.

In addition to the questions of who should annotate, any manual annotation exercise has to deal with the question of what to annotate. As Hahn et al. (2012) report, many clinical entities have fuzzy definitions. For example, it is clear that *appendicitis* is a disease, but is *high blood pressure*? It is also not clear where one annotation starts and another ends. For example, in *lung cancer*, what should be annotated as a *Disease* entity, and what as an *Anatomy* entity? In other cases, it is not clear what span of text should be annotated. For example, is the disease mentioned in *clumsiness in her left extremities*, the entire phrase, or just *clumsiness*? Hahn et al. (2012) suggest solving this problem with better definition through iterative guideline development, the use of clear guidelines with clear demarcation of what is and is not an entity, and allowing an approximate annotation span for cases where the extent of the disease mention in text is not clear.

1.6 CLEF: a Clinical E-Science Framework, and IE

The body of this thesis, in Chapters 2, 3, and 4, presents a corpus of clinical texts, semantically annotated for entities and relations, and the training and evaluation of a supervised clinical information extraction system from this corpus. The corpus and system were developed for the CLEF project (Rector et al., 2003). This section describes the CLEF project, the evolution of the CLEF IE system, and position it in the context of the other

research on clinical IE and corpora described in Sections 1.4 and 1.5 above.

1.6.1 The CLEF project

CLEF and the follow on CLEF Services project were multi-site research projects funded by the United Kingdom Medical Research Council (MRC, grant references GO300607 and RB106367 respectively) (MRC, 2012). The projects researched the development of technologies and techniques required for a high quality repository of electronic patient records, together with the issues of security and interoperability raised by the use of such a repository. CLEF worked in the area of cancer informatics, using medical records provided by the Royal Marsden Hospital (RMH), a large specialist oncology centre and CLEF partner.

One strand of the research was to create a structured representation of the textual portion of the record through clinical information extraction (Harkema et al., 2005), in order that this could be integrated alongside the structured record within the CLEF repository. This augmented structured record would then be made available via the repository for search and retrieval, in order to support both day-to-day care and clinical research. Two end-user applications were created by CLEF to aggregate the data across all of the information extracted from documents, and all structured data, for a single patient record within the repository. These applications were:

1. The CLEF chronicle, building a chronological model for each patient, integrating events from both the structured and unstructured record (Rogers et al., 2006).

2. CLEF report generation, creating aggregated graphical and textual reports from the chronicle (Hallett et al., 2006).

1.6.2 Historical background of the CLEF IE system

The CLEF IE system was developed to provide information for use in both of the above end-user applications. The initial IE system built for the CLEF project was based on the AMBIT IE system (Harkema et al., 2005), but was never fully adapted to CLEF. This was superseded by the second CLEF IE system, as described in this thesis. The shift away from AMBIT is relevant to the aims and objectives of this thesis, and in order to understand this, we will first describe the evolution of AMBIT.

AMBIT was historically descended from the University of Sussex system developed for MUC-5, which was itself an adaptation of a system developed for monitoring police reports of traffic incidents (Gaizauskas et al., 1993). This was adapted to form the LaSIE system used in MUC-6 (Gaizauskas et al., 1995; University of Sheffield, 1996), and then

this ported to GATE (Humphreys et al., 1998b) for MUC-7 [6]. Over the next few years, the LaSIE system was adapted for the biomedical domain in the Enzyme and Metabolic Pathways Information Extraction (EMPathIE) project (Humphreys et al., 2000a,b), and the Protein Active Site Template Acquisition (PASTA) project (Gaizauskas et al., 2000, 2003). The PASTA system was ported to GATE 2 (Cunningham et al., 2002) in the context of the CLEF and MyGrid (Goble et al., 2003) projects, and renamed AMBIT. A large scale terminology resource, Termino (Harkema et al., 2004b,a), was added, and the application was adapted to serve as a core system for general biomedical text, which could be specialised for specific sub-genres such as journal or clinical text (Harkema et al., 2005).

The AMBIT system, as described by Harkema et al. (2005), comprised three stages of processing. The first stage carried out simple lexico-syntactic processing followed by a finite state recogniser (FSR) compiled from a term database, Termino (Harkema et al., 2004b), and a set of term grammars, used to combine short terms into longer terms. The second stage consisted of a partial syntactic and semantic parse using the bottom-up chart parser derived from earlier MUC competition systems (Gaizauskas et al., 1995). The third stage integrated the results of the previous stage into a discourse model, built in a formal ontology language (Gaizauskas and Humphreys, 1996), also inherited from the MUC systems, and exported information from this model as MUC-style templates. Although AMBIT had elements of deep understanding, such as the discourse model component, there was a significant adaptation to specific domains such as CLEF, through the use of Termino, and the construction of term grammars. Development was largely through an introspective analysis of patient notes. The addition of Temino was the major change between the PASTA system and AMBIT, and was included to deal with the scale of biomedical terminology. For the CLEF project, the major part of the terms in Termino consisted of those taken from UMLS, with some additions and exclusions specific to CLEF.

1.6.3 From AMBIT to the CLEF IE system

The AMBIT-based approach to IE in CLEF encountered a number of problems, reported in Roberts et al. (2009) and Chapter 2, and considered here by comparison to the second CLEF IE system that replaced it. In doing this, we will also delineate the position of the second CLEF IE system relative to the landscape of clinical IE and corpora introduced in Sections 1.4.2 and 1.5.

First and foremost, AMBIT faced a problem of requirements definition within CLEF. The original CLEF templates were intended to model 15 entity types, each with several

[6] LaSIE used GATE version 1 as described in Cunningham et al. (1997), for the later versions of GATE used in this thesis, refer to Cunningham et al. (2002) and Cunningham et al. (2011))

properties. The planned number of relations is not recorded, but would likely be greater than the number of entities. The number of entities was later reduced to 9, with 16 relations between them (Roberts et al. (2009), also Chapter 2). This compares to four entities and three relations in MUC-7 (Chinchor and Marsh, 1998). Beyond the definition of templates, there was also an intention to define specific extraction tasks using these templates, within the clinical domain, such as mining radiology reports for signs indicative of lung cancer, and the relationship between these signs and anatomical locations (Harkema et al., 2005). Very few of these more specialised tasks were completed. As well as being ambitious in scale, the plan also suffered from a lack of requirements coming directly from end-user clinicians, who did not have the time needed to fully engage with requirements gathering.

This problem was tackled within the second CLEF IE system by reducing the number of entities to six, and relations to four. No properties were modelled, and instead three additional entity-like objects called *modifiers* were defined, which could modify other entities with properties such as laterality and negation. Co-reference was modelled as another relation. This schema was developed by a group consisting of computational linguists and several clinicians. There was no attempt to define specific tasks beyond extraction of these entities and relations, it being assumed that once IE could be achieved and demonstrated through the exposure of these entities and relations in CLEF applications, then further requirements might be elicited.

The second problem with AMBIT use in CLEF, stemmed from the difficulty in defining requirements, and concerned the gold standard corpus used for AMBIT development. The original schema, or template definition, was formally described. There were, however, no guidelines as to how templates should be filled from text, which led to gold standard templates created to a different set of goals than those of the IE system. The biggest problem encountered was that manual templates were created for every mention of the same thing in text. For example, two mentions of the same bladder would lead to two manually created templates, whereas AMBIT would create one. Automatic merging of these duplicate manual templates was not possible. The second CLEF IE system tackled this problem by defining a set of guidelines, rigorously developed by a team of clinicians and computational linguists, and by developing a methodology for manual analysis of texts by multiple clinicians trained in the use of the schema for this purpose. The schema and guidelines were used to drive the creation of a much larger gold standard than had been available to AMBIT. This is described in Chapter 2, and is similar to other such efforts reported in Section 1.5.

The new gold standard also represented a departure from MUC-style templates. Templates are independent of the text. They have no link to a particular span of the text, but

instead describe objects in the world, that are also referred to by the text. The new gold standard, however, was based on textual annotation, with each annotation being anchored to a defined span of text, and describing the entities and relations in the text. This represented a shift away from the text understanding systems of MUC, in which models of the world described in the text were built. Instead, the new IE system had the task of extracting all entities and relations from the text. Basing the gold standard on annotations, instead of templates, meant that entity and relation extraction could be tackled with a supervised ML approach. ML is applied to text by making some classification decision about every unit of the text, where the unit of classification is usually words or sentences. In the case of word-based classification, ML forces a decision about every word in the text: every one must be examined by the ML algorithm and some decision reached about it. Being anchored in the text, annotations can be used to provide features for this ML, whereas templates are removed from the text. With a template-based deep understanding system, it is possible, and usual, to choose to ignore aspects of the text. Constructs and phenomenon that are difficult to deal with can be left un-modelled. Statistical NLP is more rigorous, as everything is examined, and nothing left un-modelled.

The shift to supervised ML was also a conscious decision taken to maximise use of the annotations, and to speed up development time. It was thought that selecting appropriate features and training supervised ML models would be faster than the introspective development of rules for the extraction of entities and relations. It was also felt that a supervised ML approach would be more scalable, in that the same application could be re-used for other entities and relations, the cost of this being restricted to the provision of additional training examples. The move to supervised ML parallels that described for general and clinical IE in Sections 1.3.4.1 and 1.4.2.1.

Both AMBIT and the second CLEF IE system were based on a standard NLP framework, such as the ones described in Section 1.4.2.3. AMBIT, however, used custom components that were not distributed with the main framework, such as its parser, the discourse model, and Termino. The second system was built from components distributed with the framework, with the exception of Termino. The major functional difference between the two systems was in the replacement of hand-crafted rules with ML models of entities and relations. The consequence of this shift was a greater separation of the system and data, and shift in the skills needed to build the system. The major efforts became creation of training examples (annotated texts) by domain experts, and the empirically-driven selection of features for ML algorithms.

The CLEF IE system is not just a break from its historical predecessors. It is also novel within the wider field of clinical IE. The shift to supervised machine learning required a large and complex annotation exercise. This led to an exploration of annotation and

corpus creation for clinical IE that went beyond the depth and extent of all previous work, and resulted in the most richly semantically annotated clinical text corpus yet built. This work is described in Chapter 2. The annotation guidelines used in this exercise are given in Appendices B and C.

The CLEF corpus has subsequently been used to create the entity and relation extraction components of a full IE system. Entity extraction made use of both a large scale terminology resource, and statistical ML, as described in Chapter 3. Relation extraction made use of a statistical ML approach, and is the first reported application of this to the extraction of clinical relationships. This work is described in Chapter 4.

1.7 Contributions of this Thesis

This thesis describes the construction of a corpus of clinical text, manually annotated for entities and relations, and a clinical IE system trained on this corpus. When we consider the landscape of IE, as presented in Section 1.3.4, it is clear that there has been a move to supervised ML-based systems, and that there has been a recognition of the part played by software engineering in building practical and portable IE systems. These trends are also becoming apparent in clinical IE, as discussed in Section 1.4.2, although there has been a time lag in their uptake. Clinical IE is carried out in a complex and wide-ranging domain, over text with its own particular properties, and where human understanding of that text requires specialist knowledge. We ask, how far can supervised ML and generic architectures be applied to clinical IE, and what are the implications for system development?

The objectives of this thesis, as stated in Section 1.1.3, arise directly from this question, and are summarised below.

- To adopt a supervised ML approach to clinical IE, using a rigorously developed corpus of manually annotated clinical texts.
- Previous clinical IE systems have made use of a rich body of knowledge resources. Can these be used with supervised ML?
- To use as far as possible, and with minimal change, off-the-shelf system NLP and ML frameworks. How far away are these systems from non-expert use?

Each of these objectives relates to the knowledge needed to build a system. Our aim, as stated in Section 1.1.3, is to lower the barrier to building a clinical IE system, through the separation of linguistic, domain, and engineering knowledge, and through the re-use of pre-existing resources in reach of these areas. The unstructured portion of the medical record persists, and is likely to continue to do so. In order to re-use the record, and close

the loop between clinical practice and research, we need to lower the barrier to extraction of information from this record.

1.7.1 Specific contributions

The general trend in clinical IE over the last few years has been towards ML-based systems trained from rigorously constructed corpora. The work reported in this thesis is a part of that trend, and has helped to form it. The specific contributions of the work reported are:

Corpus creation. The work has explored the problem of producing a corpus annotated for clinical IE to a greater depth and extent than before, and made a significant contribution to research on clinical language processing in terms of the methodology adopted to develop the corpus. Chapter 2 reports the creation of the most richly semantically annotated resource for clinical text processing built, and the first corpus with annotation of clinical relations and co-reference.

Entity extraction. As reported in Chapter 3, the work has evaluated the relative performance of machine learned statistical models of entities, and dictionary based lookup of entities, for clinical text.

Relation extraction. As reported in Chapter 4, the work demonstrates the novel application of statistical machine learning techniques to the extraction of clinical relationships, negation and other modifiers. Taken with the work in Chapter 3, this constitutes the first supervised ML system for clinical IE.

Chapter 2

Building a semantically annotated corpus of clinical texts

Foreword

The following Chapter is reproduced in full from Roberts et al. (2009):

> A. Roberts, R. Gaizauskas, M. Hepple, G.Demetriou, Y. Guo, I. Roberts, and A. Setzer. Building a semantically annotated corpus of clinical texts. *Journal of Biomedical Informatics*, 42(5):950–66, October 2009

Preliminary work that contributed to Roberts et al. (2009) appeared in Roberts et al. (2007) and Roberts et al. (2008b):

> A. Roberts, R. Gaizauskas, M. Hepple, N. Davis, G. Demetriou, Y. Guo, J. Kola, I. Roberts, A. Setzer, A. Tapuria, and B. Wheeldin. The CLEF Corpus: Semantic Annotation of Clinical Text. In *Proceedings of the 2007 American Medical Informatics Association Annual Symposium*, pages 625–629, Chicago, IL, USA, 2007

> A. Roberts, R. Gaizauskas, M. Hepple, G. Demetriou, Y. Guo, A. Setzer, and I. Roberts. Semantic annotation of clinical text: The CLEF corpus. In *Proceedings of Building and evaluating resources for biomedical text mining: workshop at LREC 2008*, Marrakech, Morocco, May 2008b

Author's contribution

The author of this thesis wrote the first complete draft of Roberts et al. (2009), and of the two earlier papers (Roberts et al., 2007, 2008b), and led the writing of all subsequent drafts. The author made the following contributions to the work described in the paper:

- designed and executed document stratification and selection (Section 2.4);
- contributed significantly to the design of the annotation schema (Section 2.5);
- led the development and writing of the annotation guidelines (Sections 2.5.1 to 2.5.4);
- contributed significantly to the design of the methodology (Section 2.5.5);
- contributed significantly to the design of the evaluation software, which is reported separately in Demetriou et al. (2008) (Section 2.5.5);
- trained the annotators and managed the annotation effort (Section 2.5.5);
- carried out the evaluations and analysis (Section 2.6);
- constructed the final corpus (Section 2.7);
- developed of the CLEF IE system introduced in this paper, and which is detailed further in Chapters 3 and 4 (Section 2.9).

The author did not contribute to the following work described in the paper:

- annotation of UMLS CUIs (Section 2.5.6);
- temporal annotation (Section 2.8).

Copyright and permission to use

Supplementary information

Corpus description

The original paper given in this Chapter did not include summary information of the different corpora discussed, and did not give the relative sizes of the corpora as word counts. This information is provided here, in Table 2.1. The corpora described in Table 2.1 are also those used in the work reported in Chapters 3 and 4.

Access to the corpus is restricted at the time of writing. A mock example letter written by clinicians working on the CLEF project, and in the style of actual corpus letters, is given in Appendix A. Examples of short pieces of text in the style of the corpus may also be found throughout the annotation guidelines in Appendix B.

Annotation guidelines

The annotation guidelines described in Section 2.5.1 are given in Appendix B. The consensus annotation guidelines described in Section 2.5.5.3 are given in Appendix C.

Document type	Number of documents	Cross-reference	Word counts				
			Total	Minimum	Maximum	Mean	Median
Whole corpus							
Narratives	364384	Chapter 2	63384028	2	3821	173.95	147
Histopathology	17211	Chapter 2	1679672	1	935	97.59	74
Imaging	214457	Chapter 2	12192665	1	591	56.85	39
Gold standard: stratified random sample							
Narratives	50	Chapter 2 and Table 2.13	8981	16	399	179.62	153
Histopathology	50	Chapter 2 and Table 2.14	3957	21	358	79.14	71
Imaging	50	Chapter 2 and Table 2.15	3164	4	248	63.28	42
Gold standard: Stratified random sample, whole patient records, and additional documents							
Narratives	77	Section 2.9, Chapters 3 and 4	15530	9	1173	201.69	154
Histopathology	52	Not reported	4124	21	358	79.31	71
Imaging	64	Not reported	4181	4	248	65.33	43

Table 2.1: Corpus description, giving number of documents and word counts for different portions of the CLEF corpus, and cross-references to descriptions of each portion in the text. Word counts are counts of white space delimited tokens. The *whole corpus* refers to all documents, whether annotated or not. The *gold standard* portions refer to those portions of the whole corpus that were manually annotated. For some experiments, a gold standard consisting of a *stratified random sample* of the whole corpus was used. For other experiments, this stratified random sample was combined with a sample consisting of all documents for two whole patient records, and a small number of additional randomly drawn documents that had been annotated over and above the original numbers required. Note that the documents described in Table 1.1 are an initial portion of the whole corpus narratives described here, and that the total number of documents referred to in Section 2.4.1 is an approximate count of all document types for the whole corpus.

2.1 Abstract

In this paper, we describe the construction of a semantically annotated corpus of clinical texts for use in the development and evaluation of systems for automatically extracting clinically significant information from the textual component of patient records. The paper details the sampling of textual material from a collection of 20,000 cancer patient records, the development of a semantic annotation scheme, the annotation methodology, the distribution of annotations in the final corpus, and the use of the corpus for development of an adaptive information extraction system. The resulting corpus is the most richly semantically annotated resource for clinical text processing built to date, whose value has been demonstrated through its use in developing an effective information extraction system. The detailed presentation of our corpus construction and annotation methodology will be of value to others seeking to build high-quality semantically annotated corpora in biomedical domains.

2.2 Introduction

We describe the creation of a semantically annotated corpus of clinical texts. The documents of this corpus are drawn from the free text component of patient records, and the annotations capture clinically significant information communicated by these texts. The corpus is intended for use in developing and evaluating systems that can *automatically* extract this kind of clinically significant information from the textual component of patient records. The corpus has been created within the context of the CLinical E-Science Framework (CLEF) project (Rector et al., 2003): a multi-site research project that has been developing the technology and techniques required for a high quality repository of electronic patient records. Such a repository must meet high standards of security and interoperability, and should enable ethical and user-friendly access to patient information, so as to facilitate both clinical care and biomedical research. CLEF has chosen to work in the area of cancer informatics, as one of the project partners–the Royal Marsden Hospital (RMH)–is a large specialist oncology centre.

Although much of the patient information needed to populate such a repository exists as structured data, e.g. database records of drug prescriptions and clinic appointments, free text material still forms an important component of electronic patient records, and contains information that is potentially significant both for day-to-day care and clinical research. For example, letters written from the secondary to the primary care physician (e.g. from specialist consultant to patient GP) form a major component of any UK medical record, and free text plays a key role in the reporting of imaging and pathology findings.

Clinical narratives may record, for instance, why drugs were given or discontinued, the results of physical examination, and issues considered important when discussing patient care but which are not coded for audit. Such information, when combined with that from the structured record, and suitably presented, could contribute to individual patient care, e.g. providing a consultant with a concise summary of their patient's clinical history, or access to concise histories for patients with similar conditions elsewhere. Aggregation of information across all the records in a large repository could bring benefits for clinical research. For example, being able to get answers to questions such as "*How many patients with stage 2 adenocarcinoma who were treated with tamoxifen were symptom-free after 5 years?*" could assist a researcher in formulating hypotheses that could be later explored in clinical trials.

The need to make the information that exists in clinical texts available for integration with the structured record, for subsequent use in clinical care and research, has been addressed within CLEF through the use of *information extraction* (IE) technology (Grishman, 2003; Harkema et al., 2005). Although some IE research has focused on unsupervised methods of developing systems, as in the earlier work of Riloff (1996), most practical modern IE work requires data that have been manually annotated with the events, entities and relationships that are considered to express key content for the given domain. These data serve three purposes. First, the analysis of data that is required to create the annotation scheme serves to focus and clarify the information requirements of the task and domain. Second, the annotated data provide a *gold standard* against which to assess the performance of systems designed to automatically identify this information in texts. Third, it serves as a resource for system development: extraction rules may be created either automatically or by hand, and statistical models of the text may be built by machine learning algorithms.

This paper reports on the work done within CLEF to create an annotated corpus, to aid the development and evaluation of the CLEF IE system. To the best of our knowledge, no one else has explored the problem of producing a corpus annotated for clinical IE to the depth and extent reported here, and the resulting corpus is the most richly semantically annotated resource for clinical text processing built to date. Our annotation exercise draws its texts from a large background corpus of clinical narratives, covers multiple text types, and involves over 20 annotators. Results are encouraging, and suggest that a rich corpus to support IE in the medical domain can be created.

We reported the early development of the CLEF corpus in Roberts et al. (2007). The current paper elaborates quantitative results from this development process, giving a much greater level of detail. Quantitative results have also previously been given, for the partially complete corpus, in Roberts et al. (2008b). The results in the current paper are

final, reflecting the finished corpus. In addition, the current paper provides results and descriptions not previously published, including: annotation with UMLS CUIs; annotation of temporal expressions; the summary results of an annotator difference analysis; a discussion of time taken to annotate; detailed descriptions of the annotation guidelines, their development and application; and greater detail of our annotation methodology. We also summarise work on the corpus in use, to train and evaluate a working IE system. We believe that this detailed account of our methodology, corpus, and its use will be of benefit to other groups contemplating similar exercises.

The paper is organised as follows: in the next section, we summarise previous efforts to create annotated corpora in biomedical domains. Section 2.4 describes how material was selected for inclusion in our corpus, and then in Section 2.5, we describe the semantic annotation schema, the annotation methodology, the development of the annotation guidelines, as well as the measures for assessing the consistency of human annotations. Section 2.6 presents an analysis of aspects of the annotation process and Section 2.7 presents inter annotator agreement scores for the finished corpus, and figures on the distribution of entity and relation types by document type across the corpus. The next section describes work carried out subsequent to the initial corpus construction work, to add a layer of temporal annotation. Finally, in Section 2.9, we mention on-going use of the corpus for training and evaluation of our supervised machine learning IE system.

2.3 Annotated corpora for biomedical research

Annotated corpora, or text collections, are now recognised as resources of central importance in biomedical language processing research. They may be taxonomized in various ways. For example, they can be grouped by domain (e.g protein-protein interactions and oncology), document type or genre (e.g. research article, clinical narrative, and radiology report), type of annotation (e.g. semantic–entities, relations and/or syntactic–part-of-speech, parse structure), intended language processing application (e.g. information extraction, text classification), intended mode of use (e.g. for training adaptive systems, for specific system evaluation, for community wide shared task evaluation), or availability (e.g. publicly available or not publicly available). It is not our intention to attempt a complete characterisation and review of all annotated corpus resources that have been used in biomedical language processing research. Instead we focus on a few that enable us to show where the CLEF corpus fits in the context of prior research and what novel contribution it makes.

The CLEF corpus may be characterised as a semantically annotated corpus of clinical documents of mixed type (clinic letters, radiology, and histopathology reports) which

is designed to support both automated training and evaluation of information extraction systems. While it is not publicly available at time of writing we are working towards its release (see below) and reusability has been an important consideration informing its design.

There are now a significant number of publicly available semantically annotated corpora designed to support information extraction research comprising texts drawn from the biomedical research literature. For example, the GENIA corpus is a collection of ~200 MEDLINE abstracts in the area of molecular biology that has had mentions of specific biological entities and events annotated within it (Kim et al., 2003, 2008). The PennBioIE corpus (Kulick et al., 2004) consists of ~2300 MEDLINE abstracts, in the domains of molecular genetics of oncology and inhibition of enzymes of the CYP450 class and is annotated for biomedical entity types (it is also annotated syntactically for parts-of-speech amd some portion of it has been annotated for Penn Treebank style syntactic structure). The Yapex corpus contains 200 MEDLINE abstracts annotated for protein names (Franzén et al., 2002). The BioText project has made several semantically annotated corpora available, including one for disease-treatment relation classification consisting of ~3500 sentences drawn from MEDLINE abstracts labelled for DISEASE and TREATMENT and seven types of relation holding between them (Rosario and Hearst, 2004), and one for protein-protein interaction classification consisting of ~800 sentences drawn from full-text journal papers, where each sentence contains mentions of an interacting protein pair (Rosario and Hearst, 2005). The ITI TXM corpus (Alex et al., 2008) has annotated tissue expressions in 238 full-text documents drawn from PubMed and protein-protein interactions in 217 documents obtained from Pub MedCentral and PubMed.

While these corpora have been developed in the contexts of specific research projects they have been developed with a view to reusability and have been released to the wider research community. Other semantically annotated corpora drawn from the biomedical research literature have been developed specifically for the purpose of shared task evaluations of information extraction systems. These evaluations include the Biocreative challenge, which utilised the GENETAG corpus containing 20,000 sentences with gene/protein names annotated (Tanabe et al., 2005), the LLL05 challenge task, which supplied training and test data for the task of identifying protein/gene interactions in sentences from MEDLINE abstracts (Nédellec, 2005b), and the TREC Genomics Track, which, while focussed on information retrieval rather than information extraction, did yield some datasets which could be viewed as semantically annotated, e.g. the TREC 2007 task for which human relevance judgements include lists of domain-specific entities associated with relevant passages (TREC, 2008).

The corpora mentioned so far consist of texts drawn from the research literature. Corpora consisting of clinical texts, e.g. clinic letters, radiology, and histopatholgy reports, are much rarer–getting access to clinical text for research purposes is difficult due to issues of patient confidentiality and getting permission to release them to the wider research community is even more challenging. To our knowledge the only annotated corpora intended to support research in clinical information retrieval and extraction that have been released to the wider research community are those developed in the context of several recent shared task challenges. For example, the corpus prepared and released for the Computational Medicine Challenge (Pestian et al., 2007) consists of 1954 (978 training and 976 test) radiology reports annotated with ICD-9-CM codes, where the challenge is to automatically code the unseen test data. The ImageCLEFmed 2005 and 2006 image test collections consist of ~50,000 images with associated textual annotations (case descriptions and imaging reports) and in some cases metadata (e.g. DICOM labels), together with query topics and relevance judgements (Hersh et al., 2006; Müller et al., 2007). While intended to support medical image retrieval research, the textual component of this resource could have purely language processing applications. Finally, the I2B2 challenges, have provided training and evaluation data for de-identification of discharge summaries, the identification of smoking status from discharge summaries, and the identification of obesity and co-morbidities from discharge summaries (I2B2, 2007).

These are the only publicly released semantically annotated clinical corpora of which we are aware. However, various research projects have developed and published descriptions of clinical corpora used for training and/or evaluation within their project which may be viewed as "semantically annotated" in some sense. Ogren et al. (2006), for example, describe work on annotating disorders within clinic notes with a view to training and testing a named entity recognition system. Meystre and Haug (2006) describe the development of corpus of 160 clinical documents of mixed type (diagnostic procedure reports, radiology reports, history and physicals, etc.) in which medical problems are identified manually for use in evaluating their system which attempts to extract a patient "problem list" from a clinical document. However it appears that specific mentions of these problems are not annotated where they occur in the text, but rather that problems are associated with a text at document level, reducing the utility of the corpus for supervised learning. Denny et al. (2003) construct a "gold standard" corpus of medical school lecture documents in which biomedical concepts have been manually identified for use in evaluating their KnowledgeMap tool which aims to automatically identify such concepts. Again it appears that in the gold standard the concepts are associated with the text at document level, rather than at the mention level within the running text. Assessing the ability to correctly identify the negations of clinical concepts in clinical texts is the focus

of a study by Elkin et al. (2005) who have manually verified whether the clinical concepts in a set of 41 clinical documents are negated or not, yielding an annotated evaluation resource for concept negation in clinical texts. Of course the long history of interest in constructing clinical information extraction systems has left a correspondingly long series of gradually maturing evaluations of these systems many of which produced evaluation resources that can be viewed as semantically annotated corpora. Friedman and Hripcsak (1998) present an extensive review of work on evaluating natural language processing systems in the clinical domain, especially information extraction systems, prior to 1998, including discussion of any evaluation resources these evaluations have produced.

The CLEF corpus may be differentiated from the annotation work mentioned above in several regards. First, so far as we are aware, it is the first corpus of clinical texts to be annotated with information about clinical relations as well entities. Second, the range of entity types for which all mentions are annotated in the running text, as opposed to merely being associated with the text at document level is much wider than in previous efforts, making the resource of significantly greater utility for supervised learning. Third, it is the first biomedical corpus to be annotated with temporal information. Taken together these features make the CLEF corpus the richest semantically annotated corpus of clinical texts yet developed. Finally, it is worth mentioning that the corpus has been designed with a view to reuse by using standards such as XML for the markup and by producing documentation for others to use, something that differentiates it from many project-specific evaluations.

2.4 Selection of corpus material

Our corpus comes from CLEFs main clinical partner, the Royal Marsden Hospital, Europe's largest specialist oncology centre. The entire corpus consists of both the structured records and free text documents from 20,234 deceased patients. The free text documents are of three types: clinical narratives (with sub-types as shown in Table 2.2); histopathology reports; and imaging reports. Patient confidentiality is ensured through a variety of technical and organisational measures, including automatic pseudonymisation and manual inspection. Approval to use this corpus for research purposes within CLEF was sought and obtained from the Thames Valley Multi-centre Research Ethics Committee (MREC).

Document		Diagnosis						Total
Type	**Subtype**	**Digestive**	**Breast**	**Haema-tology**	**Respira-tory**	**Female genital**	**Male genital**	
Narrative	**To GP**	9.41	12.36	11.59	5.63	4.64	4.91	48.56
	Discharge	7.08	2.74	1.75	2.27	2.63	0.52	16.98
	Case note	4.25	2.95	2.07	1.96	2.41	1.07	14.72
	Other letter	1.92	1.57	1.30	0.76	0.83	0.50	6.88
	To consultant	1.31	2.04	0.75	0.80	0.61	0.25	5.77
	To referer	1.50	0.40	0.32	0.65	0.37	0.32	3.56
	To patient	0.57	0.95	0.21	0.25	0.33	0.30	2.60
	Report	0.15	0.20	0.14	0.11	0.11	0.02	0.72
	Audit	0.01	0.18	0.00	0.01	0.00	0.00	0.21
Narratives total		26.21	23.38	18.13	12.45	11.94	7.89	100.00
Imaging	**CT scan**	10.00	3.58	3.99	3.45	4.84	1.64	27.51
	Mammogram	0.02	1.03	0.03	0.02	0.02	0.00	1.11
	MRI	0.51	0.82	0.45	0.32	0.16	0.62	2.88
	Ultrasound	1.81	3.76	1.28	0.60	1.30	0.48	9.24
	X-ray	11.64	13.35	15.30	9.82	5.38	3.78	59.27
Imaging total		23.98	22.54	21.04	14.22	11.70	6.51	100.00
Histopathology (all)		22.74	18.48	28.94	6.49	15.9	7.44	100.00

Table 2.2: Percentage of all CLEF documents by diagnosis and document sub-type

2.4.1 Document sampling

Given the expense of human annotation, the annotated portion of the corpus–which we refer to as the gold standard corpus–has to be a relatively small subset of the whole corpus of 565,000 documents. In order to avoid events that are either rare or outside of the main project requirements, the gold standard is restricted by diagnosis, and only considers documents from those patients with a primary diagnosis code in one of the top level sub-categories of ICD-10 Chapter II (neoplasms) (WHO, 2008). In addition, it only contains those sub-categories that cover more than 5% of the total number of narratives and reports in the whole corpus. The gold standard corpus consists of three portions, selected for slightly different purposes.

2.4.1.1 Whole patient records

Two applications in CLEF involve aggregating data across a single patient record. The CLEF chronicle builds a chronological model for a patient, integrating events from both the structured and unstructured record (Rogers et al., 2006). CLEF report generation creates aggregated graphical and textual reports from the chronicle (Hallett et al., 2006). These two applications require whole patient records for development and testing. Two whole patient records were selected for this portion of the corpus, from two of the major diagnostic categories, to give median numbers of documents, and a mix of document types and lengths. For each patient, the record comprises nine narratives, one imaging report and seven histopathology reports, plus associated structured data.

2.4.1.2 Stratified random sample

The major portion of the gold standard serves as development and evaluation material for IE. In order to ensure even training and fair evaluation across the entire corpus, the sampling of this portion is randomised and stratified, so that it reflects the population distribution along various axes. Table 2.2 shows the proportions of clinical narratives along two of these axes. The random sample consists of 50 each of clinical narratives, histopathology reports, and imaging reports.

The numbers of documents chosen for annotation were based on two factors. First, preliminary experiments using documents annotated with a small number of entity types had shown that performance of an adaptive IE system plateaued with around 40 documents used for training. Second, from a purely pragmatic point of view, we only had a limited amount of annotator time. We used empirically based estimates of the time taken to annotate each document, to calculate the number of documents we could annotate in the time available. Time for annotator training was factored in.

Thirty-two documents of mixed type were also randomly chosen for use in annotator training and guideline development. These documents were annotated, but were not used as part of the final gold standard.

2.4.1.3 Development corpus

The stratified random corpus was only ever examined by annotators, and not by system developers, who remained blind to its contents throughout. This policy was implemented to avoid there being any developments of the system which were cued specifically by the characteristics of documents that might ultimately be used in scoring the system's performance, as this would contaminate the evaluation.

It is, however, essential for developers to have some documents to work with. A "mirror" corpus of the stratified random corpus was therefore created. This consisted of different documents, but with the same document types, and stratified in the same proportions along the same axes. This corpus was never annotated. It was available to system developers as required.

2.5 The CLEF annotation schema and its development

The CLEF gold standard is a semantically annotated corpus. We are interested in identifying the key clinical entities mentioned in the text. By entity, we mean some real-world thing or occurrence referred to in the text such as the drugs that have been administered, the tests that were carried out, etc. We are also interested in determining the relationships between entities: the condition indicated by a drug, the result of an investigation, etc.

Annotation is anchored in the text. Annotators mark spans of text with a type: drug, locus, and so on. Annotators may also mark words that modify spans (such as negation), and mark relationships as links between spans. Two or more spans may refer to the same entity in the real world, in which case they co-refer. Co-referring CLEF entities are linked by the annotators. An example illustrating some aspects of annotation is shown in Fig. 2.1. The types of annotation are described in a schema, shown in Fig. 2.2. The CLEF entities, relations, modifiers, and co-reference are also listed in Tables 2.3 and 2.4, along with descriptions and examples.

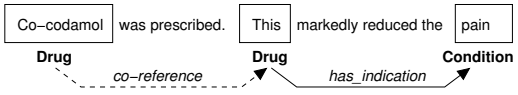

Figure 2.1: Annotations, co-reference, relationships.

Entity type	Description	Examples
Condition	Symptom, diagnosis, complication, conditions, problems, functions and processes, injury.	• This patient has had a lymph node biopsy which shows melanoma in his right groin. • It is clearly secondaries from the melanoma on his right second toe.
Intervention	Action performed by doctor or other clinician targeted at a patient, **Locus**, or **Condition** with the objective of changing (the properties) of, or treating, a **Condition**.	• Although his PET scan is normal he does need a groin dissection. • We agreed to treat with DTIC, and then consider radiotherapy.
Investigation	Interaction between doctor and patient or **Locus** aimed at measuring or studying, but not changing, some aspect of a **Condition**. **Investigations** have findings or interpretations, whereas **Interventions** usually do not.	• This patient has had a lymph node biopsy ... • Although his PET scan is normal he does need a groin dissection. • We will perform a CT scan to look at the left pelvic side wall ...
Result	The numeric or qualitative finding of an **Investigation**, excluding **Condition**.	• Although his PET scan is normal ... • Other examples include the numeric values of tests, such as "80mg".
Drug or device	Usually a drug. Occasionally, medical devices such as suture material and drains will also be mentioned in texts.	• This pain was initially relieved by co-codamol.
Locus	Anatomical structure or location, body substance, or physiologic function, typically the locus of a **Condition**.	• This patient has had a lymph node biopsy which shows melanoma in his right groin ... • It is clearly secondaries from the melanoma on his right second toe. • Although his PET scan is normal he does need a groin dissection. • We will perform a CT scan to look at the left pelvic side wall.

Table 2.3: CLEF entities. In the examples, mentions of the entity type are underlined. Adapted from the CLEF Annotation Guidelines (see Availability).

Relation type	First argument type	Second argument type	Description	Examples
has_target	**Investigation** **Intervention**	**Locus**	Relates an intervention or an investigation to the bodily locus at which it is targetted.	• This patient has had a [arg2] lymph node [arg1] biopsy • . . . he does need a [arg2] groin [arg1] dissection
has_finding	**Investigation**	**Condition** **Result**	Relates a condition to an investigation that demonstrated its presence, or a result to the investigation that produced that result.	• This patient has had a lymph node [arg1] biopsy which shows [arg2] melanoma • Although his [arg1] PET scan is [arg2] normal . . .
has_indication	**Drug or device** **Investigation** **Intervention**	**Condition**	Relates a condition to a drug, intervention, or investigation that is targetted at that condition.	• Her facial [arg2] pain was initially relieved by [arg1] co-codamol
has_location	**Condition**	**Locus**	Relationship between a condition and a locus: describes the bodily location of a specific condition. May also describe the location of malignant disease in lymph nodes, relating an involvement to a locus.	• . . . a biopsy which shows [arg1] melanoma in his right [arg2] groin • It is clearly secondaries from the [arg1] melanoma on his right [arg2] second toe • Her[arg2] facial [arg1] pain was initially relieved by co-codamol
Modifies	**Negation signal**	**Condition**	Relates a condition to its negation or uncertainty about it.	• There was [arg1] no evidence of extra pelvic [arg2] secondaries
Modifies	**Laterality signal**	**Locus** **Intervention**	Relates a bodily locus or intervention to its sidedness: *right, left, bilateral*.	• . . . on his [arg1] right [arg2] second toe • [arg1] right [arg2] thoracotomy
Modifies	**Sub-location signal**	**Locus**	Relates a bodily locus to other information about the location: *upper, lower, extra*, etc.	• [arg1] extra [arg2] pelvic
Co-refers	**Any**	**Any**	Relates two spans of text where they refer to the same entity in the real world. Includes both lexical co-reference and co-reference that requires domain knowledge, as in the examples.	• [arg1] Haemoglobin 7.5g/dl. Given this [arg1] Hb, treatment was postponed. • He has a [arg1] melanoma. The [arg1] tumour is in his 2nd toe.

Table 2.4: CLEF relations, modifiers, and co-reference. Each example shows a single relation of the given type. Arguments are underlined and preceded by their argument number. Adapted from the CLEF Annotation Guidelines (see Availability).

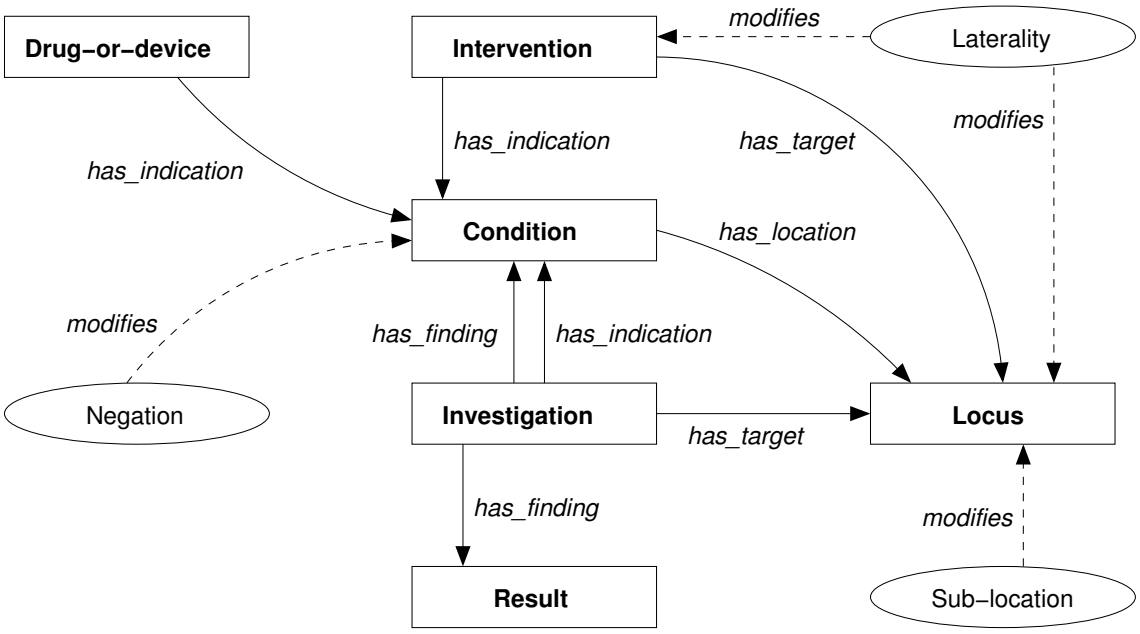

Figure 2.2: The CLEF annotation schema. Rectangles: entities; ovals: modifiers; solid lines: relationships; dotted lines: modifier relationships.

Relationships include those that are obvious from the linguistic structure of the text, and those that need some level of domain knowledge to infer. As an example of the latter, consider the example: *"FBC and U&E were requested. She was severely anaemic"*. In this, knowledge is required to infer that there is a relationship `FBC has_finding anaemia`. In practice, the distinction between linguistic and domain knowledge is blurred, and it proves difficult to decide which relationships are based on which type of knowledge. We have therefore made no attempt to differentiate between these two categories of relationship in our schema, taking the view that such a distinction could be added as a separate layer of annotation if required.

The schema is based on a set of requirements developed between clinicians and computational linguists in CLEF. The schema types are mapped to types in the UMLS semantic network, which enables us to utilise UMLS vocabularies in entity recognition. The aim of annotation was to provide general semantic types for entities, and not to map entities to any particular codified terminology. Mapping to specific terminologies was considered to be an extra layer of annotation, performed for specific applications that require it, as described in Section 2.5.6. For the purposes of annotation, the schema is modelled as a Protégé-Frames ontology (Gennari et al., 2003). Annotation is carried out using an adapted version of the Knowtator plugin for Protégé (Ogren, 2006). This was chosen for its handling of relationships, after evaluating several such tools.

2.5.1 The annotation guidelines

Consistency is critical to the quality of a gold standard. It is important that all documents are annotated to the same standard. Questions regularly arise when annotating. For example, should multi-word expressions be split? Should "myocardial infarction" be annotated as a condition only, or as a condition and a locus? To ensure consistency, a set of guidelines is provided to annotators. These describe in detail what should and should not be annotated; how to decide if two entities are related; how to deal with co-reference; and a number of special cases. The guidelines also provide a sequence of steps, a recipe, which annotators should follow when working on a document. This recipe is designed to minimise errors of omission. The guidelines themselves were developed through a rigorous, iterative process, which is described below.

2.5.2 The origin of the guidelines

The guidelines originated from IE *template* definitions, in an initial CLEF IE system (Harkema et al., 2005), which were themselves patterned on the set of template definitions used in the Message Understanding Conferences (see e.g. NIST (d)). A template is a structured object representing domain-specific entities, their properties, and the relationships between them. A template represents something in the real world. The template does not, however, relate directly to a specific span of text: it is independent of the text. A template may be instantiated, even though the entity it describes is not directly mentioned in the text. For example, a text that discusses angina could lead to a `heart` template being created.

The CLEF templates modelled a large and ambitious set of nine entities with sixteen different relationships between them. Each entity also had a number of properties that were to be extracted, for example, the `course` of a `condition`, or the `goal` of an `intervention`. The entities and relationships were themselves based on an ontology that attempted to model every aspect of the patient and treatment, as described in the clinical documents.

The template definitions were drawn up in collaboration with a single medical informatician, and were tested by the same medical informatician, by manually filling the templates for a small number of documents. This set of documents became a gold standard for system development and testing. With use, a number of problems became apparent in this gold standard. First, although there was a good formal description of how templates should be filled, there was no description of how they should be created. Should a single template be created for every mention of a patient's bladder, or should just one be created? This led to template construction that was idiosyncratic, and at odds with the

requirements of information extraction. Second, the complexity of the ontology, the resulting templates, and the limitations of the tools used (text editors), meant that template filling was slow and painful. This in turn led to insufficient data for system development and testing. Lastly, templates are not anchored in the text. This means that when comparing a template in the gold standard to a template created by a IE system, we must first decide whether they are referring to the same thing. For example, suppose a text mentions the two distinct kidneys of a patient, and as a consequence, in the gold standard there are two `kidney` templates instantiated. If an IE system only finds a single `kidney` template, then a choice needs to be made as to which of the two gold standard templates it must be aligned with for evaluation.

Taken together, the problems we encountered meant that it was difficult to decide if evaluation scores reflected the system being evaluated, or some problem in the gold standard. The problems that we identified with our template model are in part inherent to the template representation, and in part due to the complexity of our specific template model. As originally used in the Message Understanding Conferences (NIST, d), templates are independent of the text: a product of research into full text understanding systems. Our simpler task is to extract those entities and relations explicitly mentioned in the text. This task is better served by a representation that anchors those entities and relations directly to the text.

2.5.3 Developing the guidelines

As a consequence of these difficulties, it was decided to create a new gold standard consisting of textually-anchored annotations, rather than templates. This would make evaluation easier, would simplify supervised learning using annotated text, and would also mean that one of the dedicated tools available for this style of annotation could be used. A larger number of documents would be annotated with a simplified set of entities and relations, and these would be described in explicit, methodically developed guidelines. The guidelines would be developed by a team of clinicians and computational linguists, and would be tested against a significant number of documents, before use for annotation of the final gold standard.

The starting points for the writing of the guidelines were the original ontology and template definitions. These were simplified to give an initial set of six entities and six relations, plus two modifiers (later additions changed this to the schema presented in this paper, as shown in Fig. 2.2). The entities and relationships were agreed between a small group of computational linguists and clinicians. An initial draft set of guidelines describing the entities and relationships were then drawn up, and discussed by a larger

group.

The guidelines were developed and refined using an iterative process, designed to ensure their consistency. This is shown in Fig. 2.3. Two qualified clinicians annotated different sets of documents in five iterations (covering 31 documents in total). We measured the agreement between annotators according to a number of metrics which are defined below in Section 2.5.5.2. Agreement for these iterations are shown in Table 2.5. As can be seen, agreement remains consistently high after the five iterations, after which very few amendments were required to the guidelines. Relation agreement does not appear so stable on iteration 5. Difference analysis showed that over half of the difference was due to a single, simple type of disagreement across a limited number of sentences in one document. One annotator had co-referred mentions with a plural or set that encompassed that mention. For example, "nail of the right thumb" has been co-referred with "all of the hand nails". Scoring without this document gave a much improved level of agreement.

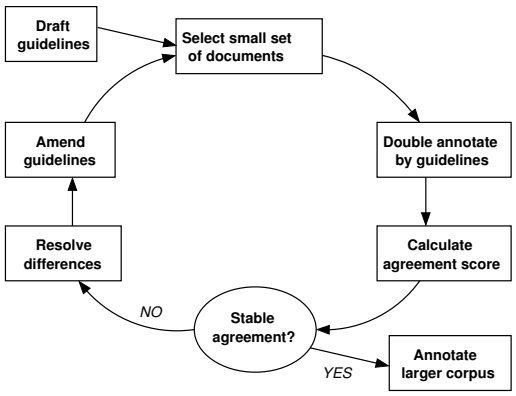

Figure 2.3: Iterative development of guidelines.

		Debug iteration				
		1	**2**	**3**	**4**	**5**
Entities	**Matches**	244	244	308	462	276
	Partial matches	2	6	22	6	1
	Non-matches	45	32	93	51	22
	IAA	**84**	**87**	**74**	**89**	**92**
Relations	**Matches**	170	78	116	412	170
	Partial matches	3	5	14	6	1
	Non-matches	31	60	89	131	103
	IAA	**84**	**56**	**56**	**75**	**62**

Table 2.5: Lenient inter annotator agreement (IAA, %) for each guideline development iteration of five documents. During development, IAAs were calculated using the Knowtator annotation tool.

During each development iteration, the clinician annotators made notes on the clarity of the guidelines, and on the relevance of the resulting annotations. At the end each iteration, a difference analysis was performed on the two sets of annotations, listing points

of difference between the two annotators. The annotator notes and the difference analysis were fed into a post-iteration discussion, which informed a rewrite of the guidelines. Many of the changes consisted of either minor clarifications, or the addition of informative examples. Occasionally, major changes were made. For example, it had been intended to annotate any discussion of lymph node involvement. However, no examples were found in the development documents, and the few examples found in a larger selection of the entire CLEF corpus were difficult to interpret. In another example, it was thought that `Investigation` entities would always stand in a `has_finding` relations to an entity type of `Condition`. However, this proved false, and the schema was augmented with a new entity type of `Result`, when it was realised that not all cases could be annotated in this way.

2.5.4 The guidelines as a tool

The guidelines are written as a *wiki*: a set of hyperlinked web pages that can be edited and created by anyone who has access to them. Use of a wiki means that the guidelines can be edited, corrected, and updated by a number of people involved in their writing. Although written in this way, the guidelines are provided to annotators as a read-only web site. Publication as a web site meant that the guidelines were dynamic and hyperlinked. The dynamic nature of the site meant that as guidelines were updated, annotators would always be accessing the latest version. Pages of "news" were provided to publicise recent changes, and to answer common queries. Sample pages from the web site are shown in Fig. 2.4.

The hyperlinked nature of the guidelines is in contrast to the more common method of presenting annotation guidelines as a technical document. Hyperlinking meant that annotators could quickly navigate them, finding the relevant section for their work, and could easily move to related sections. For example, an annotator thinking about how to annotate the `has_location` relation, could easily jump to the section about the `Locus` entity, an argument of that relation, via hyperlinks on every mention of `Locus` on the `has_location` pages. In addition to hyperlinks within pages, each page was provided with a top level menu bar, giving access to tables summarising the guidelines, and to the top level sections. Links for the next and previous page were also provided, so that the guidelines could be read in a linear style if required.

The idea of guidelines-as-a-tool is also reflected in the writing style. Writing is in an easily digested style with short sentences, heavy use of bullet points, tables, examples, and sub-sections. The aim is to present the information clearly, and in a quickly accessible form. Annotators work with the guidelines open in a web browser, switching back and

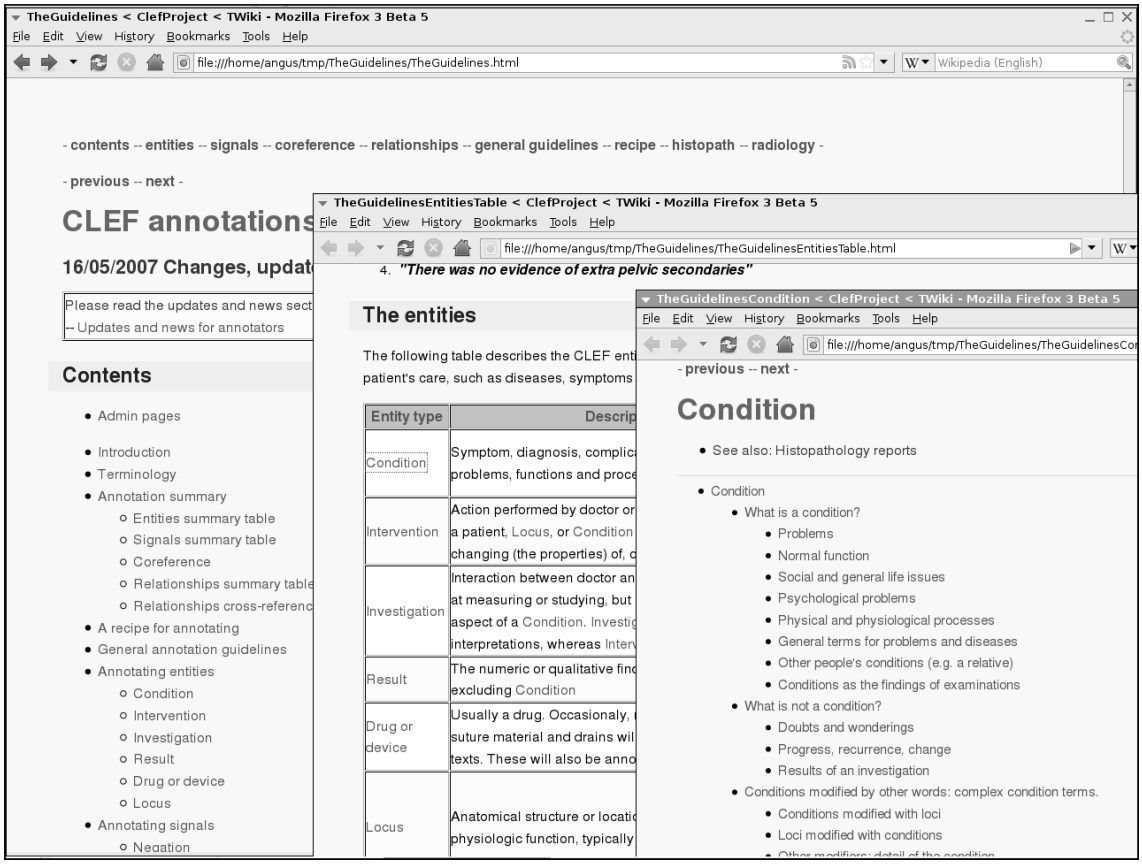

Figure 2.4: The CLEF Annotation Guidelines web site. From a window showing the menus and contents, the user has opened a table of all entities, and from this window has opened the `Condition` guidelines.

forth from the guidelines to their annotation tool. The guidelines comprise nine main sections:

1. **News:** a section describing recent changes to the guidelines, answers to common questions, and other annotation-related news items.
2. **Terminology:** a table giving definitions and examples of the technical terms used in annotation, such as *Entity*, *Co-reference*, etc.
3. **Summary tables:** of entities, modifiers, and relations, each type with a description, examples, and hyperlinks to the relevant guidelines. Tables 2.3 and Tables 2.4 are adapted from these.
4. **A *recipe*** for annotating: a step-by-step guide of how to read a document and mark the relevant annotations. This recipe was independent of the annotation tool used.
5. **General guidelines:** that give a high-level philosophy of what should and should not be annotated.
6. **Entity guidelines:** specific guidelines for each entity.
7. **Relation guidelines:** specific guidelines for each relation.
8. **Modifier guidelines:** specific guidelines for each modifier.
9. **Report guidelines:** guidelines specific to histopathology and imaging reports.

The annotation recipe describes in detail how a document should be annotated. It was expected that a consistent annotation method would produce more consistent annotations. In reality, however, it is difficult to supervise annotation, and so it is not clear whether annotators always adopted the recipe, or opted for faster shortcut methods of annotation. The recipe is summarised below:

1. Read the document through in its entirety, marking no annotations, to get an understanding.
2. Read the document a second time, adding annotations for the mentions (including pronouns) of the entities.
3. Go through each of the conditions, loci, and interventions, checking for modifiers, qualifications, and associated text that signify further annotations.
4. Go through each of the mentions in turn, and check to see if it co-refers with any other mention.

5. Go through each of the mentions in turn, and decide if any have relationships with other entities.

6. Record any questions, uncertainties, ambiguities, tool bugs and issues.

The general guidelines give a high-level philosophy of what should and should not be annotated. They discuss issues such as whether to annotate overlapping terms; how and when complex terms should be broken down into their component parts; how to treat conjunctions; whether annotator domain knowledge may be applied to infer relationships, or whether they should be clearly stated in the text.

Each entity, relationship, and modifier has a single web page detailing specific guidelines for that annotation. These pages have a consistent format. For entities, the page first lists the kinds of things that should be annotated as this entity type, each with an example. This is followed by the kinds of things that should not be annotated, again with examples. The next section describes how mentions of this entity type take part in complex phrases, and how they are modified by other words. Other sections may follow, specific to the entity type. For relations, the possible arguments are first described, in tabular form. This is followed by further sections, discussing for example: when entities do and do not take part in this relation type; the use of clinical knowledge to infer relations; whether one-to-many relations are allowed for this relation type.

2.5.5 Annotation methodology

The annotation methodology follows established natural language processing standards (Boisen et al., 2000). Annotators work to agreed guidelines; documents are annotated by at least two annotators; documents are only used where there is an acceptable level of agreement between annotators; differences are resolved by a third experienced annotator. These points are discussed further below.

2.5.5.1 Double annotation

A singly annotated document can reflect many problems: the idiosyncrasies of an individual annotator; one-off errors made by a single annotator; annotators who consistently under-perform. There are many alternative annotation schemes designed to overcome this, all of which involve more annotator time. Double annotation is a widely used alternative, in which each document is independently annotated by two annotators, and the sets of annotations compared for agreement.

2.5.5.2 Agreement metrics

Agreement between annotators is defined in terms of *matches* and *non-matches* between the two double-annotation sets created for each document, one set created per annotator. An annotation in one set matches that in the other set if they have the same type, and the same character offsets (textual span). In all other cases, the annotation is considered a non-match. For every match in the first set, there will be an equivalent match in the second set. The total number of matches is the sum of these (i.e. double the number of matches in any one set). The total number of non-matches is the sum of non-matches in each set. Agreement between double-annotated documents can then be calculated as inter annotator agreement (IAA), as in the following equation:

$$IAA = \frac{\text{matches}}{\text{matches} + \text{non-matches}} \tag{2.1}$$

We report IAA as a percentage. Overall figures are macro-averaged across all entity or relationship types. In addition to the "strict" version of IAA described above, in which entity spans must match exactly, we use a second "lenient" IAA, in which partial matches, i.e. overlaps, are counted as a half-match. Together, these show how much disagreement is down to annotators finding similar entities, but differing in the exact spans of text marked. We used both scores in development. Results given below explicitly state the score being used.

Two variations of IAA for relations were also used. First, all relationships found were scored. This has the drawback that an annotator who failed to find a relationship because they had not found one or both the entities would be penalised. To overcome this, a Corrected IAA (referred to as CIAA) was calculated, including only those relationships where both annotators had found the two entities involved. This allows us to isolate, to some extent, relationship scoring from entity scoring.

In the initial stages of the annotation exercise, during guideline development, IAA was calculated directly with the Knowtator plugin for Protégé (Ogren, 2006). During the training of annotators and "production" annotation, we wished to have a more fine-grained control over IAA calculation, giving the different types of IAA scores for different combinations of annotators and parameters, and producing hyperlinked error reports. To this end, we customised our own ANNALIST scoring tool (Demetriou et al., 2008). Unless otherwise stated, scores given in this paper have been calculated using ANNALIST.

The metrics used are equivalent to others more commonly used in IE evaluations, as shown in Table 2.6. IAA also approximates the widely used κ score, which is itself not appropriate in this case (Hripcsak and Rothschild, 2005).

Agreement metric	IE evaluation metric
Match	2 × correct
Non-match	Spurious + missing
IAA	F1 measure

Table 2.6: Equivalence of annotator agreement metrics and standard IE metrics

2.5.5.3 Difference resolution

Double annotation can be used to improve the quality of annotation, and therefore the quality of statistical models trained on those annotations. This is achieved by combining double annotations to give a set closer to the "truth" (although it is generally accepted as impossible to define an "absolute truth" gold standard in an annotation task with the complexity of CLEF's). The resolution process is carried out by a third experienced annotator, the *consensus* annotator. All agreements from the original annotators are accepted into a consensus set, and the third annotator adjudicates on differences, according to a set of strict consensus guidelines. These consensus guidelines are designed to ensure that annotations remain at least double annotated, and that the consensus annotator cannot easily overrule both of the double annotators to enforce their own single annotation. The consensus annotator cannot, for example, create new annotations that have not been previously created by one of the double annotators, and cannot delete an annotation that has been created by both double annotators. Amongst other rules, the consensus annotation guidelines rule how to deal with overlapping annotations; how to deal with annotations of the same span but different type; and how to deal with different arguments for relationship annotations.

2.5.6 Annotating CUIs

As described in Section 2.5, the CLEF entity types map to high-level types in the UMLS semantic network. This gives a coarse-grained semantic typing to entities, appropriate for most CLEF use cases. For one CLEF use case, however, a more fine-grained typing was required over a small number of narratives, using UMLS concept identifiers (CUIs). We therefore assigned CUIs to all entity mentions in a portion of the narratives: 35 from the stratified random sample, and 5 from a single patient of the whole patient record.

It is not easy to assign CUIs fully automatically, as a term may be ambiguous, and relate to several concepts in the UMLS. The term "cold", for example, has a CUI associating it with the temperature, and a CUI associating it with the infection. The context in which a term is mentioned is therefore required to disambiguate the possible CUIs. We therefore adopted a semi-automated approach to CUI annotation, using the GATE language processing toolkit (Cunningham et al., 2002; University of Sheffield, 2012). A custom

GATE module took each entity mention in turn from annotated gold standard documents. The mention was queried against the UMLS Knowledge Source Server API (UMLSKS API) (NLM), to fetch a list of possible CUIs for that mention, together with their UMLS semantic type, and a textual definition if available. The results were presented to a single human annotator, who examined them in the light of the mention's surrounding context. Where a single CUI had been automatically assigned, the annotator could either choose or reject that assignment. Where several CUIs were possible for a mention, the annotator could choose either one or none of the CUIs. In those cases where no suitable CUI had been automatically assigned, the annotator performed a more sophisticated manual search of the the UMLS via its web interface. The most suitable CUI found via the web interface was attached to the mention.

2.6 Analysis of the annotation process

This section presents some qualitative and quantitative results relating to the annotation process and guideline development.

2.6.1 Annotator expertise

In order to examine how easily the guidelines could be applied by other annotators with varying levels of expertise, we also gave a batch of documents to the two clinicians who assisted in guideline development (Section 2.5.3), another clinician, a biologist with some linguistics background, and a computational linguist. Each was given very limited training. The resultant annotations were compared with each other, and with a consensus set created from the two development annotators. The IAA matrices for this group are shown in Table 2.7 for entities, and Table 2.8 for relations. It is interesting to note that both the biologist and the computational linguist achieve closer agreement with the consensus set, than does the clinician. A difference analysis suggested that the computational linguist was finding more pronominal co-references and verbally signalled relations than the clinician, but that unsurprisingly, the clinician found more relations requiring domain knowledge to resolve. A combination of both linguistic and life science knowledge appears to be best: of the three non-development annotators, the biologist with some linguistics background achieved the closest agreement with the consensus set.

This difference reflects a major issue in the development of the guidelines: the extent to which annotators should apply domain-specific knowledge to their analysis. Much of clinical text can be understood, even if laboriously and simplistically, by a non-clinician armed with a medical dictionary. The basic meaning is exposed by the linguistic con-

D2	77 *(72)*				
C	67 *(60)*	68 *(62)*			
B	76 *(70)*	80 *(74)*	69 *(64)*		
L	67 *(62)*	73 *(66)*	60 *(53)*	69 *(62)*	
Consensus	**85** ***(82)***	**89** ***(86)***	**68** ***(61)***	**78** ***(72)***	**73** ***(68)***
	D1	D2	C	B	L

Table 2.7: Entity agreement by annotators by expertise, over five documents. Lenient IAA, with strict IAA in italics and parentheses, both as %. D1 and D2: development annotators; C: clinician; B: biologist with linguistics background; L: computational linguist.

D2	63 *(45)*				
C	51 *(35)*	57 *(37)*			
B	56 *(41)*	57 *(43)*	63 *(40)*		
L	57 *(36)*	62 *(42)*	49 *(27)*	51 *(33)*	
Consensus	**87** ***(74)***	**74** ***(66)***	**50** ***(34)***	**55** ***(40)***	**56** ***(36)***
	D1	D2	C	B	L

Table 2.8: Relation agreement by annotators by expertise, over five documents. Corrected IAA, with uncorrected IAA in italics and parentheses, both as %. D1 and D2: development annotators; C: clinician; B: biologist with linguistics background; L: computational linguist.

structs of the text. Some relationships between entities in the text, however, require deeper understanding. For example, the condition for which a particular drug was given may be unclear to the non-clinician. In writing the guidelines, we decided that such relationships should be annotated, although this requirement is not easy to formulate as specific rules.

2.6.2 Different text sub-genres

The guidelines were mainly developed against clinical narratives. We were interested to see if the same guidelines could be applied to imaging and histopathology reports. We found that the guidelines could be quickly adapted with minimal change, to give excellent IAA after only two iterations, as is shown in Table 2.9. Of those entities and relationships with an IAA below 75%, the majority reflect bias due to a small sample size. The fact that report IAA is better than clinical narrative IAA may reflect the greater regularity of the reports.

2.6.3 Annotation: training and consistency

In total, around 25 annotators were involved in guideline development and annotation. They included practicing clinicians, medical informaticians, and final year medical students. Each was given an initial 2.5 h of training.

After the initial training session, annotators were given two training batches to annotate, which comprised documents originally used in the debugging exercise, and for which consensus annotations had been created. IAA scores were computed between annotators,

		Narratives	Imaging	Histopath.
Iterations		5	2	2
Entities	**Condition**	91	100	92
	Intervention	82	100	n/a
	Investigation	97	75	95
	Result	100	20	80
	Drug or device	83	100	n/a
	Locus	94	97	92
	Negation signal	100	93	64
	Laterality signal	100	83	100
	Sub-location signal	100	67	50
	All	**92**	**90**	**88**
Relations	**has_target**	83	96	70
	has_finding	86	0	63
	has_indication	44	0	0
	has_location	66	90	81
	modifies (Negation)	100	100	91
	modifies (Laterality)	100	82	95
	modifies (Sub-location)	100	75	100
	corefers	52	92	67
	All	**62**	**84**	**70**

Table 2.9: Lenient IAA (entities) and corrected IAA (relations) on different document types. IAA was measured after the given number of guideline development iterations, with each iteration consisting of five documents. n/a means that there were no entities or relations for that type

and against the consensus set. The results are shown for one group of annotators, in Table 2.10 for entities, and Table 2.11 for relationships. These figures allowed us to identify and offer remedial training to under-performing annotators and to refine the guidelines further.

The matrices allow us to look at two factors. First, the IAA between annotators and the consensus set gives us a measure of consistency between annotators and our notion of truth. For entities, the trainee annotators clearly agree with the consensus as closely as the expert annotators do. For relations, they do not agree so closely. Second, the matrices allow us to examine the internal consistency between trainee annotators. Are they applying the guidelines consistently, even if not in agreement with the consensus? The wide range of relation IAA scores suggests that relationship annotation is inconsistent. Again, this may reflect the difficulty in applying highly domain-specific knowledge to relationships between entities.

2.6.4 Annotator difference analysis

During the initial guideline development process, we exhaustively examined differences between double annotators, and used the results of these analyses to both inform guideline writing, and to provide feedback to annotators. During the annotation of the final gold standard, a full analysis of all differences between the double annotations over the entire gold standard would be prohibitively time consuming, and so has not been carried

D2	77 *(73)*							
1	76 *(70)*	79 *(71)*						
2	76 *(73)*	81 *(76)*	79 *(73)*					
3	76 *(72)*	83 *(78)*	89 *(86)*	82 *(77)*				
4	75 *(70)*	84 *(79)*	83 *(78)*	81 *(80)*	85 *(82)*			
5	76 *(62)*	84 *(79)*	71 *(62)*	88 *(66)*	80 *(53)*	78 *(62)*		
6	78 *(75)*	84 *(77)*	89 *(86)*	84 *(81)*	95 *(94)*	87 *(84)*	82 *(78)*	
7	79 *(75)*	81 *(75)*	81 *(75)*	83 *(79)*	86 *(83)*	82 *(79)*	82 *(79)*	88 *(84)*
C	**85** ***(82)***	**89** ***(86)***	**84** ***(80)***	**84** ***(80)***	**88** ***(86)***	**85** ***(81)***	**83** ***(80)***	**91** ***(87)***
	D1	D2	1	2	3	4	5	6

Table 2.10: Lenient IAA (strict IAA in italics and parentheses)(%) for entities in five documents, between 7 trainee annotators, two expert development annotators (D1 and D2) and a consensus C created from D1 and D2.

D2	63 *(45)*							
1	54 *(42)*	44 *(36)*						
2	55 *(39)*	44 *(35)*	41 *(32)*					
3	65 *(48)*	59 *(48)*	60 *(53)*	49 *(39)*				
4	74 *(58)*	64 *(54)*	54 *(45)*	59 *(44)*	62 *(53)*			
5	66 *(41)*	48 *(37)*	43 *(31)*	47 *(40)*	54 *(41)*	54 *(35)*		
6	56 *(41)*	51 *(44)*	50 *(46)*	54 *(44)*	66 *(62)*	56 *(49)*	46 *(35)*	
7	69 *(52)*	54 *(43)*	52 *(43)*	52 *(41)*	59 *(52)*	61 *(48)*	64 *(50)*	57 *(50)*
C	**87** ***(74)***	**74** ***(66)***	**52** ***(46)***	**52** ***(42)***	**61** ***(54)***	**68** ***(59)***	**57** ***(44)***	**61** ***(56)***	**71** ***(61)***
	D1	D2	1	2	3	4	5	6	7

Table 2.11: Corrected IAA (uncorrected IAA in italics and parentheses)(%) for relations in five documents, between 7 trainee annotators, two expert development annotators (D1 and D2) and a consensus C created from D1 and D2.

out. Where documents showed poor agreement between the annotators, ad-hoc difference analysis was carried out to provide feedback and information for the consensus annotator. Most differences fell into a small number of categories. Some of these are described below, with examples from narratives given in Table 2.12.

1. *Occurrence:* A straightforward difference in which one annotator marked a span of text or a relation, and the other did not. Such an error could be due to a disagreement, or due to one annotator unintentionally missing something: reasons are not always clear.

2. *Textual extent:* The two annotators marked overlapping spans with the same entity type. They agreed that an annotation occurred, but disagreed on exactly what text should be marked.

3. *Typing:* The annotators agreed on annotating a specific extent of text, but assigned different entity types to that extent. Most commonly, there were confusions between `Intervention` and `Investigation`, and also between `Condition` and `Result`.

4. *Term decomposition:* One annotator marked a span as a multi-word term, with a single annotation. The other annotator decomposed the term. This was most common with `Condition` and `Locus`. For example, should "lung cancer" be marked as a single `Condition`, or a `Condition` and `Locus`? Despite rigid guidelines on how to decompose terms (based on occurrence in a standard dictionary), differences still arose.

5. *Granularity:* Usually where one annotator marked a high-level `Investigation` name and the other marked a nearby component part of that `Investigation`.

6. *Term ambiguity:* One annotator marked a span of text, but it was being used in a different sense to that implied by the annotation entity type.

7. *Locus modification:* `Locus` may be modified by both `Sub-location` and `Laterality` (e.g. "Right lobe of the lower pole of the thyroid"). This sometimes led to differences when annotating a complex anatomy expression.

8. *Multiple compounding differences:* Some examples show multiple differences that compound each other. Differences in the way in which a `Locus` and its modifiers are annotated can lead to differences in relationships, and so on.

Text	Annotator 1 response	Annotator 2 response	Type of difference
no evidence of disseminated disease	disease [condition]	disseminated disease [condition]	Textual extent
tumour markers demonstrate CA125 306	CA125 [investigation] has_result 306 [result]	tumour markers [investigation] has_result CA125 306 [result]	Textual extent; granularity
emergency admission with acute renal failure	acute renal failure [condition]	acute [condition] and failure [condition], both has_location renal	Term decomposition (Annotator 2 may have meant an acute failure has_location kidney)
I will continue to liase with the Renal team	–	renal [locus]	Occurrence; term ambiguity (Renal is an elision of "renal medicine", and not a reference to a patient's anatomical locus)
CT scan shows a partial response in the left lung lesion	CT scan [investigation] has_finding partial response [result]	1. CT scan [investigation] 2. response [condition] has_location lung [locus]	Typing; occurrence (relation). (Annotator 2 gave no [result]).
no change in the right apical mass	no [negation] modifies change [condition]	no change [negation] modifies mass [condition]	Textual extent
After discussion at the meeting today	discussion [intervention]	–	Occurrence (entity)
an infusional Morphine pump	1. infusional [intervention] 2. morphine [drug or device]	morphine pump [drug or device]	Occurrence (entity); textual extent
widespread metastatic disease to bone	1. metastatic [condition] 2. bone [locus]	metastatic disease [condition] has_location bone [locus]	Textual extent; occurrence (relation)
thoraco lumber bony tenderness	tenderness [condition] with three has_location: thoraco [locus]; lumber [locus]; bony [locus]	1. tenderness [condition] has_loation bony [locus] 2. thoraco lumber [sub-location] modifies bony [locus]	Locus modification
Blood tests were performed	tests [investigation] has_location blood [locus]	blood tests [investigation]	Term decomposition
chest: dullness to percussion in the right hemi-thorax	1. chest [locus] 2. hemi-thorax [locus] modified by left [laterality] 3. percussion [investigation] has_finding dullness [result] 4. percussion [investigation] has_target hemi-thorax [locus]	1. dullness [condition] has_location chest [locus] 2. percussion [investigation] has_finding dullness [result] 3. percussion [investigation] has_target chest [locus] 4. thorax [locus] modified by left [laterality] 5. thorax [locus] modified by hemi [sub-location]	Compounding of multiple differences in a single small example

Table 2.12: Examples of annotator difference, for narratives. In the annotator responses, annotated text is underscored, followed by an entity type in square brackets and teletype. Relation types are also in teletype, with modifiers simplified to a single `modifies` relation and its reverse, `modified by`. Text in a normal font with no underlining are comments. Where an annotator created several entities and relations, these may be numbered. A dash – means that no annotation was given by that annotator. The types of difference listed are described in Section 2.6.4.

2.6.5 Time taken to annotate

During the initial guideline development process, we timed the annotation of five narratives by a single annotator, in order to provide data for planning the main annotation process. The time to annotate these narratives had a range of 15–70 min, with a mean of 34 min. The wide range of times was not a simple function of document length: the annotators have reported that some of the shortest documents have been some of the hardest to annotate, and vice versa. Although we did not measure time to annotate documents in the main annotation exercise, the mean time of our small sample was born out by anecdote, with annotators reporting around half an hour per narrative throughout the full annotation exercise.

It should also be remembered that each document was double annotated, and followed by a consensus annotation (15 min for this last step, by anecdote). Together with the time taken to process annotations, check IAA scores and so on, each document probably took around 1.5 h to fully annotate. This excludes time taken for training, guideline and schema development, CUI annotation and time annotation.

2.7 Constructing the final corpus

Once guideline development and annotator training had been completed, annotators proceeded to double annotate the "production" corpus, consisting of the stratified random corpus and the whole patient corpus. Documents were annotated in batches of 5. On completion of a batch by two annotators, IAA was calculated for that batch. If IAA was not acceptable, then the batch was re-annotated by a further annotator. If IAA was acceptable, then the batch was put forward for consensus annotation. In the initial stages of the annotation exercise, an acceptable IAA was considered to be one that passed an arbitrary threshold of at least 65% lenient entity IAA, and at least 50% relation CIAA. As the annotation progressed, however, it became apparent that IAA could be skewed below these thresholds for one of two reasons. Firstly, there were occasional "outlier" batches with very few relations, in which a small absolute number of disagreements could lead to poor IAA. Second, a single, simple, obvious, and repeated, mistake on the part of one annotator, could also skew the IAA below the threshold. For example, one annotator completely omitted to annotate an obvious `Intervention` mentioned multiple times in one document, whereas the other annotator marked it. Given the expense of repeating annotation, it was therefore decided that low agreement on a particular double-annotation batch should not mean that the batch was rejected, if these systematic errors could be corrected in the consensus annotation stage. Consensus annotation of batches with IAA below the

threshold was therefore allowed where IAA had suffered in one of the above ways, and if the consensus annotator was confident of being able to correct the mistake.

Once consensus annotation had been completed, the consensus annotations were processed into two forms for use throughout the CLEF project, and beyond CLEF if we are able to make the corpus publicly available. First, the annotations were processed into XML files conforming to an XML schema embodying Fig. 2.2, and incorporating attributes for character offsets, text of the mentions, and CUIs where appropriate. Second, the annotations were processed into GATE datastores, for use in training and evaluation of the CLEF IE system.

The final stratified random portion of the corpus is described in Tables 2.13 (narratives), 2.14 (histopathology reports), and 2.15 (imaging reports). Each table shows distribution of entities and relations across that document type. The tables also show the IAA between the double annotators, for each entity and relation type. Note that the final gold standard consists of a consensus of the double annotation, created by a third annotator. Systems trained and evaluated with the gold standard use this consensus. The IAAs between double annotators that are given do not therefore provide an upper bound on system performance, but an indication of how hard a recognition task is.

Entity	**Number**	**Strict IAA**	**Lenient IAA**
Condition	429	81	84
Drug or device	172	84	85
Intervention	191	64	66
Investigation	220	77	82
Locus	284	78	81
Result	125	69	74
Laterality	76	95	95
Negation	55	67	76
Sub-location	49	63	64
Overall	**1601**	**77**	**80**
Relation	**Number**	**IAA**	**CIAA**
has_finding	233	48	76
has_indication	168	35	51
has_location	205	59	80
has_target	95	45	64
Modifies (Laterality)	73	70	93
Modifies (Negation)	67	63	90
Modifies (Sub-location)	43	52	98
Overall	**884**	**52**	**75**

Table 2.13: Distribution and IAA (%) of entities and relations in the 50 narrative documents in the CLEF stratified random corpus.

The results illustrate that despite training and the use of extensive guidelines, clinically trained annotators are well below perfect agreement on single annotation tasks, such as finding all of the `Investigations` in a document. The results also illustrate that relation annotation is highly dependent on entity annotation, as would be expected. CIAA, corrected for entity recognition, is significantly higher than uncorrected IAA. It is appar-

Entity	Number	Strict IAA	Lenient IAA
Condition	357	67	73
Drug or device	12	59	59
Intervention	53	57	62
Investigation	145	56	58
Locus	357	71	75
Result	96	29	33
Laterality	14	88	88
Negation	50	71	78
Sub-location	77	29	36
Overall	**1161**	**62**	**67**
Relation	**Number**	**IAA**	**CIAA**
has_finding	263	26	69
has_indication	47	15	30
has_location	270	44	70
has_target	86	20	47
Modifies (Laterality)	14	70	89
Modifies (Negation)	54	67	100
Modifies (Sub-location)	79	29	100
Overall	**813**	**36**	**72**

Table 2.14: Distribution and IAA (%) of entities and relations in the 50 histopathology reports in the CLEF stratified random corpus.

Entity	Number	Strict IAA	Lenient IAA
Condition	270	77	81
Drug or device	13	32	42
Intervention	10	43	43
Investigation	66	70	74
Locus	373	75	81
Result	71	48	52
Laterality	85	91	92
Negation	53	65	76
Sub-location	125	36	46
Overall	**1066**	**69**	**75**
Relation	**Number**	**IAA**	**CIAA**
has_finding	156	33	55
has_indication	12	14	22
has_location	268	45	77
has_target	51	67	81
Modifies (Laterality)	82	55	80
Modifies (Negation)	59	51	94
Modifies (Sub-location)	125	32	93
Overall	**753**	**43**	**76**

Table 2.15: Distribution and IAA (%) of entities and relations in the 50 imaging reports in the CLEF stratified random corpus.

ent that the overall annotation of a document is hard. Annotators are asked to look for multiple, coarsely defined entities and complex relationships between them. Documents vary in their type, from simple letters to complex reports; they vary in the style of writing; in size; and in the pathophysiology being discussed.

2.8 Temporal annotation

If the course of a patient's illness and treatment is to be modelled then the clinical entities and relationships found within text must be located in time so that they can be integrated with time-stamped information from the structured component of the patient record to construct a coherent history. To support this modelling the annotation scheme for clinical entities and relations specified above has been augmented to capture aspects of temporal information. In this section we describe the temporal annotation schema, the process of temporal annotation and the distribution of temporal annotations found in the portion of the corpus annotated so far.

2.8.1 Temporal annotation schema

Only a subset of the clinical entities identified above are 'event-like' and hence temporally situated. These are the CLEF investigations, interventions, and conditions, which we refer to in the following as TLCs (Temporally Located CLEF entities). It is interesting to note that the clinical events that we wish to temporally locate are mostly expressed in clinical text by nouns and noun phrases, which contrasts with the predominant use of verbs to express events elsewhere. We observe that most occurrences of CLEF entities in these three categories correspond to events that we would hope to temporally anchor, the exceptions being a small proportion of uses that are generic and hence not temporally situated. The exclusion of other CLEF entity types, such as drugs and results, from the TLC class is not meant to imply that time considerations do not arise for the other CLEF entity types. For example, a drug might be prescribed or discontinued at a particular time, and a result produced by an investigation that is done at a particular time. But here the temporal involvement of the drug or result is a secondary consequence of its relation to the event which is temporally locatable. Directly anchoring a drug to a date, for example, has no clear meaning without also characterising the event, i.e. was the drug prescribed or discontinued on that day? We take such considerations to be a matter of broader temporal analysis, and instead here restrict our attention to just the CLEF entity types that can be directly temporally located.

The aim of the CLEF temporal gold standard is to capture temporal relations between

TLCs and time expressions. Time expressions include dates and times (both absolute and relative), as well as durations, as specified in the TimeML TIMEX3 standard (Pustejovsky et al., 2003). Temporal relations are encoded as CTlink annotations which identify the TLCs and time expression related as well as specifying the relation type. Relation types include, for example, `before`, `after`, `overlap`, and `includes`. For a full list see Table 2.16 or Fig. 2.5. Our scheme requires annotation of only those temporal relations holding between TLCs and the date of the letter (Task A), and between TLCs and temporal expressions appearing in the same sentence (Task B). These tasks are similar to, but not identical with, those addressed by the TempEval challenge within SemEval 2007 (Verhagen et al., 2007). The scheme is graphically depicted in Fig. 2.5.

CTLink	**Task A**	**Task B**
After	5	18
Ended_by	3	0
Begun_by	4	0
Overlap	7	26
Before	5	135
None	4	8
Is_included	31	67
Unknown	6	14
Includes	13	137
Total	**78**	**405**

Table 2.16: Distribution of CTLinks by type for tasks A and B, over 10 development documents.

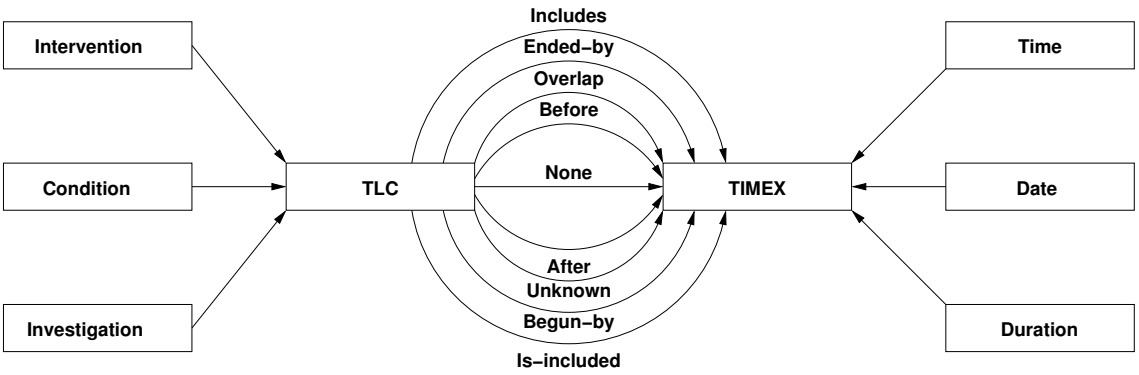

Figure 2.5: The Temporal Annotation Schema.

2.8.2 Annotation of temporal information

The temporal annotation scheme described in the previous section, which is still under development, has to date been used to annotate 10 patient letters (narrative data) from the clinically-annotated corpus described above in Section 2.4. In time we intend to annotate all of the gold standard corpus.

Temporal annotation is done through a combination of manual and automatic methods. TLCs can be immediately identified from the clinical entity annotations already present in the letters. Temporal expressions are annotated and normalised to ISO dates by the GUTime tagger (Mani and Wilson, 2000), which annotates in accordance with the TIMEX3 standard. This annotation is manually checked and corrected as necessary. After these automatic steps, we manually annotate the temporal relations holding between TLCs and the date of the letter (Task A), and between TLCs and temporal expressions appearing in the same sentence (Task B).

2.8.3 Distribution of temporal annotations

The distribution of annotations for the different subtypes of CTLinks, TLCs and time expressions for the ten development documents annotated so far are shown in Tables 2.16 and 2.17. Note that some TLCs are marked as hypothetical. For example in *no palliative chemotherapy or radiotherapy would be appropriate* the terms *chemotherapy* and *radiotherapy* are marked as TLCs but clearly have no 'occurrence' that can be located in time and hence will not participate in any CTLinks.

TLCs	Not hypothetical	243
	Hypothetical	16
	Total	**259**
Time Expression	Duration	3
	Date	52
	Total	**55**

Table 2.17: Distribution of TLCs and temporal expressions, over 10 development documents.

2.9 Using the corpus: the CLEF IE system

The CLEF corpus has been created to enable the training and evaluation of the CLEF IE system, which can be applied to previously unseen clinical texts, to automatically extract the entities, modifiers, and relationships that the annotation schema describes. This system has been built using the GATE NLP toolkit (Cunningham et al., 2002; University of Sheffield, 2012), which allows language processing applications to be constructed as a pipeline of processing components. Documents are passed down the pipeline being analysed by each component in turn, with the results of this analysis being available to later components. The CLEF IE pipeline is outlined in Fig. 2.6, with separate pipelines being shown for training and application of the system (although the two pipelines substantially overlap). In either case, the pipeline has three main parts:

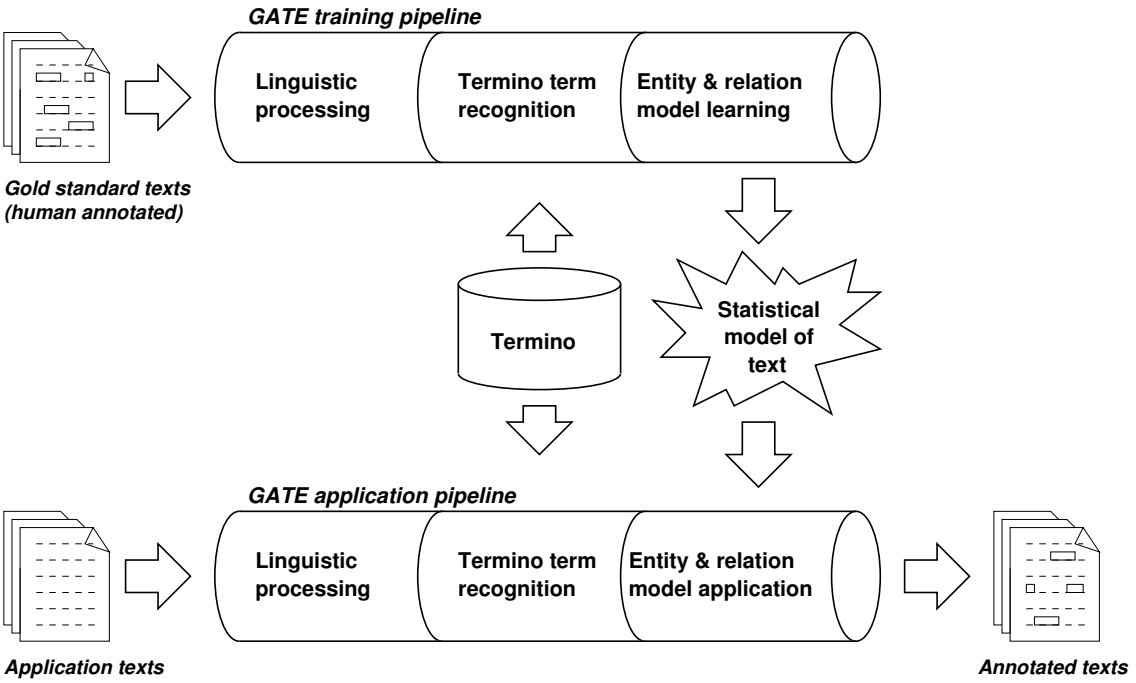

Figure 2.6: The CLEF Information Extraction system.

Linguistic preprocessing: First, the text of each document is split into tokens (such as words, numbers and punctuation) and sentences, and then part of speech (POS) information is added.

Dictionary-based term look-up: Next, medically significant terms are identified, using a dictionary-based look-up approach. This is done using Termino: a large-scale terminological resource designed specifically for text processing (Harkema et al., 2004a). Termino consists of two parts. The first is a database constructed from existing terminology resources. Termino provides uniform access to these resources, and links from recognised terms back to resource entries. The second part consists of finite state recognisers compiled from terms in the database. Our principle terminology source in CLEF is the Unified Medical Language System (UMLS) (Lindberg et al., 1993), which is the largest source of medical vocabulary, and which links terms to other information, such as semantic types.

Statistical recognition of entities and relations: we treat the recognition of both entities and relations as classification tasks, using Support Vector Machines (SVMs) as trainable classifiers, as they have proven to be effective for a range of NLP tasks. We use an SVM implementation provided as part of the GATE toolkit. We will discuss the recognition of entities and relations separately in turn.

2.9.1 CLEF entity recognition

SVMs are binary classifiers, and so separate classifiers must be trained to recognise the different entity types. Furthermore, our classifiers apply to individual tokens, and so multi-token entities are recognised using a BE (Begin/End) style of boundary learning. This is handled by the GATE Learning API (Li et al., 2005). A pair of binary classifiers are trained for each entity type: one for the begin (B) token and one for the end (E) token. For our five entity types, 10 binary classifiers are therefore built, and each is applied independently of the others. A post-processing step is required to combine pairs of B and E tokens, to find the boundaries of candidate entities, and to adjudicate between conflicting (i.e. overlapping) candidates.

The features used to classify each token are based on the token itself, and the token on either side of it. Features include the morphological root and affix (for words), a generalisation of the POS, token type (e.g. word, number) and orthographic type (e.g. upper/lower case). So that dictionary look up can contribute to entity recognition, a further feature indicates whether the token is part of term recognised by Termino, taking the term's type as it value if it is, and the value `null` otherwise.

The recognition performance of this system is shown by the results in Table 2.18, which were computed over the 77 clinical narrative documents of the CLEF corpus, using ten-fold cross-validation. Scores are provided for the standard metrics of Precision (P), Recall (R) and F-measure ($F1$), with scores macro-averaged across the ten folds. As an indicator of the difficulty of each entity recognition task, the table also provides Inter Annotator Agreement (IAA) scores for the two independent annotators (but note that the system is trained on a third *consensus* annotation). Observe that the overall $F1$ performance of this system falls only 3% behind that of the overall averaged IAA.

Entity type	**Metric**			**IAA**
	P	R	F1	
Condition	0.819	0.654	0.724	0.751
Drug-or-device	0.83	0.592	0.684	0.781
Intervention	0.75	0.616	0.665	0.554
Investigation	0.831	0.659	0.73	0.745
Locus	0.8	0.616	0.694	0.793
Overall	0.807	0.631	0.707	0.737

Table 2.18: Entity recognition scores for the CLEF IE System.

The use of Termino dictionary lookup as a feature in a supervised statistical entity recognition system is an attempt to address two major challenges in entity recognition. First, pure dictionary lookup can give poor precision, due to term ambiguity with general language ("I", for example, is both a pronoun and an abbreviation for Iodine). Second, supervised statistical techniques are restricted to a model based only on those entities found in the training data. Although we have not performed a proper error analysis of

our results, inspection reveals that both types of errors still occur, even if at a reduced rate. In addition, we cannot rule out errors due to, e.g. incorrect POS tagging and morphological analysis. A more detailed account of our entity recognition approach has been published (Roberts et al., 2008c).

2.9.2 CLEF relation recognition

Relation extraction is treated as a classification task by taking a set of entity pairs that *might* be related and requiring the system to assign to each one of the relationship types, or the type `null` to indicate that no relation holds. The set of candidate pairs to be considered is restricted first by allowing only pairs whose types can be linked by some relation (e.g. no CLEF relation can link `Drug-or-device` and `Result` entities, so no such pairs are created), and second by only pairing entities that are no more than n sentences apart (we here allow only pairs for entities in the same or adjacent sentences). For classifier training, this set of candidate pairs is computed, and those for which a relation is asserted in the gold standard are assigned that relation type as class, and all others the class `null`. These pairs constitute the instances for which the classifier model is built. In classifier application, the corresponding set of entity pairs are computed for an unseen text (after entity extraction has been done) and the model applied to determine which pairs are related and how. As with entity recognition, we use an SVM implementation available in GATE, and use the GATE Learning API to handle the task of recasting this multi-class classification task as a combination of binary classifiers, with a post-processing step to reconcile conflicts.

We have explored using a range of different features sets with these classifiers, including features such as the surface string, morphological root and POS of the tokens of the two entities and of the n tokens appearing to either side of the entities. Other features include the types of the two entities, their linear order (i.e. which appears first), and the distance between them (measured as number of sentence boundaries). This feature exploration and the resultant optimally performing feature set are fully described in Roberts et al. (2008d). We used the optimally performing feature set with the system to produce the relation extraction results shown in Table 2.19, which were again computed over the 77 clinical narrative documents of the CLEF corpus, using ten-fold cross-validation, with macro-averaging of scores across the ten folds. Note that the entities provided as input to relation extraction are those of the gold standard corpus, rather than the result of automatic entity recognition, so that we can see the performance of relation extraction in isolation from the damaging effects of errorful input. To give an indication of the difficulty of relation extraction, the table includes scores for agreement between the two

independent annotators analysing texts, but these are *corrected* IAA, i.e. they compare only the relationships for which both of the related entities have been found by *both* annotators. Observe that the overall system F1 is 70%, compared to a CIAA of 75%. A more detailed account of our relation extraction approach has been published (Roberts et al., 2008a).

Relation	**Metric**			**CIAA**
	P	R	F1	
has_finding	0.63	0.82	0.71	0.80
has_indication	0.44	0.47	0.41	0.50
has_location	0.73	0.83	0.76	0.80
has_target	0.59	0.68	0.62	0.63
laterality_modifies	0.86	0.89	0.85	0.94
negation_modifies	0.81	0.93	0.85	0.93
sub_location_modifies	0.87	0.95	0.90	0.96
Overall	0.64	0.76	0.70	0.75

Table 2.19: Relation extraction scores for the CLEF IE System.

2.10 Discussion and conclusions

We have described the CLEF corpus: a semantically annotated corpus designed to support the training and evaluation of information extraction systems developed to extract information of clinical significance from free text clinic notes, imaging reports, and histopathology reports. We have described the design of the annotated corpus, including the number of texts it contains, the principles by which they were selected from a large body of unannotated texts and the annotation schema according to which clinical and temporal entities and relations of significance have been annotated in the texts. We also described the annotation process that was undertaken with a view to ensuring, as far as is possible given constraints of time and money, the quality and consistency of the annotation, and we have reported results of inter-annotator agreement, which show that promising levels of inter-annotator agreement can be achieved. We have examined the applicability of annotation guidelines to several clinical text types, and our results suggest that guidelines developed for one type may be fruitfully applied to others. We have also reported the distribution of entity and relation types, both clinical and temporal, across the corpus, giving a sense of how well represented each entity and relation type is in the corpus.

We believe the CLEF corpus makes a significant contribution to research on clinical language processing both in terms of the resource produced and the methodology adopted to develop this resource. Nonetheless there are limitations both to the resulting resource and to the methodology.

Regarding the resulting resource, we must consider the size of the resource, and the quality of annotation. The size of the corpus is a straightforward function of the available

annotator time. Quality of annotation will reflect both the consistency and completeness of the guidelines, and the correct application of those guidelines by annotators. The former could be improved by investing more time in iterative development and debugging of the guidelines. The latter could be improved by additional annotation steps. As with any annotated corpus, annotation quality will to some extent reflect the overriding expense of annotator time. Anything that reduces the burden on annotators, may be expected to improve both quality and the size of the final corpus. Techniques that might reduce this burden are discussed below.

Regarding the corpus development methodology, the most obvious limitation is that such efforts require a lot of annotator labour and that annotators find the work hard. Since the annotation requires specialist medical knowledge the pool of possible annotators is relatively small. Furthermore we found the recruitment, training and co-ordination of annotators at different sites working on sensitive data to be logistically complex, also requiring significant effort. Because the work was difficult a number of annotators resigned after a limited contribution forcing us into an iterative cycle of recruitment and training.

Various steps could be taken to address these difficulties in future annotation exercises. To attempt to utilise annotator effort most effectively, so-called active learning or mixed initiative approaches could be explored (Thompson et al., 1999; Ghani et al., 2003). In these approaches annotation and system learning stages are interleaved so that at any point an annotator is correcting and augmenting annotations that the system has added to a document rather than annotating a document from scratch. As the system learns, the amount of human annotator input per annotated document should go down and human effort should be concentrated on difficult cases, i.e. ones the system has missed or annotated incorrectly. Thus more annotated text should result from equivalent annotator effort when using active learning as compared with not using it.

To address the difficulty of the task, one approach is simply to reduce the scope of the annotation scheme and to focus on fewer entities or relations. This may or may not be possible depending on the intended application. Another approach, and one which could also help with the logistical difficulties, is to move to a distributed, collaborative annotation framework in which the grain size of annotation instances is reduced to a snippet, e.g. a single sentence. A number of such collaborative annotation tools are emerging–see, e.g. Cunningham (2008); BioNotate (2008). Such an approach has numerous advantages: the annotation effort can be distributed globally, drawing on interested parties anywhere; smaller annotation grain size reduces the unit of useful annotation meaning smaller levels of effort can be exploited, reduce the difficulty for annotators by focusing effort on single-decision types over small snippets of text; annotation of individual instances can be repeated until a satisfactory level of agreement is reached, or the instance is eliminated

as problematic; rogue or poor quality annotators can be identified and their annotations removed. There are, however, non-trivial obstacles to using such a methodology in our domain, including the need to protect patient confidentiality, and the fact that some of the inter-sentential relations annotated in our corpus would be excluded if only snippets of text were presented to annotators.

These considerations all point to ways in which the difficulties we have encountered in our annotation effort could be mitigated in future annotation projects. Nonetheless, despite these difficulties, the annotated CLEF corpus is the richest resource of semantically marked up clinical text yet created, one which we hope will be of wide-ranging interest and utility to the clinical language processing research community.

Availability

The current availability of all of the resources in this paper is described on the project web site (University of Sheffield, 2008), together with links to each available resource. Most of the software, including the ANNALIST scoring tool, is available for download, as is the final version of the guidelines.

At the time of publication, there is some limited availability of the CLEF gold standard. We are able to share small samples of data from the gold standard, which may include short extracts of documents. In order to ensure anonymity, such releases go through a triple manual inspection, by an ethicist, a clinician, and a confidentiality expert. Full release of the whole gold standard will be made on the project web site (University of Sheffield, 2008), after approval by a UK Multi-centre Research Ethics Committee.

Acknowledgements

This research was supported by UK Medical Research Council Grant No. RB106367, "CLEF Services". We would like to thank the Royal Marsden Hospital for providing the corpus; our annotators at the University of Manchester and University College London; and members of CLEF Services who have helped with clinical expertise and logistics, particularly Jay Kola, Bill Wheeldin, James Cunningham, and Colin Puleston (all at the University of Manchester); and Dipak Kalra, Archana Tapuria, and Nathan Lea (all at University College London).

Chapter 3

Combining terminology resources and statistical methods for entity recognition: an evaluation

Foreword

The following Chapter is reproduced in full from Roberts et al. (2008c):

> A. Roberts, R. Gaizauskas, M. Hepple, and Y. Guo. Combining terminology resources and statistical methods for entity recognition: an evaluation. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation, LREC 2008*, Marrakech, Morocco, May 2008c

Author's contribution

The author of this thesis wrote the first complete draft of Roberts et al. (2008c), and led the writing of all subsequent drafts. The author contributed all of the work described in the paper:

- Prepared the corpora used (Section 3.3);
- Prepared and filtered the terminologies used, including contributing to the design of the Termino dictionary based term recognition system, which is reported in a separate paper (Harkema et al., 2004b) (Section 3.4.1);
- Developed the feature sets used in the statistical entity recognition system (Section 3.4.2);

- Designed the experiment and carried out all evaluations (Section 3.5);
- Analysed the results (Section 3.6).

Copyright and permission to use

3.1 Abstract

Terminologies and other knowledge resources are widely used to aid entity recognition in specialist domain texts. As well as providing lexicons of specialist terms, linkage from the text back to a resource can make additional knowledge available to applications. Use of such resources is especially pertinent in the biomedical domain, where large numbers of these resources are available, and where they are widely used in informatics applications. Terminology resources can be most readily used by simple lexical lookup of terms in the text. A major drawback with such lexical lookup, however, is poor precision caused by ambiguity between domain terms and general language words. We combine lexical lookup with simple filtering of ambiguous terms, to improve precision. We compare this lexical lookup with a statistical method of entity recognition, and to a method which combines the two approaches. We show that the combined method boosts precision with little loss of recall, and that linkage from recognised entities back to the domain knowledge resources can be maintained.

3.2 Introduction

Specialist domains are characterised by extensive use of technical and domain specific terminology. Term recognition is an important step towards Named Entity Recognition (NER) in these domains: entities, or things in the real world, are often referred to by terms in the text. Large scale knowledge resources such as terminologies and ontologies are typically available in these same domains. We might expect such resources to have some use in term and entity recognition. We might also expect entity recognition to add value by linking entities back to these knowledge resources, making additional information available to applications and their users.

Although large scale resources offer big advantages, they also have a major disadvantage: most have not been designed with natural language processing in mind. They may suffer from low coverage in some area of importance to an application, and from problems of ambiguity in other areas. Through combining dictionary lookup with statistical models, we hope to overcome these disadvantages, while retaining the advantages of linking to the underlying resources. Can, in practice, use of these large scale resources be shown to benefit entity recognition? This is our research question. Our question parallels a long-standing question of gazetteer use for NER in Information Extraction: are large gazetteers useful for NER (Stevenson and Gaizauskas, 2000), or can statistical models of context alone provide sufficient performance (Mikheev et al., 1999)? We examine this question with respect to biomedicine. Specifically, we look at clinical documents. This domain is characterised by complex terminologies, and by a wealth of large terminology resources. Our question is, however, pertinent to any technical domain.

Dictionary lookup in the biomedical domain is especially prone to problems of ambiguity. This has been noted for gene names (Proux et al., 1998; Hirschman et al., 2002), but is also true for clinical text. Large numbers of abbreviations are used, and these are often ambiguous with short words in general language. For example, many one and two character words are abbreviations for chemical elements, after which medical investigations are named. “I” is an abbreviation for Iodine, and used to mean an Iodine test, but of course most commonly appears as the personal pronoun. Some dictionary lookup methods, our own included, match morphological roots of tokens, rather than token strings. For example, the verb “be” (and therefore its derivatives if matching morphological roots) is ambiguous with “BE”, an abbreviation for Bacterial Endocarditis.

Hirschman et al. (2002), looking at gene names, demonstrated the scale of this problem with a simple baseline experiment. Using a standard resource, they extracted gene names from research paper abstracts, with a precision of 7% and a recall of 31%. By eliminating potential names of three or less characters, precision rose to 29%, while recall only dropped to 26%. Several solutions to this problem have been investigated. From the examples above, it would seem sensible to use additional information, such as part of speech, to disambiguate dictionary matches. Proux et al. (1998) used such an approach to recognise gene names ambiguous with general language words: a potential gene name was eliminated from consideration if it had a non-noun part-of-speech. Other solutions have shown that syntactic information is not always necessary, instead using the domain specificity of potential terms. For example, Stevenson and Gaizauskas (2000) looked at entity recognition in newswire, showing that large gazetteers can improve recall, but that they may also introduce ambiguity. They used two methods to overcome this. First, they removed those words from the gazetteer that also occur in a standard dictionary. Sec-

ond, they removed those words that occurred more frequently in their training corpus as non-terms than terms. Both of these methods showed improved results.

Dictionary lookup can be contrasted to machine learning approaches. Such techniques are widespread in the biomedical domain, especially for term and entity recognition of proteins and genes (see Ananiadou and Nenadic (2006) and Park and Kim (2006) for reviews). Several applications have used a "pure" machine learning approach, in which no external dictionaries are used. Tanabe and Wilbur (2002), for example, used transformation based learning to build ABGENE, a gene and protein name recogniser. ABGENE includes a Brill POS tagger trained on a corpus that has been hand-labelled with gene and protein names. Others have combined dictionary lookup and machine learning of statistical models. Mika and Rost (2004) trained several Support Vector Machines (SVMs) on lexical features. A further SVM was trained on the outputs of these, combined with a dictionary lookup. Use of the dictionary increased performance significantly. Yamamoto et al. (2003) used an SVM to find protein names in text. Features included several that encoded whether a term appeared in a dictionary, which was built from a biomedical corpus and protein knowledge bases. These lookup features proved crucial.

We examine entity recognition of medically important entities in texts from patient records. Although statistical and machine learning techniques have been used in this domain (see Pakhomov et al. (2005) for example), they are not as widely used as for protein and gene recognition. In clinical text, dictionary lookup combined with syntactic parsing is much more common. Our experiments use a system which contains a dictionary based lookup of terms from large scale terminologies, filtering of ambiguity from this dictionary lookup, and supervised learning of statistical entity recognition models. As with protein and gene recognition, these approaches are not mutually exclusive: a dictionary based term lookup can be used to provide features for statistical models. We therefore examine these components independently, and in combination. We also look at whether a combined method can retain a major advantage of dictionary lookup, linkage from recognised entities back to domain resources.

3.3 Corpus

A major difficulty when evaluating natural language processing (NLP) over clinical texts, is the almost complete absence of gold standards for the domain. This is largely due to issues of data confidentiality. The CLEF project (Rector et al., 2003) has been fortunate in obtaining a large corpus of over 500K documents from over 20K patients. We have used a small subset of these documents to build a gold standard of manually annotated entities and relations. The gold standard has been carefully constructed using best practice

Entity type	Brief description	Number of instances
Condition	Symptom, diagnosis, complication, conditions, problems, injuries etc.	739
Drug or device	Usually a drug, but can be other prescribed items such as medical devices	272
Intervention	Action performed by a clinician, targeted at a patient, locus, or condition	298
Investigation	Tests, measurements, and studies	325
Locus	Anatomical structure or location, body substance etc.	490
Total		**2124**

Table 3.1: Entity types and numbers of instances in a gold standard corpus of 77 narratives.

methods, as described fully in (Roberts et al., 2007). Documents were annotated by two independent, clinically trained, annotators, and a consensus annotation created by a third.

For the experiments reported in this paper, we used 77 gold standard documents of a single type, clinical narratives (generally letters from one clinician to another that describe a patient's progress). We used consensus annotations of five entity types on these narratives. By *entity*, we mean some real-world thing, event or state referred to in the text. The entity types are shown in Table 3.1, together with the total number of instances of each type in all 77 documents. In addition to the annotated gold standard, we have also built an unannotated development corpus of similar documents. This was used whenever inspection of documents was required as part of system development. The annotated corpus was never inspected in development.

3.4 Algorithms and resources

The corpus is pre-processed by tokenisation, sentence splitting, and part of speech tagging, using the GATE text mining toolkit (Cunningham et al., 2002). Our entity recognition components are also implemented in GATE.

3.4.1 Dictionary based term recognition

For dictionary based lookup, we use Termino: a large-scale terminological resource designed specifically for text processing (Harkema et al., 2004a). Termino consists of two parts. The first is a database constructed from existing terminology resources. Termino provides uniform access to these resources, and links from recognised terms to resource entries. The second part consists of finite state recognisers (FSRs) compiled from the database. Terms found by a FSR are associated with a unique ID linking back to the external resource, and with a semantic type derived from the external resource.

Our principle terminology resource in CLEF is the Unified Medical Language System (UMLS) (Lindberg et al., 1993) [1]. UMLS is the largest source of medical vocabulary, being a superset of other resources, and provides links from terms to other information, such as semantic types.

We import UMLS terms into Termino. A significant number of the terms in UMLS are of little value for medical NLP tasks. For example, they represent non-medical concepts, are case variants of other terms, or are complex knowledge engineering class names that are unlikely to be found in text. These terms degrade the performance of NLP applications based on UMLS (Aronson, 2005). We filter out such terms, prior to importing into Termino. For example, we reject long terms (> 5 words) and terms containing certain constructs that mark them as class names. The full set of rejection criteria is derived from McCray et al. (2001), McCray et al. (2002), and Aronson (2005).

3.4.1.1 Filter and Supplementary Term Lists

Despite this rejection of many UMLS terms that are not suitable for NLP, described above, we still found that Termino falsely matched common general language words. To identify these, we ran Termino over our development corpus, and manually inspected the results. From all matches, we created a list of spurious terms in the development corpus, as follows:

1. Add all unique terms of length = 1 to the list.
2. For all unique terms of length ≤ 6, manually inspect, and for each:
    - add to the list if it matches a common general language word, a common abbreviation (e.g. *pm*, or *Mr*), or an SI unit;
    - add to the list if it has a numeric component;
    - reject from the list if an obvious technical term;
    - reject from the list if none of the above apply.

This gave a list of 232 terms, which we call the *filter list*. This list was added to Termino, as a list of terms to ignore. The list counters the tendencies of dictionary lookup methods to over-recognise. In use, it performs a similar function to the methods of Hirschman et al. (2002) and Stevenson and Gaizauskas (2000) discussed in the Introduction. Filtering uses no syntactic information, and instead relies on simple heuristics (such as term length), and on knowledge of the domain specificity of terms.

[1] We use the UMLS Metathesaurus 2007AB release, taking terms from license category 3 source vocabularies.

A second list was created at the same time, of terms that were not recognised by Termino, and of special significance according to domain experts. This list consists of 6 terms, mainly of type `Intervention`. This list, called the *supplementary list*, was added to Termino as a list of additional terms to recognise. Neither of these lists took more than a few hours to construct. Their benefits will be demonstrated in the results section.

3.4.2 Statistical entity recognition

There are many algorithms suitable for statistical entity recognition. We build supervised statistical entity recognition models using SVMs, which have the advantage of good performance over the sparse training data commonly found in text applications. By using SVMs, we are comparing our dictionary based lookup with an approach used in many popular and state of the art systems. We use a variant SVM algorithm, SVM with uneven margins, as provided with the GATE text mining toolkit's Learning API (Li et al., 2005). Kernel parameters were set to those that gave the best results in initial experiments with a pilot corpus, prior to the construction of the corpus used for the experiments reported here.[2] All other GATE Learning API parameters were left at their defaults.

SVMs are binary classifiers, and so different classifiers must be trained to recognise the different entity types. Furthermore, our classifiers apply to individual tokens, and so multi-token entities are recognised using a BE (Begin/End) style of boundary learning. This is handled by the GATE Learning API. A pair of binary classifiers are trained for each entity type: one for the begin (B) token, and one for the end (E) token. For our five entity types, ten binary classifiers are therefore built. Each is applied independently of the others.

For each entity type, post-processing combines pairs of B and E tokens to find the boundaries of candidate entities, according to these rules:

1. each token classified as a B is paired with all following tokens classified as E;

2. a token that is classified as both a B and an E by a pair of classifiers will be considered a candidate single token entity;

3. for overlapping candidates:

   (a) remove those candidates that do not have the same length as any training entity of the same type;

[2] Specifically, we used a polynomial kernel with degree 3, cost parameter c of 0.7, and the uneven margins parameter τ set to 0.6.

(b) select the remaining candidate with the maximum confidence, where candidate confidence is the product of confidences calculated from the outputs of the B and E classifiers.

We use a very simple set of token features for our models. Features are constructed for a window of one token on either side of the token being classified.

The features and window size used have been derived by trial of various combinations, and are those that gave the best results (some of this experimentation is currently under review for publication). It is possible that better results can be achieved by extending and tailoring the feature set, but those used give reasonable performance, and are an easily implemented basis for the experiments reported. Our purpose is a comparison of statistical and non-statistical methods, not optimisation of SVMs. The following token features are used:

- Morphological root
- Affix
- Generalised part of speech (POS) category
- Orthographic type (e.g. lower case, upper case)
- Token kind (e.g. number, word)

Most of these features are provided by the standard tokeniser and POS tagger components of the GATE toolkit. The exception is generalised POS category, which is the first two characters of the full POS tag. This takes advantage of the Penn Treebank tagset used by GATE's POS tagger, in which related POS tags share the first two characters. For example, all six verb POS tags start with the letters "VB".

To combine dictionary lookup with statistical entity recognition, we augment token features with a term type feature. If a token is part of a term recognised by Termino, this feature takes the term's type as its value. Otherwise, it is given a value of `null`. The final recognition decision is made by an SVM, using this feature amongst others. Again, we use a window of one token on each side of a candidate token.

3.5 Evaluation

Evaluation metrics are defined in terms of true positive, false positive and false negative matches between entities in a system annotated *response* document and a gold standard *key* document. A response entity is a true positive if an entity of the same type, and

with the exact same text span, exists in the key. Matching of response entities to key entities is therefore strict (i.e. overlapping key and response entities do not contribute to scoring). Corresponding definitions apply for false positive and false negative. Counts of these matches are used to calculate standard metrics of Recall (R), Precision (P) and $F1$ measure.

As Termino does not need any gold standard training data, evaluation of Termino is by a direct comparison of the gold standard entities to the terms matched by Termino, assuming that each term matched corresponds to an entity. For Termino, we report metrics for entity types macro-averaged across all documents. As our statistical entity recognition is supervised, we need the gold standard for training data. We have therefore trained and evaluated using ten fold cross-validation, with metrics macro-averaged over all ten folds.

The metrics do not say how hard entity recognition is: there is nothing against which to compare the system. We therefore provide Inter Annotator Agreement (IAA) scores from the gold standard. The IAA score gives the agreement between the two independent double annotators. It is equivalent to scoring one annotator against the other using the $F1$ metric (Hripcsak and Rothschild, 2005). Note that the measure compares two human annotators. As the system is trained on a third *consensus* annotation, the IAA does not give an upper bound on performance. It is possible for the system to score a higher $F1$ than the IAA for the same entity type.

Entity type	Metric	Termino			SVM + tokens	SVM + tokens + best Termino	IAA
		UMLS	UMLS + filter	UMLS + filter + supplementary			
Condition	**P**	0.1971	0.4656	0.4656	0.7994	0.8186	
	R	0.7224	0.7171	0.7171	0.5670	0.6540	
	F1	0.3097	0.5646	0.5646	0.6604	0.7242	0.7504
Drug or device	**P**	0.2680	0.6224	0.6224	0.7333	0.8301	
	R	0.7308	0.7205	0.7205	0.4433	0.5920	
	F1	0.3922	0.6679	0.6679	0.5456	0.6840	0.7808
Intervention	**P**	0.2921	0.5158	0.5272	0.8102	0.7500	
	R	0.5582	0.5582	0.6301	0.5753	0.6157	
	F1	0.3835	0.5362	0.5741	0.6504	0.6649	0.5535
Investigation	**P**	0.1841	0.5438	0.5438	0.8349	0.8308	
	R	0.6941	0.6763	0.6763	0.5608	0.6592	
	F1	0.2910	0.6029	0.6029	0.6671	0.7300	0.7448
Locus	**P**	0.4453	0.5654	0.5654	0.8057	0.8004	
	R	0.7409	0.7409	0.7409	0.5298	0.6158	
	F1	0.5563	0.6413	0.6413	0.6347	0.6940	0.7925
Overall	**P**	0.2458	0.5224	0.5238	0.7931	0.8065	
	R	0.6999	0.6939	0.7042	0.5417	0.6308	
	F1	0.3638	0.5961	0.6008	0.6423	0.7071	0.7373

Table 3.2: Entities found by Termino using UMLS and other term lists; entities found by SVM trained with token features; and entities found by SVM trained with token features plus features from the best Termino configuration. All scored on corpus C77, and shown with inter-annotator agreement for the same corpus.

3.6 Results

3.6.1 Dictionary Lookup

The first set of experiments looked at various configurations of Termino, with and without filter terms and supplementary terms. These show the performance of simple dictionary lookup based on UMLS, and of dictionary lookup tailored with additional cheaply constructed lists. The results of these experiments are reported on the left of Table 3.2. The table shows that Termino loaded with just UMLS gave a recall of > 0.69 for all entity types except `Intervention` at 0.55. Precision, however, was low at between 0.18 and 0.47. Overall precision was 0.25. Error analysis with our development corpus showed that the low precision was due to the large amount of ambiguity inherent in such large scale resources, as discussed above. The second column shows the effect of using a filter term list to disallow these common spurious matches. Precision more than doubles in most cases, to 0.52 overall. Recall drops by less than 2% in all cases, not changing at all in some. The filter list clearly makes a big difference to performance. The terms that it removes are almost always spurious, and rarely genuine.

We also added a small list of terms considered important by domain experts in the CLEF project, but not included in UMLS. The results for Termino with this list included are show in the third column. The supplementary list only has an effect on `Intervention`, where recall increases by > 7%. This is clearly significant in the case of a specific entity type, but has little overall impact (< 0.5% increase in overall $F1$)

With both lists added, Termino achieves an $F1$ around 10% to 20% below IAA for most entity types. The exception to this is `Intervention`, which is > 1.5% above the IAA. `Intervention` has the lowest IAA, 0.55, indicating that is difficult for human annotators to reach agreement on this entity type. This difficulty is reflected by the fact that a dictionary lookup performs just as well.

3.6.2 Statistical Models

The second set of experiments looked at SVM entity learning. The first of these experiments used simple token features. The second experiment combined simple token features with a Termino feature, as described above, in Section 3.4.2 The results of these experiments are also reported in Table 3.2.

The **SVM + tokens** column in Table 3.2 shows the performance of a system trained with no terminological knowledge. The only features used were those that described the surface form of the token (e.g. string and orthography), and its POS. For each entity type, recall is below that of the best Termino system, with differences in the range 5% to

28%. These results show that Termino contains a reasonable proportion of the entity terms appearing in the gold standard, and this will presumably also be true for the remainder of the corpus. The SVM, on the other hand, is limited to build a model only based on terms annotated in the gold standard. Turning to precision, we find that it is 10% to 30% above that of the best Termino system. Despite being limited in its scope, the model that the SVM does build is accurate, avoiding the ambiguity from which dictionary lookup with Termino suffers. The increase in precision is not mirrored exactly by a drop in recall: $F1$ does not stay the same for all entities. While for most, $F1$ is higher with the SVM, for `Locus` it is slightly lower, and for `Drug or device` it is 12% lower, showing that higher precision is at the expense of a much bigger drop in recall than for other entity types. Dictionary lookup appears to be especially useful in this case.

The **SVM + tokens + best Termino** column of Table 3.2 shows the SVM with term features added to the previous token features. A feature is added that records whether a dictionary lookup term coincides with a token. The features used were from the best Termino, using UMLS, filter terms, and supplementary terms. The most consistent trend over SVM with token features only, is an increase in recall, of between 4% and 15%. The additional terminological information has presumably enabled the SVM to build a more general model that is able to exploit the broader knowledge that Termino contributes. While precision also improves, overall (> 1%), the improvement is not so clear cut. For two entity types (`Locus` and `Investigation`, it drops very slightly. For `Intervention`, it drops by 6%. While generally good, the SVM has not always been able to overcome the ambiguity inherent in dictionary lookup. In terms of $F1$, SVM with token and Termino features consistently outperforms SVM with token features only, by around 6% overall.

Across all systems, SVM with token and Termino features performs best, with $F1$ 3% to 10% below IAA (and in one case, `Intervention`, 11% above). The combined systems manages to gain from the higher recall of dictionary lookup, while not suffering from a loss in the precision of the statistical method.

3.6.3 Linkage of Entities to External Resources

An advantage of dictionary-based term recognition over statistical methods is that a dictionary-based system such as Termino can provide entry points into the source terminologies and ontologies. These entry points make the information from the external resources available for further text processing steps, for querying, and for other applications. Can this advantage be carried through to the combined dictionary-lookup and statistical method?

In Termino, entry points to source terminologies and ontologies are implemented by

Entity type		CUIs assigned						
		0	**1**	**2**	**3**	**4**	**5**	**> 0**
Condition	**Number**	40	180	54	14	1	2	251
	%	13.75	61.86	18.56	4.81	0.34	0.69	86.25
Drug or device	**Number**	10	101	8	1	0	0	110
	%	8.33	84.17	6.67	0.83	0	0	91.67
Intervention	**Number**	47	21	55	0	0	0	76
	%	38.21	17.07	44.72	0	0	0	61.79
Investigation	**Number**	20	68	36	5	0	0	109
	%	15.50	52.71	27.91	3.88	0	0	84.50
Locus	**Number**	29	116	37	11	5	1	170
	%	14.57	58.29	18.59	5.53	2.51	0.50	85.43
Total	**Number**	146	486	190	31	6	3	716
	%	16.94	56.38	22.04	3.60	0.70	0.35	83.06

Table 3.3: Numbers of external resource identifiers (UMLS CUIs) assigned to terms found in a development corpus of 50 documents, by a combined SVM and Termino system.

annotating each term with unique identifiers for entries in those resources. In the case of the UMLS, this is a *Concept Unique Identifier*, or CUI. In the Termino-only system, every term found will have at least one CUI. Some terms will be ambiguous in UMLS, and may have more than one CUI. For example, the term *chemotherapy* is ambiguous between a type of drug therapy, and a course of treatment (*chemotherapy regimen* elided).

In the combined system, there will be an overlap between entities found by the SVM and those found using Termino terms alone. Some terms will have been found by Termino but rejected as entities by the SVM, some terms found by Termino and confirmed as entities by the SVM, and other entities will have been found by the SVM alone. As the SVM is the final arbiter in the combined system, these last two groups make up those entities ultimately recognised. We assign CUIs to entities from the combined system where Termino has also found a term at the same point in the text. The Termino term must also have the same type as the entity recognised by the SVM: for example, there is no point in assigning a CUI for a `Locus` term found by Termino, to a `Condition` entity found by the SVM.

We tested CUI assignment in the combined system, by training the system on all 77 gold standard documents, and applying it to our development corpus. The numbers of CUIs assigned are shown in Table 3.3. Overall, 83% of all entities were assigned at least one CUI. Only Intervention had more than 20% of entities assigned no CUIs. By this measure, it does seem that the linkage provided by a dictionary-based method is carried through to the combined method. However, there are two problems with this result. First, we cannot be sure of the precision of CUI assignment, as our gold standard does not contain CUIs. It seems likely, however, that as CUI assignment is based on a direct lookup on UMLS terms, precision will be high. Second, a considerable number of entities had more than one CUI assigned: nearly 27% overall. Most of these were assigned two, but a small number were assigned 3 or more. Clearly, some form of disambiguation is needed

— this would also be true of a pure Termino approach. CUI assignment may be viewed as a form of word sense disambiguation, a topic reviewed by Schuemie et al. (2005) for the biomedical domain.

3.7 Conclusion

We have examined entity recognition using dictionary lookup, and using machine learning of statistical models with SVMs. Dictionary lookup based on a very large terminology resource gave good recall, but poor precision. The low precision was largely due to the ambiguity inherent in such terminology resources. We found that much of the ambiguity was due to a small number of terms, and that filtering these out doubled precision. The filter list was hand built using simple heuristics, and used no syntactic information.

SVM based entity recognition, trained on lexico-syntactic features alone, outperformed dictionary lookup in terms of precision, but gave lower recall. In terms of $F1$, the SVM system outperformed dictionary lookup overall, but was much worse for one entity type (`Drug or device`), suggesting that dictionary lookup is especially useful in some cases.

When the SVM was combined with dictionary lookup, by training on term features in addition to the lexico-syntactic features, precision was maintained overall, although it did drop for specific entity types. Recall improved significantly in all cases, although it did not attain the overall recall levels of the best dictionary lookup. This system gave the best overall $F1$ of 0.71, 3% below the overall Inter Annotator Agreement. The combined system also retained an advantage of dictionary lookup, by achieving linkage from recognised entities to domain resources in 83% of cases.

We have shown that large scale terminology resources can be used to benefit clinical entity recognition, and that statistical models can overcome some of the shortcomings of dictionary lookup over such resources.

Availability Most of the software described here is open source and can be downloaded as part of GATE. We are currently packaging Termino for public release, at which point the whole application will be made available.

Acknowledgements

CLEF is funded by the UK Medical Research Council. We would like to thank the Royal Marsden Hospital for providing the corpus, and our clinical partners in CLEF for assistance in developing the annotation schema, and for gold standard annotation.

Chapter 4

Mining clinical relationships from patient narratives

Foreword

The following Chapter is reproduced in full from Roberts et al. (2008d):

> A. Roberts, R. Gaizauskas, M. Hepple, and Y. Guo. Mining clinical relationships from patient narratives. *BMC Bioinformatics*, 9 Suppl 11(S3), November 2008d

An earlier version appeared as Roberts et al. (2008a):

> A. Roberts, R. Gaizauskas, and M. Hepple. Extracting clinical relationships from patient narratives. In *Proceedings of the Workshop on BioNLP 2008*, Columbus, OH, USA, June 2008a. Association for Computational Linguistics

Author's contribution

The author of this thesis wrote the first complete draft of Roberts et al. (2008d), and of the earlier paper (Roberts et al., 2008a), and led the writing of all subsequent drafts. The author made the following contributions to the work described in the paper:

- Contributed to the development of the relationship schema (Section 4.3.1);
- Prepared the corpus (Section 4.3.2);
- Designed and developed the relation extraction system (Section 4.3.3);
- Set up and configured the classifiers (Section 4.3.3.1);

- Developed the feature sets used, with the exception of the two syntactic features (Section 4.3.3.2);
- Developed the evaluation metrics (Section 4.3.4);
- Carried out all experiments and analysed all results, with the exception of the two syntactic features (Section 4.4).

The author did not contribute to the following work described in the paper:

- Development of the syntactic features (final paragraph of Section 4.3.3.2, and Section 4.4.1.2)
- Development of the baseline described in Section 4.4.5

Copyright and permission to use

4.1 Abstract

Background: The Clinical E-Science Framework (CLEF) project has built a system to extract clinically significant information from the textual component of medical records in order to support clinical research, evidence-based healthcare and genotype-meets-phenotype informatics. One part of this system is the identification of relationships between clinically important entities in the text. Typical approaches to relationship extraction in this domain have used full parses, domain-specific grammars, and large knowledge bases encoding domain knowledge. In other areas of biomedical NLP, statistical machine learning (ML) approaches are now routinely applied to relationship extraction. We report on the novel application of these statistical techniques to the extraction of clinical relationships.

Results: We have designed and implemented an ML-based system for relation extraction, using support vector machines, and trained and tested it on a corpus of oncology narratives hand-annotated with clinically important relationships. Over a class of seven relation types, the system achieves an average F1 score of 72%, only slightly behind an indicative measure of human inter annotator agreement on the same task. We investigate

the effectiveness of different features for this task, how extraction performance varies between inter- and intra-sentential relationships, and examine the amount of training data needed to learn various relationships.

Conclusions: We have shown that it is possible to extract important clinical relationships from text, using supervised statistical ML techniques, at levels of accuracy approaching those of human annotators. Given the importance of relation extraction as an enabling technology for text mining and given also the ready adaptability of systems based on our supervised learning approach to other clinical relationship extraction tasks, this result has significance for clinical text mining more generally, though further work to confirm our encouraging results should be carried out on a larger sample of narratives and relationship types.

4.2 Background

Natural Language Processing (NLP) has been widely applied in biomedicine, particularly to improve access to the ever-burgeoning research literature. Increasingly, biomedical researchers need to relate this literature to phenotypic data: both to populations, and to individual clinical subjects. The computer applications used in biomedical research therefore need to support genotype-meets-phenotype informatics and the move towards translational biology. This will undoubtedly include linkage to the information held in individual medical records: in both its structured and unstructured (textual) portions.

The Clinical E-Science Framework (CLEF) project (Rector et al., 2003) is building a framework for the capture, integration and presentation of this clinical information, for research and evidence-based health care. The project's data resource is a repository of the full clinical records for over 20000 cancer patients from the Royal Marsden Hospital, Europe's largest oncology centre. These records combine structured information, clinical narratives, and free text investigation reports. CLEF uses information extraction (IE) technology to make information from the textual portion of the medical record available for integration with the structured record, and thus available for clinical care and research. The CLEF IE system analyses the textual records to extract entities, events and the relationships between them. These relationships give information that is often not available in the structured record. Why was a drug given? What were the results of a physical examination? What problems were not present? The relationships extracted are considered to be of interest for clinical and research applications downstream of IE, such as querying to support clinical research.

The approach taken by the CLEF IE system is one that combines the use of existing

terminology resources with supervised Machine Learning (ML) methods. Models of clinical text are trained from human annotated example documents – a gold standard – which can then be applied to unseen texts. The human-created annotations of the gold standard documents capture examples of the specific content that the IE system is required to extract, providing the system with focussed knowledge of the task domain, alongside the broader domain knowledge provided by more general terminology resources. The advantage of this approach is that the system can be adapted to other clinical domains largely through the provision of a suitable gold standard for that domain, for retraining the system, rather than through the creation of new specialised software components or some major exercise in knowledge engineering.

The approach taken to entity extraction in the CLEF IE system has been described in detail elsewhere (Roberts et al., 2008c). This paper focusses instead on relationship extraction in the CLEF IE system. Our approach uses Support Vector Machine (SVM) classifiers to learn these relationships. The classifiers are trained and evaluated using novel data: a gold standard corpus of oncology narratives, hand-annotated with semantic entities and relationships. We describe a range of experiments that were done to aid development of the approach, and to test its applicability to the clinical domain. We train classifiers using a number of different features sets, and investigate their contribution to system performance. These sets include some comparatively simple text-based features, and others based on a linguistic analysis, including some derived from a full syntactic analysis of sentences. Clinically interesting relationships may span several sentences, and so we compare classifiers trained for both intra- and inter-sentential relationships (spanning one or more sentence boundaries). We also examine the influence of training corpus size on performance, as hand annotation of training data is the major expense in supervised machine learning. Finally, we investigate the impact of imperfect entity recognition on relation extraction performance, by comparing relation extraction done over perfect gold-standard entities to that done over imperfect recognised entities. The paper is an expanded version of Roberts et al. (2008a), but extends that paper with a more detailed description of our relation extraction approach, a more thorough discussion of our earlier experimental results, and a report of some additional experiments and their results (specifically those concerning syntactically-derived features and the impact of imperfect entity recognition).

4.2.1 Previous work

Extracting relations from natural language texts began to attract researchers' attention as a task in its own right during the evolution of information extraction challenges that took

place as part of the Message Understanding Conferences (MUCs) (see e.g. NIST (d)), though of course extraction of relational information from text is a part of any attempt to derive meaning representations from text and hence significantly predates MUC. Specifically, relation extraction emerged as a stand-alone task in MUC-7 (Chinchor, 1998), i.e. requiring participants to extract instances of the employee_of, product_of, and location_of relations, holding between organisations and persons, artefacts and locations respectively, from newswire text. The introduction of this task was part of the factorisation of complex event extraction tasks (for events such as terrorist attacks or joint ventures) that had dominated earlier MUCs, into component tasks that were easier to address and evaluate and would be of relevance in multiple domains (examples of other component tasks factored out in this evolution are named entity recognition and co-reference resolution). The best score obtained on blind test data on this relation extraction task was 75.6% F1-measure (67% precision, 86% recall), where participants had to recognise automatically the entities standing in the relation as well (NIST, d). At the time of MUC-7 the approach adopted by most researchers was to analyse training examples by hand and author patterns to match contexts which expressed the relevant relation. However, even at that time the move away from manually authored extraction patterns towards trainable systems that learned rules or statistical patterns from data was underway, with one participating system (not the highest scoring) using a technique based on automatically augmenting a statistical parser with task specific semantic information obtained from shallow semantic annotation of a training corpus (Miller et al., 1998).

Since the MUC evaluations there has been increasing work on relation extraction, far more than can be reviewed here. This work can be characterised along several dimensions: the text type (e.g. newswire, scientific papers, clinical reports); the relations addressed (e.g. part-of, located-in, protein-protein interaction); the techniques used (e.g. knowledge-engineering rule-based techniques, supervised learning techniques); whether it was carried out in the context of a shared task challenge for which publicly available task definitions, annotated corpora and evaluation software exist (e.g. the ACE relation extraction challenges (Doddington et al., 2004), the LLL genic interaction extraction challenge (Nédellec, 2005a), the BioCreative-II protein-protein interaction task(BioCreAtIvE, 2006)). We concentrate on the points in this space closest to our own work.

There has been little work on relation extraction from clinical texts, presumably because of the difficulty in getting access to texts of this type. In the work carried out to date, extraction of relationships from clinical text is usually carried out as part of a full clinical IE system. Several such systems have been described. They generally use a syntactic parse with domain-specific grammar rules. The Linguistic String project (Sager et al., 1994) used a full syntactic and clinical sub-language parse to fill template data structures

corresponding to medical statements. These were mapped to a database model incorporating medical facts and the relationships between them. MedLEE (Friedman et al., 1994), and more recently BioMedLEE (Lussier et al., 2006) used a semantic lexicon and grammar of domain-specific semantic patterns. The patterns encode the possible relationships between entities, allowing both entities and the relationships between them to be directly matched in the text. Other systems have incorporated large-scale domain-specific knowledge bases. MEDSYNDIKATE (Hahn et al., 2002) employed a rich discourse model of entities and their relationships, built using a dependency parse of texts and a description logic knowledge base re-engineered from existing terminologies. MENELAS (Zweigenbaum et al., 1995) also used a full parse, a conceptual representation of the text, and a large scale knowledge base. Note that all these approaches are knowledge-engineering approaches, based on manually authored grammars, lexicons and ontologies. While supervised machine learning has also been applied to clinical text, its use has generally been limited to entity recognition. The Mayo Clinic text analysis system (Pakhomov et al., 2005), for example, uses a combination of dictionary lookup and a Naïve Bayes classifier to identify entities for information retrieval applications. To the best of our knowledge, statistical methods have not been previously applied to extraction of relationships from clinical text.

By contrast there has been extensive work on relation extraction from biomedical journal papers and abstracts. Much early work in this area and some recent work as well has been done within the hand-written rule base/knowledge engineering paradigm. For example Blaschke et al. (1999); Thomas et al. (2000); Pustejovsky et al. (2002); Fundel et al. (2007); Gaizauskas et al. (2003) all aim to identify gene/protein interactions using simple co-occurrence heuristics or linguistic rules of varying degrees of sophistication to parse sentences and then map syntactic arguments or dependency relations of domain specific verbs into relational structures. Not all the attention has been on protein-protein interactions: Rindflesch et al. (2003) discusses such an approach for extracting causal relations between genetic phenomena and diseases and Ahlers et al. (2007) discusses an extension of this approach to a broad range of relations in pharmacogenetics.

In current work on relation extraction more broadly, however, the dominant trend is using supervised ML techniques to train relation classifiers on human annotated texts. Training examples are typically relation instances expressed as a relation type associated with a linked pair of typed entity mentions tagged in a text. The result is a relation classifier capable of recognising relations in entity-tagged text. Approaches differ chiefly according to the ML algorithms and the features employed. Keeping to applications within biomedicine, researchers have explored maximum entropy approaches (Grover et al., 2007), conditional random fields (Bundschus et al., 2008) and rule learning meth-

ods such as boosted wrapper induction and RAPIER (Bunescu et al., 2005) and inductive logic programming (Goadrich et al., 2005). SVMs have been used for relation extraction, but not extensively in biomedical applications (though see Giuliano et al. (2006)); examples include Zelenko et al. (2002); Zhou et al. (2005); Bunescu and Mooney (2005). We use SVMs due to their generally high performance at classification tasks, as it is in these terms that we have recast relation extraction.

A wide range of features have been explored for use by supervised ML approaches to relation extraction in biomedical applications. Given a sentence (or text) containing entity mentions whose relationships are to be determined, features investigated have included: orthographic and lexical features of the words between entity mentions and possibly outside the context as well (Grover et al., 2007; Bundschus et al., 2008; Giuliano et al., 2006); part-of-speech and other shallow syntactic features of these words (Giuliano et al., 2006); syntactic information, typically dependency parse information, about the grammatical relations between entity mentions (Katrenko and Adriaans, 2007). While all researchers use orthographic and lexical features, the utility of syntactic information remains a topic of debate and one to which the current study contributes.

4.3 Methods

4.3.1 Relationship schema

The CLEF IE system extracts entities, relationships and modifiers from text. By *entity*, we mean some real-world thing, event or state referred to in the text: the drugs that are mentioned, the tests that were carried out, etc. *Modifiers* are words that qualify an entity in some way, referring e.g. to the laterality of an anatomical locus, or the negation of a condition ("no sign of inflammation"). Entities are connected to each other and to modifiers by *relationships*: e.g. linking a drug entity to the condition entity for which it is indicated, linking an investigation to its results, or a negating phrase to a condition. Note that we treat negation as a modifier word, together with its relationship to a condition. This is in contrast to others (for example Chapman et al. (2001)), who identify negated diseases and findings as complete expressions.

The entities, modifiers, and relationships are described by both a formal XML schema, and a set of detailed definitions. These were developed by a group of clinical experts, working in collaboration with a computational linguist, through an iterative process, until acceptable agreement was reached. Entity types are manually mapped to types from the Unified Medical Language System (UMLS) semantic network (Lindberg et al., 1993), each CLEF entity type being mapped to several UMLS types. Relationship types are those

felt necessary to capture the essential clinical dependencies between entities referred to in patient documents, and to support CLEF end user applications. The schema is described further in Roberts et al. (2008b).

Each relationship type is constrained to hold only between pairs of specific entity types, e.g. the `has_location` relation can hold only between a `Condition` and a `Locus`. Some relationships can hold between multiple type pairs. The full set of relationships and their argument types are shown in Table 4.1, with a description and examples of each. The schema is shown graphically in Figure 4.1.

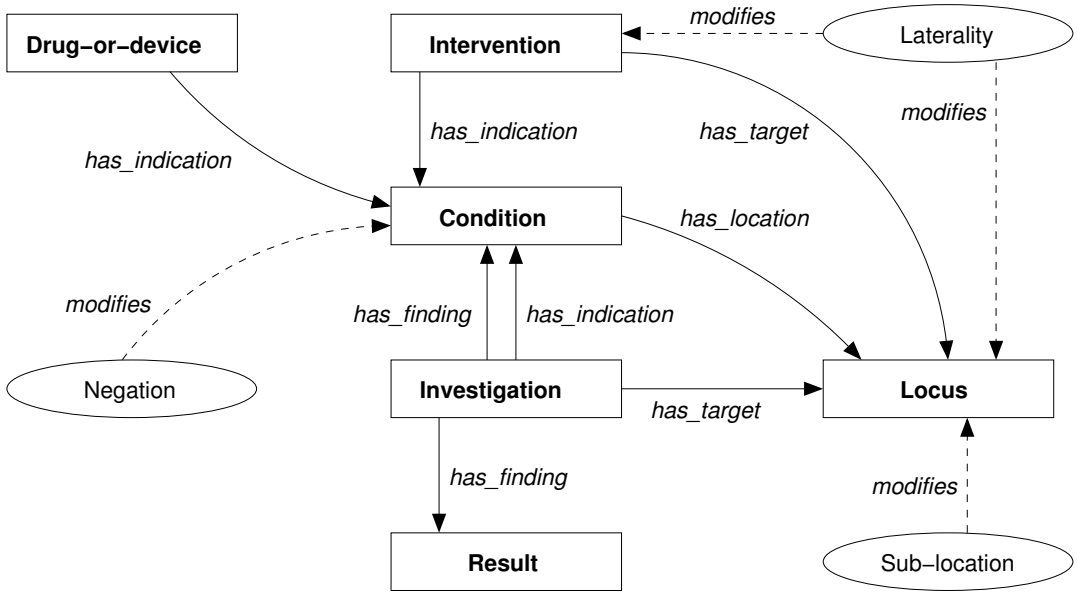

Figure 4.1: **The relationship schema.** The relationship schema, showing entities (rectangles), modifiers (ovals), and relationships (arrows).

Some of the relationships considered important by the clinical experts were not obvious without domain knowledge. For example, in

> *He is suffering from nausea and severe headaches. Dolasteron was prescribed.*

domain knowledge is needed to identify the `has_indication` relation between the drug "Dolasteron" and the "nausea" condition. As in this example, many such relationships are inter-sentential.

A single real-world entity may be referred to several times in the same text. Each of these co-referring expressions is a *mention* of the entity. The schema includes encoding of co-reference between different textual mentions of the same entity. For the work reported in this paper, however, co-reference is ignored, and each entity mention is treated as

Relation type	First argument type	Second argument type	Description	Examples
has_target	**Investigation Intervention**	**Locus**	Relates an intervention or an investigation to the bodily locus at which it is targetted.	• This patient has had a [arg2] lymph node [arg1] biopsy • . . . he does need a [arg2] groin [arg1] dissection
has_finding	**Investigation**	**Condition Result**	Relates a condition to an investigation that demonstrated its presence, or a result to the investigation that produced that result.	• This patient has had a lymph node [arg1] biopsy which shows [arg2] melanoma • Although his [arg1] PET scan is [arg2] normal . . .
has_indication	**Drug or device Investigation Intervention**	**Condition**	Relates a condition to a drug, intervention, or investigation that is targetted at that condition.	• Her facial [arg2] pain was initially relieved by [arg1] co-codamol
has_location	**Condition**	**Locus**	Relationship between a condition and a locus: describes the bodily location of a specific condition.	• . . . a biopsy which shows [arg1] melanoma in his right [arg2] groin • Her[arg2] facial [arg1] pain was initially relieved by co-codamol
negation_modifies	**Negation signal**	**Condition**	Relates a condition to its negation or uncertainty about it.	• There was [arg1] no evidence of extra pelvic [arg2] secondaries
laterality_modifies	**Laterality signal**	**Locus Intervention**	Relates a bodily locus or intervention to its sidedness: *right, left, bilateral*.	• . . . on his [arg1] right [arg2] second toe • [arg1] right [arg2] thoracotomy
sub_location_modifies	**Sub-location signal**	**Locus**	Relates a bodily locus to other information about the location: *upper, lower, extra*, etc.	• [arg1] extra [arg2] pelvic

Table 4.1: **Relationship types and examples.** Relationship types, their argument type constraints, a description and examples. Each example shows a single relation of the given type. Arguments are underlined and preceded by their argument number.

a different entity. Relationships between entities can be considered, by extension, as relationships between the single mentions of those entities. We return to this issue below.

4.3.2 Gold standard corpus

The schema and definitions were used to hand-annotate the entities and relationships in oncology narratives, to provide a gold standard for system training and evaluation. By "narrative" we mean letters, notes, and summaries written by the oncologist, describing the patient's care. Most are very loosely structured, and may be described as consisting of general language with a high terminology content, rather than consisting of formulaic sublanguage or boilerplate. Approval to use this corpus for research purposes within CLEF was obtained from the Thames Valley Multi-centre Research Ethics Committee (MREC). The corpus comprises 77 narratives, which were carefully selected and annotated according to a best practice methodology, as described in Roberts et al. (2008b). Narratives were selected by randomised and stratified sampling from a larger population of 565 000 documents, along various axes such as purpose of narrative and neoplasm. Narratives were annotated by two independent, clinically trained, annotators, and then a consensus annotation created by a third. We refer to the corpus as C77. Corpora of this small size are not unusual in supervised machine learning, and reflect the expense of hand annotation.

Annotators were asked to first mark the mentions of entities and modifiers, and then to consider each in turn, deciding if it had relationships with mentions of other entities. Although the annotators marked co-reference between mentions of the same entity, they were asked to ignore this for relationship annotation. Both the annotation tool and the annotation guidelines enforced the creation of relationships between mentions, not entities. The gold standard is thus analogous to the style of relationship extraction reported here, with relations being assigned between entity mentions, ignoring co-reference. Annotators were further told that relationships could span multiple sentences, and that it was acceptable to use clinical knowledge to infer when a relationship existed. Counts of all relationships annotated in C77 are shown in Table 4.2, sub-divided by the number of sentence boundaries spanned.

	Sentence boundaries between arguments										
	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**>9**
has_finding	265	46	25	7	5	4	3	2	2	2	0
has_indication	139	85	35	32	14	11	6	4	5	5	12
has_location	360	4	1	1	1	1	1	0	0	0	4
has_target	122	14	4	2	2	4	3	1	0	1	0
laterality_modifies	128	0	0	0	0	0	0	0	0	0	0
negation_modifies	100	1	0	0	0	0	0	0	0	0	0
sub_location_modifies	76	0	0	0	0	0	0	0	0	0	0
Total	1190	150	65	42	22	20	13	7	7	8	16
Cumulative total	1190	1340	1405	1447	1469	1489	1502	1509	1516	1524	1540

Table 4.2: **Relationship counts in the gold standard.** Count of relations in 77 gold standard documents, sub-divided by the number of sentence boundaries between relations.

4.3.3 Relationship extraction

Our system is built using the GATE NLP toolkit, which is an architecture allowing language processing applications to be constructed as a pipeline of processing components (Cunningham et al., 2002). Documents are passed down this pipeline, being analysed by each component in turn, with the results of this analysis being available to later components. The system is shown in Figure 4.2, and is described below.

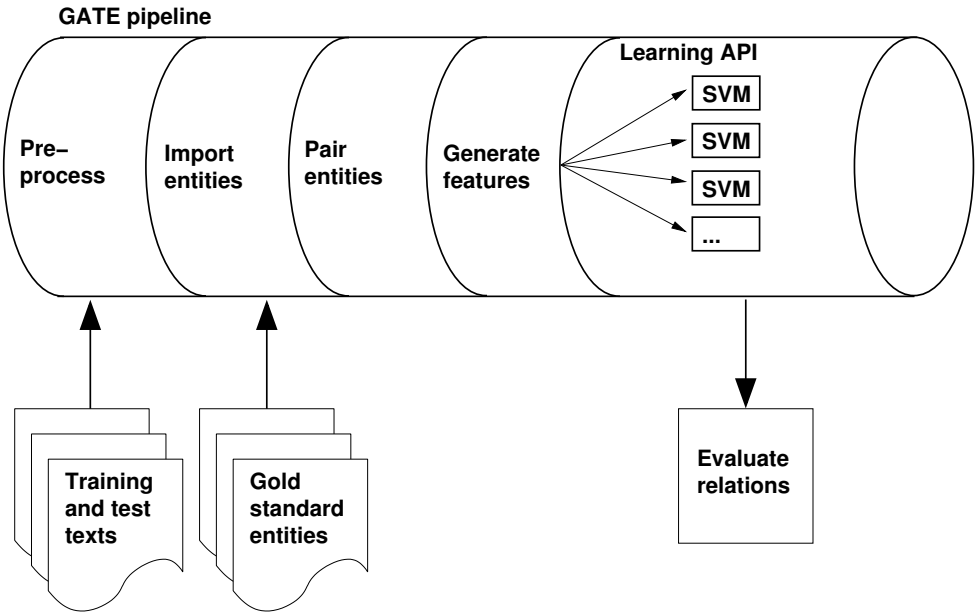

Figure 4.2: **The relationship extraction system.** The relationship extraction system, as a GATE pipeline.

Narratives are first pre-processed using standard GATE modules. Narratives were tokenised, sentences found with a regular expression-based sentence splitter, part-of-speech (POS) tagged, and morphological roots found for word tokens. Each token was also labelled with a more generic POS tag, consisting of the first two characters of the full POS tag. This takes advantage of the Penn Treebank tagset used by GATE's POS tagger, in which related POS tags share the first two characters. For example, all six verb POS tags start with the letters "VB". We will refer to this as a "generalised" POS tag.

After pre-processing, mentions of entities within the text are annotated. In the experiments reported, unless otherwise stated, we assume perfect entity recognition, as given by the entities in the human annotated gold standard described above. Our results are therefore higher than would be expected in a system with automatic entity recognition. It is useful and usual to fix entity recognition in this way, to allow tuning specific to relationship extraction, and to allow the isolation of relation-specific problems. Ultimately, however, relation extraction does depend on the quality of entity recognition. To illustrate

this, we provide a comparison with relations learned from automatic entity recognition, in the Results section.

4.3.3.1 Classification

We treat clinical relationship extraction as a classification task, training classifiers to assign a relationship type to an *entity pair*. An entity pair is a pairing of entities that may or may not be the arguments of a relation. For a given document, we create all possible entity pairs within two constraints. First, entities that are paired must be within n sentences of each other. For all of the work reported here, unless stated, $n \leq 1$ (crossing 0 or 1 sentence boundaries). Second, we constrain the entity pairs created by argument type (Rindflesch and Fiszman, 2003). For example, there is little point in creating an entity pair between a `Drug or device` entity and a `Result` entity, as no relationships exist between entities of these types, as specified by the schema. Entity pairing is carried out by a GATE component developed specifically for clinical relationship extraction. In addition to pairing entities according to the above constraints, this component also assigns features to each pair that characterise its lexical and syntactic qualities (described further in the following section).

The classifier training and test instances consist of entity pairs. For training, an entity pair which corresponds to the arguments of a relationship present in the gold standard is assigned that relationship type as its class — or the class `null` if there is no corresponding gold standard relation. The classifier builds a model of these entity pair training instances, from their features. In classifier application, entity pairs are created from unseen text, under the above constraints. The classifier assigns one of our seven relationship types, or `null`, to each entity pair.

We use SVMs as trainable classifiers, as these have proved to be robust and efficient for a range of NLP tasks, including relation extraction. We use an SVM implementation developed within our own group, and provided as part of the GATE toolkit. This is a variant on the original SVM algorithm, SVM with uneven margins, in which classification may be biased towards positive training examples. This is particularly suited to NLP applications, in which positive training examples are often rare. Full details of the classifier are given in Li et al. (2005). We used the implementation "out of the box", with default parameters as determined in experiments with other data sets.

The SVM with uneven margins algorithm is a binary classifier. Thus to apply it to a multi-class problem requires mapping the problem to a number of binary classification problems. There are several ways in which a multi-class problem can be recast as binary problems. The commonest are *one-against-one* in which one classifier is trained for every possible pair of classes, and *one-against-all* in which a classifier is trained for a binary

decision between each class and all other classes, including `null`, combined. We have carried out extensive experiments (not reported here), with these two strategies, and have found little difference between them for our data. We have chosen to use one-against-all, as it needs fewer classifiers (for an n class problem, it needs n classifiers, as opposed to $n(n-1)/2$ for one-against-one).

The resultant class assignments by multiple binary classifiers must be post-processed to deal with ambiguity. In application to unseen text, it is possible that several classifiers assign different classes to an entity pair (test instance). To disambiguate these cases, the output of each one-against-all classifier is transformed into a probability, and the class with the highest probability is assigned. Re-casting the multi-class relation problem as a number of binary problems, and post-processing to resolve ambiguities, is handled by the GATE Learning API.

4.3.3.2 Features for classification

The SVM classification model is built from lexical and syntactic features assigned to tokens and entity pairs prior to classification. We use features developed in part from those described in Zhou et al. (2005) and Wang et al. (2006). These features are split into 15 sets, as described in Table 4.3.

The `tokN` features are POS and surface string taken from a window of N tokens on each side of both paired entities. For $N = 6$, this gives 48 features. The rationale behind these simple features is that there is useful information in the words surrounding the two mentions, that helps determine any relationship between them. The `gentokN` features generalise `tokN` to use morphological root and generalised POS. The `str` features are a set of 14 surface string features, encoding the full surface strings of both entity mentions, their heads, their heads combined, the surface strings of the first, last and other tokens between the mentions, and of the two tokens immediately before and after the leftmost and rightmost mentions respectively. The `pos`, `root`, and `genpos` feature sets are similarly constructed from the POS tags, roots, and generalised POS tags of the entity mentions and their surrounding tokens. These four feature sets differ from `tokN` and `gentokN`, in that they provide more fine-grained information about the position of features relative to the paired entity mentions.

For the `event` feature set, entities were divided into events (`Investigation` and `Intervention`) and non-events (all others). Features record whether an entity pair consists of two events, two non-events, or one of each, and whether there are any intervening events or non-events. This feature set gives similar information to `atype` (semantic types of arguments) and `inter` (intervening entities), but at a coarser level of typing. The feature sets `allgen` and `notok` are combinations of the above feature sets, as specified by

Feature set	Size	Description
tokN	$8N$	Surface string and POS of tokens surrounding the arguments, windowed $-N$ to $+N$, $N = 6$ by default
gentokN	$8N$	Root and generalised POS of tokens surrounding the argument entities, windowed $-N$ to $+N$, $N = 6$ by default
atype	1	Concatenated semantic type of arguments, in arg1-arg2 order
dir	1	Direction: linear text order of the arguments (is arg1 before arg2, or vice versa?)
dist	2	Distance: absolute number of sentence and paragraph boundaries between arguments
str	14	Surface string features based on Zhou et al. (2005), see text for full description
pos	14	POS features, as above
root	14	Root features, as above
genpos	14	Generalised POS features, as above
inter	11	Intervening mentions: numbers and types of intervening entity mentions between arguments
event	5	Events: are any of the arguments, or intevening entities, events?
allgen	96	All above features in root and generalised POS forms, i.e. gentok6+atype+dir+dist+root+genpos+inter+event
notok	48	All above except tokN features, others in string and POS forms, i.e. atype+dir+dist+str+pos+inter+event
dep	16	Features based on a syntactic dependency path.
syndist	2	The distance between the two arguments, along a token path and along a syntactic dependency path.

Table 4.3: **Feature sets for learning.** Feature sets used for learning relationships. The table is split into non-syntactic features, combined non-syntactic features, and syntactic features. The size of a set is the number of features in that set.

the descriptions in Table 4.3.

For the final two feature sets shown in Table 4.3, we used the Stanford Parser (Klein and Manning, 2003) to parse the C77 corpus. This parser generates a dependency analysis, consisting of a graph of syntactic relations amongst sentence tokens. The feature set `dep` consists of 16 features derived from the parse, which are only computed when the entities appear in the same sentence (and otherwise take value `null`). The features encode characteristics of the dependency path connecting the paired entities, of the immediate left context in the dependency analysis of the leftmost entity, and of the corresponding right context of the rightmost entity. The `syndist` set adds two further features, which firstly count the number of links on the dependency path connecting the paired entities and the number of tokens between the two entities, and then maps these values to labels NEAR, MIDDLE and FAR, to reduce data sparseness.

4.3.4 Evaluation methodology

We use the standard evaluation metrics of Recall and Precision, which are defined in terms of true positive (TP), false positive (FP) and false negative (FN) matches between relations recorded in a system annotated *response* document and a gold standard *key* document. A response relation is a true positive if a relation of the same type, and with the exact same arguments, exists in the key. Corresponding definitions apply for false positive and false negative. Counts of these matches are used to calculate Recall (R) and Precision (P) scores, as defined below. The harmonic mean of these two values provides a single combined indicator of performance. This metric, known as $F1$, as also defined below.

$$R = \frac{TP}{TP + FN} \quad P = \frac{TP}{TP + FP} \quad F1 = \frac{2PR}{P + R}$$

We used a standard ten-fold cross validation methodology in our experiments. Various tables given later report the results of these experiments, showing recognition scores for the different relation types and for relation recognition overall. The scores for individual relations are produced by computing the P, R and $F1$ scores for each relation type on each fold, and then macro-averaging these values (i.e. computing their simple mean) across the folds to give the corresponding relation-specific cross-validated score. This approach can produce results that may at first sight seem anomalous, e.g. cases where the $F1$ score for a given relation does not fall between the P and R scores. Overall scores for relation recognition are produced by first micro-averaging scores for the different relation types within the fold, i.e. simply adding their counts for true-positives, false-negatives and false-positives, and using these summed values to compute P and R values directly. The resulting combined scores are then macro-averaged across folds to produce the cross-

validated overall scores.

The metrics do not say how hard relationship extraction is. We therefore also provide Inter Annotator Agreement (IAA) scores from the creation of the gold standard. The IAA measures the level of agreement between the two annotators who independently annotated each text to produce its double annotation. It is equivalent to scoring one annotator against the other using the $F1$ metric (i.e. treating one annotation as key and the other as response).

IAA scores are not directly comparable here to system extraction scores, as relationship annotation is a slightly different task for the human annotators. The relationship extraction system is given entities, and finds relationships between them. Human annotators must find both the entities and the relationships. Where one human annotator fails to find a particular entity, they can never find its relationships. The raw IAA score does not take this into account: if an annotator fails to find an entity, they will also be penalised for all relationships with that entity. We therefore give a Corrected IAA (CIAA) in which annotators are only compared on those relations for which they have both found the entities involved. In our results, we give both IAA and CIAA, for each relation type and for relations overall. As our results will show, it is clear that it is difficult for annotators to reach agreement on relationships, some more so than others. Further, lower values for IAA than for CIAA show this difficulty is compounded massively by lack of agreement on entities. The level of agreement that is achieved between annotators is often seen as providing an *upper bound* for what can be expected of system performance. The situation here however is complicated by the fact that the gold standard used in training and evaluation is produced by a further consensus process, so that gold standard annotations may exhibit a greater degree of regularity, reliability and correctness than can be expected of the output of any one annotator, making it at least possible for the system to score higher on some relation than the observed annotator agreement level.

A second basis for evaluating system performance is comparison against *baseline* scores for the given task, which are scores that can be achieved using some quite simplistic method. Baseline scores can be viewed as providing a (reasonable) *lower bound* for performance, and the improvement over the baseline is a measure of the benefit achieved by using a more complex approach. For classification tasks, a common baseline is to assign to all members of a group of instances the most common class found for that group within the gold standard. A baseline method for relation extraction will begin with the set of possible entity pairs for each document, as discussed earlier for our relation recognition method proper, where the possible entity pairs are restricted to only those whose entities are of suitable types, and which occur in the same or adjacent sentences, and each entity pair assigned as their class either a relation type from the gold standard or the value `null`.

An obvious baseline approach is to subdivide this overall set of instances (i.e. possible pairs) into subsets in terms of the types of the two entities, and for each subset to determine the most common class and assign this as the default to all instances in the subset. If the most common class is `null`, then all the entity pairs will be treated as unrelated.

More complicated baseline methods might use further criteria for subdividing the possible entity pairs into subsets for which most common classes are computed. In this paper, we also consider baselines using the left-right order of the two entities or whether they appear in the same sentence or not. Going too far along this route, however, can lead to more complicated methods that do not obviously deserve the title 'baseline', and can involve the work that is most naturally done by machine learning methods being laboriously reproduced as a manual feature engineering task.

4.4 Results and discussion

4.4.1 Feature selection

We next report experiments regarding the features most useful for relation extraction, using the features sets described in Table 4.3. We divide the discussion between the case of features sets that do not use syntactic parse information and those that do.

4.4.1.1 Non-syntactic features

The first group of experiments reported looks at the performance of relation extraction with non-parse feature sets. We followed an additive strategy for feature selection: starting with basic features, we added further features one set at a time. We measured the performance of the resulting classifier each time we added a new feature set. Results are shown in Table 4.4. The initial classifier used a `tok6+atype` feature set. Addition of both `dir` and `dist` features give significant improvements in all metrics, of around 10% $F1$ overall, in each case. This suggests that the linear text order of arguments, and whether relations are intra- or inter-sentential is important to classification. Addition of the `str` features also give good improvement in most metrics, again 10% $F1$ overall. Addition of part-of-speech information, in the form of `pos` features, however, leads to a drop in some metrics, overall $F1$ dropping by 1%. Unexpectedly, POS seems to provide little extra information above that in the surface string. Errors in POS tagging cannot be dismissed, and could be the cause of this. The existence of intervening entities, as coded in feature set `inter`, provides a small benefit. The inclusion of information about events, in the `event` feature set, is less clear-cut.

Relation	Metric	tok6+ atype	+dir	+dist	+str	+pos	+inter	+event	allgen	notok
has_finding	**P**	44	49	58	63	62	64	65	63	63
	R	39	63	78	80	80	81	81	82	82
	F1	39	54	66	70	69	71	72	71	71
has_indication	**P**	37	23	38	42	40	41	42	37	44
	R	14	14	46	44	44	47	47	45	47
	F1	18	16	39	39	38	41	42	38	41
has_location	**P**	36	36	50	68	71	72	72	73	73
	R	28	28	74	79	79	81	81	83	83
	F1	30	30	58	72	74	76	75	77	76
has_target	**P**	9	9	32	63	57	60	62	60	59
	R	11	11	51	68	67	67	66	68	68
	F1	9	9	38	64	60	63	63	63	62
laterality_ modifies	**P**	21	38	73	84	83	84	84	86	86
	R	9	55	82	89	86	88	88	87	89
	F1	12	44	76	85	83	84	84	84	85
negation_ modifies	**P**	19	54	85	81	80	79	79	77	81
	R	12	82	97	98	93	92	93	93	93
	F1	13	63	89	88	85	84	85	83	85
sub_location_ modifies	**P**	2	2	55	88	86	86	88	88	87
	R	1	1	62	94	92	95	95	95	95
	F1	1	1	56	90	86	89	91	91	90
Overall	**P**	33	38	50	63	62	64	65	64	64
	R	22	36	70	74	73	75	75	76	76
	F1	26	37	58	68	67	69	69	69	70

Table 4.4: **Performance by feature set, non-syntactic features.** Variation in performance by feature set, non-syntactic features. Features sets are abbreviated as in Table 4.3. For the first seven columns, features were added cumulatively to each other. The next two columns, `allgen` and `notok`, are as described in Table 4.3.

We were interested to see if generalising features could improve performance, as this had benefited our previous work in entity extraction. We replaced all surface string features with their root form, and POS features with their generalised POS form. This gave the results shown in column `allgen`. Results are not clear cut, in some cases better and in some worse than the previous best. Overall, there is no difference in $F1$. There is a slight increase in overall recall, and a corresponding drop in precision — as might be expected.

Both the `tokN`, and the `str` and `pos` feature sets provide surface string and POS information about tokens surrounding and between related entities. The former gives features from a window around each argument. The latter two provide more positional information. Do these two provide enough information on their own, without the windowed features? To test this, we removed the `tokN` features from the full cumulative feature set, corresponding to column `+event` of Table 4.4. The results, in column `notok`, show no clear change in performance, with some relationships improving, and some worsening. Overall, there is a 1% improvement in $F1$.

It appears that the bulk of performance is attained through entity type and distance features, with some contribution from positional surface string information. Performance is between 1% and 9% lower than CIAA for each relationship, with a best overall $F1$ of 70%, compared to a CIAA of 75%.

4.4.1.2 Syntactic features

The remaining feature selection experiments look at the impact of using features derived from a dependency parse analysis of the clinical texts made using the Stanford parser (Klein and Manning, 2003), which is a dependency parser that has been developed principally in relation to newswire texts. Despite the very different genre of our clinical texts, which are heavily laden with medical language, we did not attempt to adapt the Stanford parser to the domain, hoping rather that we could still benefit from exploiting whatever dependency analysis the parser is able to produce.

Table 4.5 reiterates the `+event` column of Table 4.4, corresponding to the accumulation of *all* non-syntactic feature sets, and gives results for augmenting this set with the syntactic features of `dep` and then also `syndist`. The syntactic features contribute mainly to finding the `has_indication` and `negation_modifier` relations, with an improved $F1$ of around 4% for each, while retaining performance for other relations. Overall we see a 3% increase in $F1$ to 72%, a step closer to the CIAA of 75%. The results illustrate that the SVM classifiers can exploit the more abstract information of underlying dependency relations, to generalise beyond the surface information of token strings and distances.

Given that the dependency analyses produced by the parser do not cross sentence boundaries (i.e. they are analyses of individual sentences), and since our syntactically-

Relation	Metric	+event	+dep	+syndist
has_finding	**P**	65	73	74
	R	81	77	77
	F1	72	71	74
has_indication	**P**	42	42	43
	R	47	37	37
	F1	42	38	39
has_location	**P**	72	74	73
	R	81	86	86
	F1	75	79	78
has_target	**P**	62	65	71
	R	66	63	66
	F1	63	62	64
laterality_modifies	**P**	84	89	89
	R	88	84	90
	F1	84	85	89
negation_modifies	**P**	79	85	85
	R	93	97	93
	F1	85	90	88
sub_location_modifies	**P**	88	90	93
	R	95	95	95
	F1	91	92	94
Overall	**P**	65	71	71
	R	75	74	74
	F1	69	72	72

Table 4.5: **Performance by feature set, syntactic features.** Variation in performance by feature set, syntactic features. The first column shows the cumulative `+event` system from Table 4.4. The next two columns show the effect of cumulatively adding syntactic features to this system. Syntactic features are as described in Table 4.3.

derived features are only computed for entities in the same sentence, we can expect their use to have a positive impact only on the discovery of *intra*-sentential relations. We found that a system using the syntactic feature set `+syndist` and applied to only the intra-sentential relations achieves an $F1$ of 77% (with P=70%, R=84%), as compared to a system using the non-syntactic feature set `+event` on the same intra-sentential subset of relations (corresponding to the $n < 1$ column of Table 4.6), i.e. giving a 2% improvement in $F1$ overall.

		Number of sentence boundaries between arguments						
		inter-	intra-	inter- and intra-sentential				
Relation	**Metric**	**1 ≤ n ≤ 5**	**n < 1**	**n ≤ 1**	**n ≤ 2**	**n ≤ 3**	**n ≤ 4**	**n ≤ 5**
has_finding	**P**	24	68	65	62	60	61	61
	R	18	89	81	79	78	78	77
	F1	18	76	72	69	67	68	67
has_indication	**P**	18	49	42	42	36	32	30
	R	17	59	47	42	42	39	38
	F1	16	51	42	39	37	34	33
has_location	**P**	n/a	74	72	73	72	72	72
	R	n/a	83	81	81	81	82	82
	F1	n/a	77	75	76	75	76	76
has_target	**P**	3	64	62	59	60	59	58
	R	1	75	66	64	62	61	61
	F1	2	68	63	61	60	60	59
laterality_modifies	**P**	n/a	86	84	86	86	86	87
	R	n/a	89	88	88	88	87	88
	F1	n/a	85	84	85	86	85	86
negation_modifies	**P**	n/a	80	79	79	80	80	80
	R	n/a	94	93	91	93	93	93
	F1	n/a	86	85	84	85	86	85
sub_location_modifies	**P**	n/a	89	88	88	89	89	89
	R	n/a	95	95	95	95	95	95
	F1	n/a	91	91	91	91	91	91
Overall	**P**	22	69	65	64	62	61	60
	R	17	83	75	73	71	70	70
	F1	19	75	69	68	66	65	65

Table 4.6: **Performance by sentences.** Variation in performance, by number of sentence boundaries (n) crossed by a relationship. For all cases, the cumulative feature set `+event` of Table 4.4 was used. For the inter-sentential-only classifier $1 \leq n \leq 5$, the score fields for some relations are marked as `n/a` (not applicable). This is because some relations are either absent from the inter-sentential data (i.e. only ever appear intra-sententially), or are so rare that they do not appear in all training/test folds, and so an macro-average cannot be computed across the folds.

4.4.2 Sentences spanned

Table 4.2 shows that although intra-sentential relations account for a clear majority (77%) of relationships, 23% are inter-sentential, with 10% of all relationships holding between entities in adjacent sentences. If we consider a relationship to cross n sentence boundaries, then the classifiers described above have mostly been trained on relationships crossing $n \leq 1$ sentence boundaries, i.e. with arguments in the same or adjacent sentences. What effect does including more distant relationships have on performance? To investigate this question, we trained classifiers for the subset of relationships found under a number of different distance conditions, in all cases using the cumulative feature set `+event` from Table 4.4, producing the results shown in Table 4.6. The first column shows results for a classifier of purely *inter*-sentential relations, for the case $1 \leq n \leq 5$ (which covers 85% of all inter-sentential relations), which can be seen to perform badly for the relations for which the approach applies. (Note that some relations occur across sentence boundaries either rarely or not at all, and so have been discounted in the results.) The next two columns compare classifiers trained on only intra-sentential relationships ($n < 1$) and those spanning up to one boundary ($n \leq 1$). The latter shows that even inclusion of relationships in adjacent sentences produces a 6% drop in overall $F1$ as compared to the purely intra-sentential case. Performance continues to drop as more inter-sentential relationships are included, as the remaining columns show.

A preliminary analysis of the data suggests that the further apart the related entities are, the more likely that clinical knowledge is required to extract the relationship, and such knowledge is clearly not available to the extraction approach described.

4.4.3 Size of training corpus

The provision of sufficient training data for supervised learning algorithms is a limitation on their use. We examined the effect of training corpus size on relationship extraction. We selected subsets consisting of 25 and 50 documents from the C77 corpus, itself comprising 77 narratives, to produce sub-corpora that we refer to as C25 and C50, respectively. We trained two classifiers on these new corpora, again using the cumulative feature set `+event`, to give the results shown in Table 4.7. The table also shows the counts of the training instances for each relation type in the different corpora. Overall, performance improves as training corpus size increases ($F1$ rising from 63% to 69%), as expected. It is notable, however, that the performance for some relations (`negation_modifies` and `has_location`) appears to have plateaued even with this limited amount of training data, although it remains possible that a further increase in size may improve performance.

		Corpus size		
Relation	**Metric**	**C25**	**C50**	**C77**
has_finding	**Count**	91	216	311
	P	66	63	65
	R	74	74	81
	F1	67	67	72
has_indication	**Count**	91	117	224
	P	22	25	42
	R	30	31	47
	F1	23	25	42
has_location	**Count**	127	199	364
	P	72	71	72
	R	76	80	81
	F1	73	74	75
has_target	**Count**	51	90	136
	P	65	49	62
	R	60	65	66
	F1	59	54	63
laterality_modifies	**Count**	57	73	128
	P	77	78	84
	R	69	68	88
	F1	72	69	84
negation_modifies	**Count**	34	67	101
	P	78	79	79
	R	80	93	93
	F1	78	84	85
sub_location_modifies	**Count**	30	43	76
	P	64	91	88
	R	64	85	95
	F1	64	86	91
Overall	**Count**	481	805	1340
	P	62	63	65
	R	65	71	75
	F1	63	66	69

Table 4.7: **Performance by corpus size.** Variation in performance by training corpus size. The "Count" row gives the number of training instances of a relation type, for the given corpus. The cumulative feature set `+event` of Table 4.4 was used.

4.4.4 Extracting relations over extracted entities

The experiments described so far assume perfect entity recognition, using the entities of the gold standard as input to the relation extraction process, both for training and testing. This move is useful in allowing us to isolate the complexities of relation extraction from the vagaries of imperfect entity recognition when the method for performing the former task is under development. In operational use of the IE system, however, the limitations of entity recognition will impact the performance of relation extraction. To get a measure of this effect, we evaluated the system when applied to test data containing imperfect, extracted entities.

The entity recognition approach is as described in Roberts et al. (2008c), using a combination of lexical lookup and supervised ML. Lexical lookup uses the Termino terminology resource (Harkema et al., 2004b). A Termino database is loaded with terms from the UMLS Metathesaurus (Lindberg et al., 1993). Finite state recognisers are compiled from this database, and used to annotate terms in texts. These terms, together with a number of token-level features, are then used to train SVM classifiers: one for each entity type. This approach has been evaluated using ten fold cross validation over the C77 corpus (described above), achieving an overall $F1$ for entity recognition of 71%, macro-averaged across folds (full results are given in Roberts et al. (2008c)).

We again used ten-fold cross validation to evaluate relation extraction with extracted entities. For each of the ten testing folds, the corresponding nine folds of gold standard data were used to train both an entity recognition model and a relation recognition model, the latter again using the `+event` feature set. The entity recognition model was then applied to the test fold to produce a version containing the recognised entities, and the relation recognition model applied to this version, i.e. using the recognised entities as the basis for creating the set of possibly-related entity pairs, to which the relation classifiers are applied. The relation results are then scored against the gold standard version of the test fold, with overall scores being macro-averaged across folds, as reported in Table 4.8. As anticipated, precision for relation recognition over extracted entities generally matches that over gold standard entities, but recall of relations suffers badly, with the overall $F1$ dropping from 70% to 48%. Performance does, however, remain close to IAA (Table 4.9), which measures an analogous human task in which annotators must find both entities and relations. Clearly, good relation extraction depends on good entity recognition.

Relation	Metric	gold standard entities	extracted entities
has_finding	**P**	63	62
	R	82	32
	F1	71	41
has_indication	**P**	44	44
	R	47	27
	F1	41	32
has_location	**P**	73	68
	R	83	49
	F1	76	55
has_target	**P**	59	47
	R	68	39
	F1	62	41
laterality_modifies	**P**	86	83
	R	89	76
	F1	85	74
negation_modifies	**P**	81	81
	R	93	53
	F1	85	60
sub_location_modifies	**P**	87	71
	R	95	24
	F1	90	31
Overall	**P**	64	63
	R	76	40
	F1	70	48

Table 4.8: **Performance over extracted entities.** Performance of relation extraction over automatically extracted entities, compared to relation extraction using perfect gold standard entities. For relation extraction, the cumulative feature set `+event` of Table 4.4 was used.

Relation	Metric	notok (best system: non-syntactic)	+syndist (best system: syntactic)	baseline	IAA	CIAA
has_finding	**P**	63	74	65		
	R	82	77	76		
	F1	71	74	70	46	80
has_indication	**P**	44	43	0		
	R	47	37	0		
	F1	41	39	0	26	50
has_location	**P**	73	73	0		
	R	83	86	0		
	F1	76	78	0	55	80
has_target	**P**	59	71	0		
	R	68	66	0		
	F1	62	64	0	42	63
laterality_modifies	**P**	86	89	60		
	R	89	90	91		
	F1	85	89	72	73	94
negation_modifies	**P**	81	85	81		
	R	93	93	98		
	F1	85	88	88	66	93
sub_location_modifies	**P**	87	93	50		
	R	95	95	68		
	F1	90	94	58	49	96
Overall	**P**	64	71	36		
	R	76	74	48		
	F1	70	72	41	47	75

Table 4.9: **Overall performance evaluation.** System best performance figures (from Tables 4.4 and 4.5), and comparison to baseline performance and to inter-annotator agreement scores.

The relation models used in this evaluation were trained over texts containing gold standard entities. For relation extraction over test data containing imperfect recognised entities, however, it may be that better performance would result with models also *trained* over data containing imperfect entities, but this issue can only be answered empirically.

4.4.5 Summary of key results

Table 4.9 provides a summary of the key performance figures for the overall system, showing results for the best system configuration using only non-syntactic features (`notok`) and for the best one using syntactic features (`+syndist`). For most relation types, the syntactic system outperforms the non-syntactic one, with a macro-averaged $F1$ that is higher by 2–4%, (the exception being a 2% drop for the `has_indication` relation), giving a 2% increase in $F1$ overall. The table also provides scores for a baseline approach (to be detailed shortly) and for inter-annotator agreement, in both IAA and CIAA variants. We can see that IAA scores fall well below the system scores for all relation types, with an overall IAA of 47% compared to the overall system best of 72%, which shows simple IAA to be too pessimistic as an indicator of the likely upper bound of system performance, as expected. In contrast, CIAA scores are fairly close to, and mostly above, the system scores (the sole exception being a `+syndist` system score for `has_target` that is 1% above CIAA).

The baseline scores in the table are for a baseline system assigning different default relations to possibly-related entity pairs based on the types of the two entities, plus their left-right order and whether they appear in the same sentence or not. Other baselines were tried where only one of the latter two criteria, or neither, was used, but these showed much worse performance. The baseline scores were produced directly over the gold standard, i.e. with the set of possibly-related entity pairs being computed from the gold standard entities. For some relation types (e.g. `has_target`), we see $F1$ scores of 0%, showing that no correct instances of the relation were assigned. For some other relation types, however, this baseline approach works quite well, e.g. for `has_finding` we get a baseline $F1$ of 70%, which compares to a best system performance of 74% and a CIAA of 80%, whilst for `negation_modifies` we get a baseline $F1$ of 88%, which equals the best system performance and falls not far below the CIAA of 93%. Overall, however, the baseline method performs much worse than the best system, giving a macro-averaged $F1$ of 41% against a best system $F1$ of 72% and a CIAA of 75%. The simplest baseline, using only the types of the two entities, was found to score 0% for *all* measures (which followed from it having a `null` default for all cases).

4.5 Conclusions

We have shown that it is possible to extract clinical relationships from text, using a supervised machine learning approach. IAA scores suggest that the task is difficult, but our system performs well, achieving an overall $F1$ of 72%, just 3% below corrected IAA. Although reasonable performance is achieved using quite simple surface/token-based features, our experiments indicate a real gain from using also features based on the more complex linguistic analysis provided by a dependency parser. We believe that this work has implications for clinical text mining more generally, given the success of our approach and its adaptability for other clinical domains, though further work to confirm our encouraging results should be carried out on a larger sample of narratives and relationship types. The technology used has proved scalable. The full CLEF IE system, including automatic entity recognition, is able to process a document in sub-second time on a commodity workstation. We have used the system to extract 6 million relations from over half a million patient documents, for use in downstream CLEF applications.
Availability: The software described is open source and can be downloaded as part of GATE (University of Sheffield, 2012), except for the entity pairing component, which will be released shortly. We are currently preparing a UK research ethics committee application, for permission to release our annotated corpus.

4.6 Competing interests

The authors declare that they have no competing interests.

4.7 Authors' contributions

AR designed and built the system using the GATE framework, wrote most of the new components, prepared the data, contributed to evaluation tests, and drafted the manuscript. RG and MH are the principal and co-investigators of the project. They participated fully in the conceptual design of the CLEF IE approach and system, and of the reported experimental work, and contributed to the writing of the manuscript. YG implemented and evaluated the system augmentations for using syntactic features, and drafted the relevant manuscript sections. All authors read and approved the final manuscript.

4.8 Acknowledgements

CLEF is funded by the UK Medical Research Council, grant reference GO300607. We would like to thank the Royal Marsden Hospital for providing the corpus, and our clinical partners in CLEF for assistance in developing the schema, and for gold standard annotation.

Chapter 5

Conclusions

The aim of the research presented in this thesis is to lower the barrier to building clinical IE systems. The intention is that by separating linguistic, domain and engineering knowledge, we will be able to re-use pre-existing resources in these areas, and we will be able to better focus the available skills of domain experts, computational linguists, and software engineers.

Our objectives were threefold: to adopt a supervised ML approach, building and using a manually annotated corpus of clinical text; to re-use existing knowledge resources alongside supervised ML; and to use existing software frameworks as far as possible. Section 5.1 of this conclusion revisits these objectives. This is followed by an examination of the impact of the reported research in Section 5.2. Finally, Section 5.3 discusses unresolved questions and future work.

5.1 Summary of achievements

5.1.1 A supervised ML approach to clinical IE

Our first objective was to adopt a supervised ML approach to clinical text, using domain experts to provide a corpus of examples that capture the semantics of medical language, including the classes of entities described in clinical text, and the relationships between these entities.

Chapter 2 presented the construction of such a corpus, and summarised results of entity and relation extraction, using supervised ML trained with this corpus. The chapter described a rigorous methodology developed for corpus construction, and the measures used for assessing corpus quality. We believe that the resulting corpus, by virtue of containing entities, their properties, co-reference, relations, and negation, is still the most richly annotated resource for clinical IE built, and was novel in its inclusion of clinical

relations, negation and other modifiers. Following Chapter 2, Chapters 3 and 4 gave further details of the clinical IE system, describing entity extraction and relation extraction respectively. Taken together, these illustrate the novel application of supervised ML to a full clinical IE system.

The methodology included the stratified random selection of representative material from three different kinds of clinical text: narratives, imaging reports, and histopathology reports. A schema for annotating these texts was developed from requirements drawn up by clinicians and computational linguists, and mapped to an existing and widely used medical knowledge resource, the UMLS. The schema was applied to the texts through manual annotation to a set of guidelines. These guidelines were developed though a rigorous, iterative process to ensure that they reflected domain requirements. In addition to giving a high-level philosophy of what should and should not be annotated, the guidelines also described how annotation should proceed, in order that they were consistently applied.

The quality of entity annotations was measured using a standard measure of IAA, and a Corrected IAA metric was developed to measure the quality of relation annotations independently of the entities that comprised their arguments. Documents were double annotated, and for those with sufficient agreement, differences were resolved to give a final consensus set. The effect of annotator expertise was examined, and results of this suggest that both linguistic and domain knowledge are required for the highest quality annotation. We also gave results suggesting that the guidelines could be quickly adapted to different text sub-genres. The final gold standard corpus consisted of 3828 entity annotations with IAA of 62% to 69% depending in sub-genre, and 2450 relation annotations, with CIAA of 72% to 76%.

A clinical IE system was built, using SVMs as classifiers. The novel application of supervised learning to relations, negation and co-reference in clinical text, makes this system the most complete attempt at a fully supervised clinical IE system. The system was built using a held-out development corpus, and evaluated against the gold standard. The system achieved an overall $F1$ for entity extraction of 70.7%, 3% below the IAA for the annotations used. For relation extraction, the $F1$ was 70%, 5% below the CIAA.

5.1.2 Coupling medical domain resources and supervised ML

The second objective was to examine the use of pre-existing medical terminologies and knowledge resources in clinical IE, asking: can these resources be successfully coupled with supervised ML to enhance its performance?

Chapter 2 discussed the mapping of the CLEF annotation schema to UMLS, enabling

the utilisation of UMLS source vocabularies in annotation. The implementation of this was described in Chapter 3. The Chapter described three entity recognition methods. The first method was based on FSRs built from the contents of UMLS, using Termino; terms considered unsuitable for recognition were removed based on a set of experimentally-derived heuristics. The second method was based on SVMs trained on the gold standard described in Chapter 2, using simple lexico-syntactic features. The third method was also based on SVMs and the same lexico-syntactic features, but combined with additional features based on the output of the first method, i.e. terms recognised by the FSR. Overall, the UMLS-based FSR showed highest recall (70%) but lowest precision (52%), whereas the simple SVM showed highest precision (79%) but lowest recall (54%). The combined method benefited from the recall of the UMLS-based FSR (63%), with no loss in the precision of the simple SVM (80%). The combined system also retained an advantage of dictionary lookup, by achieving linkage from recognised entities to domain resources in 83% of cases.

In addition to lexical knowledge, medical knowledge resources contain extensive relational information. There may be potential for using this in relation extraction, but this was not investigated.

5.1.3 Building an off-the-shelf clinical IE system

Our final objective was to build a clinical IE system with off-the-shelf NLP and ML frameworks, with the minimum of tailoring. We asked if such frameworks are sufficiently advanced that constructing a clinical IE system can become a software engineering or end-user task.

Chapter 3 and Chapter 4 detailed the component parts of a clinical IE system, with the full system being presented in Chapter 4, and consisting of entity extraction, relation extraction, and extraction of modifiers (properties) of entities, such as laterality and negation. The system was built using the open-source GATE NLP framework, and the ML framework included with GATE. For entity extraction, one component used that was not part of the standard GATE distribution, the Termino term recognition engine. It is likely, however, that similar functionality could now be achieved through the use of ontology backed dictionary lookup that is distributed with more recent versions of GATE. Aside from importing and filtering terms in Termino, the work that was required to build an entity recogniser was either configuration of standard components, or small amounts of simple programming to marshal and rename annotations for presentational reasons. Providing features for extraction of entities by ML was straightforward, as all features used were present in the GATE data model as token-level annotations and attributes of those

annotations.

For relation extraction, programming work was required to provide features for ML. This was because the required features were mostly functions of the two arguments of the relation, and were not already present in the GATE data model. GATE does not have a standard model of relations that can be directly related to its relation learning component. Overall, the GATE framework and the standard components distributed with GATE constituted the major part of the final system. This does not mean, however, that configuring and using those components was an exercise solely in software engineering. The biggest piece of work in creating the final system was the selection of features for ML, and this required knowledge of computational linguistics.

In Section 1.4.2.3, we noted that Nadkarni et al. (2011) raised the question, is NLP software likely to become a commodity? They answer that current NLP toolkits are still oriented to the advanced programmer, rather than the commodity market. Our experience shows that this is currently true for a full scale clinical IE system, but that a system for entity extraction alone, could be created with no programming.

5.1.4 Lowering the barrier: separating and re-using knowledge

Our three objectives had the aim of separating and re-using linguistic, domain and engineering knowledge. We close this summary of achievements by considering each of these in turn, considering whether and how each objective has moved us towards this.

In an IE system, *linguistic knowledge* ranges from knowledge about parts-of-speech and their assignment, to the models or grammars required for a full syntactic parse. In a rule based system with semantic processing of a technical domain, there will often be no clear separation of the linguistic knowledge and the domain knowledge required to determine the semantics of the text. Understanding the semantics of the text require both linguistic and domain knowledge. The approach taken in this thesis – supervised ML trained with a corpus manually annotated with semantics by domain experts – has achieved a high degree of separation of linguistic knowledge. The bulk of linguistic knowledge was obtained from pre-existing software components. The domain experts demonstrated sufficient non-technical linguistic knowledge as users of the language, to enable them to annotate the required semantics of the text.

There are, however, some caveats to this. First, the reported results suggest that although good levels of agreement can be achieved by non-linguist domain experts, there may be a benefit to using both sets of skills. The use of non-linguist domain experts has been suggested by others (Scott et al., 2012; Chapman and Dowling, 2006, for example,), although some authors have made the point that computational linguistics input is also re-

quired (Xia and Yetisgen-Yildiz, 2012). Second, the syntactic software components used were designed for general language. We did not examine the use of part-of-speech taggers and other components trained specifically on clinical text. It may be that given the differences between the language of clinical text and general language, that such components would improve performance.

Pre-existing NLP and ML toolkits allowed some re-use of *software engineering knowledge*. Given the maturity of these frameworks, however, it is surprising that they do not have a first-class representation of relations in their data models [1].

Domain knowledge was modelled through the re-use of existing resources, and by manual annotation. In the medical domain, existing resources provide a rich source of knowledge that is accessible with little effort. The manual annotation effort, on the other hand. took more effort than any other part of system development.

The overall result of using pre-existing frameworks and supervised ML for semantic annotation, was that compared to a rule-based system, data and system were separated to a degree that is difficult to achieve with rule-based systems, software engineering costs were lowered, and that the costs of semantic grammar creation were removed. A new cost was introduced, that of gold standard creation. The effort required to manually annotate a gold standard corpus is a particular problem in the clinical domain, where expert time is rarely available.

5.2 Impact and further work

The work reported is part of a general trend towards the use of supervised ML for clinical IE, trained and evaluated with rigorously built manually annotated corpora. The methodology and metrics presented in this thesis have been discussed and cited widely, including South et al. (2009, 2010); Meystre et al. (2010a,b); Mayer et al. (2009); Irwin et al. (2009); Bada et al. (2012); Xia and Yetisgen-Yildiz (2012); Scott et al. (2012). Taking work such as the reported research forward, South et al. (2011) has carried out a systematic qualitative assessment of factors that might lead to bias, and methods to reduce annotator workload, in studies such as the one reported in Chapter 2, making a number of recommendations about tools and methodologies. The relation extraction component of the system has been cited as an example of supervised ML of relations, for example Demner-Fushman et al. (2009); Uzuner et al. (2010a). Hahn et al. (2012) uses the work to illustrate the difficulty of the relation extraction task. There has also been interest in the co-reference annotation, and this is cited by Zheng et al. (2011, 2012); Savova et al. (2011). The research has also attracted some interest outside of medical informatics, in

[1] The current GATE data model is over 15 years old.

the general medical literature, as an example of a current clinical IE system (Scott et al., 2010; Dalan, 2010).

The methodology has been applied in its entirety by the author and others to build a corpus of adverse drug events in case reports, for use in pharmacovigilance (Gurulingappa et al., 2012), and the schema and mapping to UMLS were used in the annotation of a biomedical journal abstract corpus, that was subsequently used for the discovery of a novel association between a gene and oral cancer (Johansson et al., 2012).

Work on the manual annotation of clinical text by domain experts is ongoing, and we are involved in a long-term project at the South London and Maudsley NHS Trust (SLAM), for the extraction of information from a large psychiatry Case Register (CRIS – Clinical Record Interactive Search, see Stewart et al. (2009)). This entails the extraction of quite specific entities, relations and events, in order to help answer particular research questions. Phenomenon extracted include smoking, medications, social care events, scores of cognitive ability, educational level, diagnosis, and negative symptoms of schizophrenia. The biggest problems encountered relate directly to the questions raised in this thesis. Annotation requires medical expertise, and has a high cost. Up-front annotation of the kind used in this thesis is not possible at SLAM. We are addressing this high cost through an agile annotation process. By defining simple rule-based systems, and getting domain experts to correct the output of these, we can illustrate what can be extracted to the domain experts, and use their corrections to help clarify extraction guidelines and improve system performance. Correction of annotations and improvement of extraction is carried out in an iterative manner. We plan to explore the use of corrected annotations collected from this iterative process in supervised ML. We have also trained medical end-users to build and evaluate prototype systems, with some success.

5.3 Future Work

The previous two sections have left some questions unanswered, and raised new questions. These form the basis of future work required in this area.

Components for NLP of clinical text are not yet widespread and re-useable, and the reported research made use of several lexico-syntactic components designed for general language. With the advent of publicly available corpora of clinical text, and of open source clinical IE systems, these are starting to become available (see for example Pakhomov et al. (2006)). More effort is required in this area. Work is also required on the NLP framework infrastructure, in particular the modelling of relations, and integration of this with ML frameworks.

Lexical knowledge and supervised ML can be successfully coupled, but **can rela-**

tional knowledge be coupled with supervised ML? Many medical ontologies contain information about valid relationships between concepts, such as taxonomic and partonomic relations, that might be of use in resolving ambiguities in text. Additional information is available in medical knowledge bases, for example, the symptoms of disease, and the indications for drugs. Re-use of this has been shown in rule-based systems (such as Ranum (1989)), how can we re-use this knowledge in supervised ML systems?

NLP frameworks need experts to drive them, and are not fit for use by non-experts. Is it possible to commoditise clinical IE, moving it out of the computational linguists laboratory, and into the hospital research department? What is required for this move, and what new problems will it bring?

The main cost of clinical IE is becoming **the cost of preparing manually annotated training data**. How can this be reduced? Active areas of research in this area in general NLP include active learning (Thompson et al., 1999; Ghani et al., 2003; Settles, 2010) and the use of large scale non-expert annotation through web-based voluntary and piece-rate payment services (Snow et al., 2008). In the latter, the hypothesis is that by combining the output of large numbers of cheap, non-expert annotators, performance equivalent to expert annotators can be achieved. Is this applicable to the medical domain?

Finally, the research raises the question of how best to **define requirements for clinical IE**. The supervised ML approach represented a shift from a complex set of templates that could not be operationalised because requirements were not clear. A schema existed in the form of template definitions, but there were few clear extraction tasks using these templates. In moving to a supervised ML approach, the focus became the schema alone. Once this was defined and sufficient examples had been collected, it was possible to build a clinical IE system for the extraction of a general set of clinical entities and relations, rather than a system focused on a well-defined extraction task. Although successful in the entity and relation task, the system played on a trade-off between generic re-usability, and task-specific utility with little re-usability. How requirements are crystallised, beyond the need to extract entities and relations, is an open question.

Appendices

Appendix A

Example narrative

Foreword

The corpus described in this thesis is not available for release at the time of writing. The letter reproduced below is a mock letter, written by clinicians working on the CLEF project in the style of real letters in the CLEF corpus. It was originally written to use in public demonstrations and training sessions.

Example narrative

23.09.1990.

This lady attended outpatients today. In 1984 she had a right simple mastectomy of a carcinoma of the breast and was commenced on Tamoxifen. There was no sign of tumour recurrence on follow up.

Her new symptoms are of lymphoedema in the right arm which has developed over the last six weeks. She has also complained of pain in the right hip. I note her recent FBC was normal.

I have taken the precaution of doing an X-ray of the pelvis and given her a tubigrip bandage to use for the lymphoedema in her arm. We plan to see her again in two weeks time with the result of the X-ray.

Appendix B

Annotation Guidelines, and accompanying CD

Foreword

The original CLEF annotation guidelines were written using a wiki, published as a hypertext (University of Sheffield, 2008), and made available as a web site to annotators. The web site version is included in full on the CD accompanying this thesis. A printed version of the guidelines, generated from the hypertext, is given in this appendix.

Although the guidelines may be read in a linear fashion, the original intention was that they be browsed by annotators as they worked, and with this in mind, they make heavy use of bullet points, numbered lists and tables. Several elements have been removed from the printed version given here, in order to improve readability as a linear document, and in order to remove administrative information that is not part of the guideline instructions. The elements removed are:

- Hyperlinked section contents at the head of each section.
- Lists of hyperlinks to related pages in many sections.
- Hyperlinks to summary tables in many sections.
- A "news" section for annotators, with information about changes from one version to the next.
- A revision log.

The guidelines include examples of text. None of these are taken from the CLEF corpus, although many were written in the style of similar examples found in the corpus.

Contents

B.1 Introduction

This document describes:

- Manual annotations added to the CLEF gold standard corpus, 2006 - 2007.
- Guidelines for creating those annotations
- Technical details of annotation formats and gold standard processing.

CLEF gold standard annotations mark semantic units in clinical texts: things such as diseases, drugs, body parts. The annotations also mark the relationships between these things. This document provides a common understanding of what defines an annotation. Information is provided for CLEF project members, and for any other users of the gold standard corpus. The description given is informal: a formal definition of annotations is also available if required.

B.2 Terminology

This section describes the terms used to discuss annotations. Although some annotators will be familiar with the CLEF project and its language, others may have no background in natural language processing, ontologies, or any of our other disciplines. In addition, CLEF project partners have their own terminologies for their own work. These often conflict, and can cause confusion. For the purposes of annotating the CLEF gold standard, a single terminology will be adopted.

(Technical note: as the annotations are primarily for use in an information extraction gold standard, the terminology will be based on that used in information extraction. Specifically, it will be based on the terminology that has evolved in the MUC, ACE, and TIMEX annotation exercises and evaluations.)

B.2.1 An example

The terminology used is described below. It makes use of the following example:

- ***"Mr. Jones has a melanoma. It is in his left second toe. There are no secondaries."***

B.2.2 Annotation terminology

Term	Description	Example
Entity	An entity is a thing in the world. It has an existence independent of the text: it is not a piece of text. It may be concrete or abstract. Explicit entites are mentioned in the text. Implicit entites are not mentioned, but their existence may be inferred from the text. We are not interested in annotating implied entities, only those that are explicitly mentioned in the text.	In the example, the piece of real-world tissue that is *Mr. Jones's melanoma* is an explicit entity. So is his left second toe. But the bit of flesh and bone that is *Mr. Jones's left foot* is an implied entity - it is not mentioned in the text, although we can guess that it exists. We are only interested in the things in the text: the mealnaoma and the toe, not the foot.
Mention	A mention is the textual realisation of an entity. A single explicit entity may have more than one mention. An implicit entity has no mentions.	In the example, the surface language strings "melanoma" and "it" are both mentions of the entity that is the real-world lump *Mr. Jones's melanoma*
Signal	A signal is a piece of text that provides extra information about an entity. It may modify it, providing a value for some attribute of the entity.	In the example, "left" signals something about the *toe*: its laterality attribute. Also, "no" signals something about the *secondaries* entity: that it does not exist.
Reference	Mentions refer to entities. They provide references to entities. The reference is the relation between the mention and the entity.	As with mention, "melanoma" and "it" both provide references to the entity that is *Mr. Jones's melanoma*
Co-reference	When two or more mentions refer to the same entity, they corefer.	"melanoma" and "it" corefer to the entity that is *Mr. Jones's melanoma*

Continued on next page

Continued from previous page

Term	Description	Example
Type	A type is a categorisation of an entity or signal. Each entity or signal will have a type.	In the example, we may have types of PERSON, DISEASE and BODY-PART. The entity that is *Mr. Jones's toe* has a type of BODY-PART. *Mr. Jones* has a type of PERSON.
Relationship	A relationship exists between two entites. It describes some interaction between those two entities in the world. Like entities, relationships have a type.	*Mr. Jones's melanoma* is located in his *toe*. We say that there is a relationship between the *melanoma* and *toe*. It could have a type of LOCATION.
Modifier relationship	A modifier relationship exists between every signal and the entity that it provides information about. The modifier relationship provides the entity with some attributional property. Typically, this will have a value selected from a limited set of possible values.	In the example, the *left* signal modifies the *toe* entity. It gives it an attribute with a vlaue, such as *laterality=left*.
Argument	The entities and signals that are related by a relationship are called its arguments.	In the previous example, the *melanoma* and *toe* entities are arguments to the LOCATION relationship.

B.3 The annotations: a summary

An annotation is a piece of information attached to some text, usually describing the text in some way. An annotation may be:

- attached to a particular region of a document, such as to a word or group of words.
- attached to the document as a whole, and independent of a particular span of text

In CLEF, we are interested in the sorts of entities found in clinical documents: drugs, body parts, diseases and so on. Annotating a document is the task of marking the mentions of these entities in a document, describing their type, and perhaps adding other annotations to the document to describe the relationships between these things. Typically, annotation software will display annotations by highlighting the text to which it is attached with some colour.

This section gives descriptions of the CLEF gold standard annotations for entities and relationships, and what things they refer to. It tells you:

- What CLEF annotations stand for
- How to make sense of annotations in a ready-annotated CLEF document
- What an annotation means if you find it attached to a piece of text in a CLEF document

It does **not** tell you:

- How to add annotations to an un-annotated document
- The detail of the mapping between surface text and either entities or relationships.
- It does not say how to decide whether a piece of surface text should have an annotation.

For these things, refer to the sections on annotating entites and annotating relationships in text.

B.4 Entities

B.4.1 Examples

The description of annotations in future sections make use of the following examples:

1. *"This patient has had a lymph node biopsy which shows melanoma in his right groin. Five out of ten nodes were involved. It is clearly secondaries from the melanoma on his right second toe. Although his PET scan is normal he does need a groin dissection. We will perform a CT scan to look at the left pelvic side wall and I will review him together with Dr. X next week."*

2. *"I have discussed her with x. We agreed to treat with DTIC, and then consider radiotherapy."*

3. *"This 56 year old woman was admitted to x ward on the date above, with increasing facial pain. This was initially relieved by co-codamol"*

4. *"There was no evidence of extra pelvic secondaries"*

B.4.2 The entities

The following table describes the CLEF entities. These are CLEF's basic semantic units in the text. They denote things in the real world of the patient's care, such as diseases, symptoms and drugs. For each entity type, an informal description and an example are given.

Entity type	**Description**	**Example**
Condition	Symptom, diagnosis, complication, conditions, problems, functions and processes, injury	In example 1, there are two *melanoma* entities of type condition: one in the right groin, and one in the right second toe. The *melanoma* in the right groin has two mentions: "melanoma" and "it".
Intervention	Action performed by doctor or other clinician targeted at a patient, Locus, or Condition with the objective of changing (the properties) of, or treating, a Condition.	In example 1, there is one intervention entity, the *dissection*. In example 2, the *radiotherapy* is an intervention.
Investigation	Interaction between doctor and patient or Locus aimed at measuring or studying, but not changing, some aspect of a Condition. Investigations have findings or interpretations, whereas Interventions usually do not.	In example 1, there are three investigation entities: a *biopsy*, a *PET scan* and a *CT scan*.
Result	The numeric or qualitative finding of an Investigation, excluding Condition	In example 1, the *PET scan* has a result of "normal". Other examples include the numeric values of tests, such as "80mg".
Drug or device	Usually a drug. Occasionaly, medical devices such as suture material and drains will also be mentioned in texts. These will also be annotated along with drugs.	Example 3 contains a single drug entity, *co-codamol*.

Continued on next page

Continued from previous page

Entity type	Description	Example
Locus	Anatomical structure or location, body substance, or physiologic function, typically the locus of a Condition.	There are five loci entities in example 1: *lymph node*, the right *groin*, *second toe*, the *groin* that needs a dissection, and *pelvic side wall*. It is debatable as to whether the two groins are the same entity, although you could perhaps infer that the dissection would be on the groin that contains the melanoma.

B.5 Signals

B.5.1 Examples

The description of annotations in future sections make use of the following examples:

1. *"This patient has had a lymph node biopsy which shows melanoma in his right groin. Five out of ten nodes were involved. It is clearly secondaries from the melanoma on his right second toe. Although his PET scan is normal he does need a groin dissection. We will perform a CT scan to look at the left pelvic side wall and I will review him together with Dr. X next week."*

2. *"I have discussed her with x. We agreed to treat with DTIC, and then consider radiotherapy."*

3. *"This 56 year old woman was admitted to x ward on the date above, with increasing facial pain. This was initially relieved by co-codamol"*

4. *"There was no evidence of extra pelvic secondaries"*

B.5.2 The signals

The following table describes the CLEF signals. Signals are pieces of text that provide some extra information about an entity, modifying it in some way. For each signal type, the entity type that it modifies is given, together with a brief description and examples.

Signal type	**Entity modified**	**Description**	**Example**
Negation	Condition	In general, things that are in the text are assumed to exist by the very nature of them being discussed. Sometimes, however, they do not: the text says that they are negative, or absent. In other cases, the text may say that something is unknown or uncertain. Negation signals cater for this, and are used to mark the part of the text that shows absence, negation and uncertainty. Negation signals may have values of *absent* and *uncertain*.	In example 4, the text "no evidence" signals the absence of the secondaries (more precisely, the absence of any finding of secondaries).
Laterality	Locus, Intervention	Text that signals the laterality of a Locus or Intervention. May have a value of *left*, *right*, *bilateral*.	There are three lateralities in example 1: two rights, and a left
Sub-location	Locus	Text that signals some division of, or extra information about, a Locus. Takes no specific values.	There is a sub-location in example 4: "extra" (as in external to) provides additional information about the locus "pelvis": that really, the text means the area outside of the pelvis.

B.6 Annotating co-reference in text

B.6.1 Introduction

- Co-reference is a phenomenon in language where two words refer to exactly the same thing in the world.

- For the purposes of discussion, we will distinguish between two types of co-reference:
  - Pronominal co-reference: a pronoun co-refers with something earlier in the text.
    - For example,
      - "He has a melanoma. It is in his second toe".
      - The pronoun "it" co-refers with "melanoma". They are exactly the same thing.
  - Lexical co-reference: a lexical item from an open word class, such as a noun, refers to something earlier in the text.
    - For example,
      - "He has a melanoma. The tumour is in his 2nd toe."
      - The noun "tumour" co-refers with "melanoma". They are exactly the same thing.
- For every mention of an entity in text, annotators should record all of its co--references.

B.6.2 Pronominal co-reference

For the purposes of pronominal co-reference, annotators should consider the following non-exhaustive list of pronouns that are commonly used when referring to entities in the CLEF texts:

Type of pronoun	**Examples**
Definite	it, they, them
Demonstrative	this, that, these, those
Interrogative	which, whose, what
Possesive	whose, their

B.6.3 Lexical co-reference

- Lexical co-reference is between two words for the same thing. Commonly, this coreference is between:
  - Synonyms
    - For example,

            - "Haemoglobin was 7.5g/dl. Given the Hb, further treatment was postponed until after transfusion"
            - "Haemoglobin" and "Hb" are synonymous. The two words will be co-referred.
    - A specific word and a more general form (hypernyms)
        - For example,
            - "There was a mass in his 2nd toe. The digit was excised."
            - "digit" is referring to the same thing as "toe", in a general sense. The two words will be co-referred

B.6.4 Using domain knowledge

- Co-reference should be annotated regardless of any domain knowledge needed to interpret the co-reference
- Co-reference that depends on an understanding of the domain will be annotated.
- Lexical co-reference in particular often requires more domain knowledge to understand. Such co-references are not always obvious to the non-expert, although they may be guessed from clues in the text.
- Here are some examples of lexical and pronominal co-reference to illustrate the use of domain knowledge:
    - "He has a melanoma. The tumour is in his 2nd toe."
        - implies a co-reference between melanoma and tumour
        - understanding the co-reference requires knowledge that a melanoma is a kind of tumour
        - the co-reference will be annotated
    - "He has a melanoma. It is in his 2nd toe."
        - implies a co-reference between melanoma and it.
        - understanding the co-reference requires no domain knowledge.
        - the co-reference will be annotated.
    - "X-ray showed a mass. It was excised."
        - there are two possible co-references:
            - "X-ray" and "it"

        - “mass” and “it”
        - The co-reference will be annotated.
    - “X-ray showed a mass in the left lobe. It was excised.”
        - there are three possible co-references.
        - “X-ray” and “it”
        - “mass” and “it”
        - “left lobe” and “it”
        - The co-reference will be annotated.

- Some entities are inherently co-referential. For example, a patient has one abdomen. Two abdomen mentions in the text will most likely refer to the same entity. The resolution of this coreference also requires domain knowledge. It will be marked.

- In other cases, co-reference depends on the meaning of the text. For example, two mentions of an x-ray in a text may or may not refer to the same investigation. The co-reference will be marked if it can be inferred.

B.6.5 Co-reference and conjunctions

Sometimes, a single word might refer back to several things in a previous sentence. Co-reference should not be annotated in this case.

- For example,
    - “He is suffering from mild headaches and from back pain. These are being treated with ibuprofen.”
    - The pronoun “these” is referring to “headaches and back pain”.
    - However, we do not mark a single “headaches and back pain” Condition in the document. We mark two Conditions.
    - We have no way to deal with a single co-reference to two things like this in the annotation tool at the moment
    - The co-reference should therefore not be created.

B.6.6 Co-reference and sets

Sometimes, a plural or a set of things (e.g. a patient’s limb) will be mentioned, and then a little later, a single member of that set (e.g. their left leg). The two should not be

coreferred. A single thing in the world is not the same as a set that contains that thing: your left leg is not the same as your four limbs.

- For example
  - "Her finger nails show onycholysis. The nail of the left index is bleeding from the bed"
  - In such cases, the set (finger "nails"), and the indivudual ("nail" of the left index), should be annotated as Loci.
  - They should not, however, be coreferred.
- Drugs and their classes give similar examples. For example,
  - "We will start empirical antibiotic therapy today. He will take Flucloxacillin and Metronidazole"
  - "antibiotic" should not be coreferred to either "Flucloxacillin" or "Metronidazole"
  - "antibiotic" is referring to both of them together, and possibly to other antibiotics as well. It is not the same as either one of them.

B.6.7 Be aware of relationships that are not co-reference

- Two entities in the text must be clearly referring to the same thing in the real world to be coreferring. The fact that one thing *implies* another or *causes* another is not enough for them to corefer. They must be the same.
  - For example
    - "An opacity was seen compatible with gallstones"
      - The opacity is on a film. The gallstones are in a body. They are not the same thing. They should not be coreferred
    - "A wall thickening consistent with acute colitis"
      - The wall thickening implies colitis, but on its own, it is not colitis. They should not be coreferred
- The fact that one thing is a *part* of another will not be marked as a coreference
  - For example
    - "He has a mass in his lung, in the left lobe"
    - "lobe" and "lung" will not be co-referred.

B.6.8 Scope of co-reference

- Co-reference will be annotated where the referents are in the same document.
- Co-referents may be in the same, or in different sentences

B.7 Relationships

B.7.1 Examples

The description of annotations in future sections make use of the following examples:

1. *"This patient has had a lymph node biopsy which shows melanoma in his right groin. Five out of ten nodes were involved. It is clearly secondaries from the melanoma on his right second toe. Although his PET scan is normal he does need a groin dissection. We will perform a CT scan to look at the left pelvic side wall and I will review him together with Dr. X next week."*
2. *"I have discussed her with x. We agreed to treat with DTIC, and then consider radiotherapy."*
3. *"This 56 year old woman was admitted to x ward on the date above, with increasing facial pain. This was initially relieved by co-codamol"*
4. *"There was no evidence of extra pelvic secondaries"*

B.7.2 The relationships

Relationships describe some interaction between entities. Like entities, relationships have a type. The CLEF relationships are given below. For each, its arguments are given. These are the entity or signals types that interact. Also a brief description and example.

Relationship type	**First argument type**	**Second argument type**	**Description**	**Example**
has_target	Investigation, Intervention	Locus	Relates an intervention or an invetigation to the bodily locus at which it is targetted.	There are several has_target relationships in example 1. *lymph node* is the target of the investigation *biopsy*, and *groin* is the target of the intervention *dissection*.
has_finding	Investigation	Condition, Result	Relates a condition to an investigation that demonstrated its presence, or a result to the investigation that produced that result.	In example 1, *melanoma* is a finding of the *biopsy*. *normal* is a finding of *PET scan*
has_indication	Drug or device, Intervention, Investigation	Condition	Relates a condition to a drug, intervention, or investigation that is targetted at that condition	In example 3, *co-codamol* is indicated by *pain* (which has two mentions, “pain” and “this”.
has_location	Condition	Locus	Relationship between a condition and a locus: describes the bodily location of a specific condition. has_location may also describe the location of malignant disease in lymph nodes, relating an involvement to a locus.	There are three has_location relationships in example 1, *melanoma* is located in *groin*, and a second *melanoma* entity is located in *second toe*. The involvement “5 out of 10” is located in the *lymph node* entity. In example 3, *pain* is located in *face*.

Continued on next page

Continued from previous page

Relationship type	**First argument type**	**Second argument type**	**Description**	**Example**
Modifies	Negation signal	Condition	Relates a condition to its negation or uncertainty about it	"no evidence" in example 4 is a negation of "secondaries"
Modifies	Laterality signal	Locus, Intervention	Relates a bodily locus or intervention to its sidedness: right, left, bilateral.	In example 2 there are several laterality modifiers. For instance, the *second toe* has a laterality of *right*. For an example of an intervention laterality modifier, consider "right thoracotomy"
Modifies	Sub-location signal	Locus	Relates a bodily locus to other information about the location: upper, lower, extra- etc.	In example 4 there is a sub-location modifier. The *pelvis* has a sub-location of *extra* (as in external to).

B.8 Relationships cross-reference

For reference, relationship types are given ordered by the type of first and second argument entity.

First argument type	Second argument type	Relationship type
Condition	Locus	has_location
Investigation	Condition	has_indication
Investigation	Locus	has_target
Investigation	Condition	has_finding
Investigation	Result	has_finding
Intervention	Condition	has_indication
Intervention	Locus	has_target
Drug or device	Condition	has_indication
Laterality	Locus	Modifies, laterality
Laterality	Intervention	Modifies, laterality
Sub-location	Locus	Modifies, sub-location
Negation	Condition	Modifies, negation

B.9 Annotating text: a recipe

B.9.1 Introduction

It would be possible to read a document from start to end, marking all annotations in order, as they are found. This has not, however, been found to give the most accurate results. A more methodical approach is useful. This is given below. In order for annotation to be consistent, all annotators should follow this.

Once experienced at the task, annotators may find it quicker to interleave these steps. This is understandable. Annotators who do this should afterwards go over the whole document following the recipe below, in case anything has been missed by their more ad-hoc approach.

(Technical note: this recipe is written with the assumption that annotators are using the Knowtator tool. The steps are chosen bearing this in mind)

B.9.2 Summary

Step	Sub-step
1. Read the document	
2. Mark the entities	
3. Mark the signals	
	3.1 add modifier relationships for each signal as you add the signal
4. Check for co-references	
	4.1 add any additional co-referring entities you might find, such as pronouns
5. Relationships etc.: for each entity in turn	
	5.1 Check entity spelling
	5.2 Check for relationships with other entities
6. Record additional information and time taken	

1. Read the whole document

Read the document through in its entirety, marking no annotations, to get an undestanding

2. Mark the entities

Read the document a second time, adding annotations for the mentions (including pronouns) of these basic entities, (in parallel if you find this easier):

- condition
- intervention
- investigation
- result
- drug
- locus

Certain entities may suggest that others also exist. You should bear in mind the following:

- If there is a condition, does it have a locus?
- If there is a drug or intervention, is it related to some condition?
- If there is an investigation, did it find some result or condition?
- If there is an investigation or intervention, was it targeted at a particular locus?

3. Mark the signals

Now go through each of the conditions, loci, and interventions, checking for modifiers, qualifications, and associated text that signify further annotations (in parallel if you find this easier):

- For all conditions, check for negation (and uncertainty). Annotate negation signals where they are found, and add a modifier relationship that relates the signal to the entity.
- For all loci, check for laterality and sub-locations. Annotate laterality and sub-location signals where they are found, and add a modifier relationship that relates the signal to the entity. Look for the stock phrases listed in the appropriate section: words such as “left”, “right”, “upper” etc.
- For all interventions, check for laterality. Annotate laterality signals where they are found, and add a modifier relationship that relates the signal to the entity. Look for the stock phrases listed in the appropriate section: words such as “left”, “right”, “bilateral” etc.

4. Co-reference

Now go through each of the mentions in turn, and check to see if it co-refers with any other mention. At the same time, check the text to see if you have missed any mentions that could be co-referred, in particular pronouns (things like “it”, “this”, “which”).

1. Create a co-reference annotation whenever you find two entities referring to the same thing, linking the coreference to the first mention of it.
2. Make sure that pronouns have also been marked as mentions and co-referred.
3. Add any additional entities that you spot, and co-refer them.

5. Relationships

Now go through each of the mentions in turn, and decide if any have relationships with other entities. At the same time, check the spelling of the mention.

Please take care not to over-annotate relationships. You do not need to hunt for every single possible relationship that you can deduce with clinical knowledge - only those that seem directly relevant to the section of text you are reading. Please see the general guidelines on annotating relationships for a discussion of this.

1. Is the mention or signals misspelt? If so, record it as such.

2. Consider the basic annotations and how they relate to others, adding relationships where they exist. In parallel:

    - Condition: does it have a locus?
    - Intervention: does it have any loci?
    - Intervention: was it indicated by any specific conditions?
    - Investigation: was it indicated by any specific conditions?
    - Investigation: is it targeted at any specific loci?
    - Investigation: did it find any specific conditions or results?
    - Drug or device: is it aimed at treating any particular conditions?

6. Recording additional information

As you are annotating the document, record any comments that you feel are important. You may have to do this in some text file, or perhaps in the annotation tool itself. For example, record:

- – Whether an annotation decision was hard to make
  – Any questions or uncertainty you may have about the annotations and guidelines
  – Anything unclear or ambiguous in the guidelines
  – Things that you consider clinically important, but which were not covered by the annotations allowed.
  – Annotation tool bugs and issues

B.10 Annotating text: general guidelines

B.10.1 Introduction

For our purpose, annotating text is the process of marking stretches, or spans, of text in some way, signifying that the span of text has particular semantics. Typically, an annotator will carry out this process with some tool. The tool will be used to associate annotations with bits of text, describing the semantics of those spans. In addition to annotations being associated directly with a span, other annotations may be added that describe relationships between bits of text. Annotation is about the text: what appears in it and what it means. It is not about building an abstract model of the text: it is grounded in the document itself.

In CLEF, the annotation process can be split into four sub-tasks:

- mark stretches of text as referring to entities, assigning them an entity type (such as locus)
- mark stretches of text as signalling something about an entity (such as the laterality of a locus)
- add other annotations to describe coreference links between those stretches of text that refer to the same entity
- add other annotations to describe the relationship between entities

These guidelines describe how annotators should map from the surface text to annotation:

- Which bits of text should be annotated?
- How should spans of text be mapped to entities and signals?
- When should annotations describing coreference and relationships be created?
- How should special cases be dealt with?
- What information should be recorded for a span of text?

This section gives some general guidelines for annotating text. This is followed by specific guidelines for each entity, signal, and relationship type.

B.10.2 Collaboration between annotators

- Annotators should not collaborate when marking up texts, unless explicitly requested to do so.
- A set of annotations should be the work of a single annotator only.
- There will be a designated phase of the annotation process for the discussion and resolution of differences.

B.10.3 Annotate words, not concepts

- Annotation is primarily about words.
- The presence or absence of the things in the world that those words refer to, is only of secondary importance.
- This means that words should be annotated even if the thing they are referring to does not really exist.
- Things that are in the future, are hypothesised, or even speculative, should still be annotated.
- For example,
  - "we will need to check his X-rays when he is admitted"
  - "x-ray" should be annotated as an investigation, even though it does not yet exist, and if circumsstances change, may never exist.

B.10.4 If something appears too complex to annotate, or you are unsure ...

- Then it probably is too complex to annotate, and should be left.
- There is no point in spending lots of time in philosophical knots about how something should be annotated.
- Annotation is not about trying to attach a label to every word.
- The guidelines can never cover every eventuality.
  - For example:
    - "he has difficulty clearing sputum"

    * If you are not sure which of these words is a conditon entity, then don't worry.

- This point is particularly pertinent to highly qualified problems and loci. In these cases, just annotate the main word or words.
  - For example,
    * "mild sudden onset bronchitis" - just annotate "bronchitis".
    * "multinodular goitre" - if you are not sure whether "multinodular" is important, ignore it and just annotate goitre.

B.10.5 Don't base annotation on your own view of what should be in a medical record

- For any annotators with an interest in medical information and records, this may be the hardest guideline to apply.
- Annotation is about finding those things that are listed in the guidelines.
- It is not about your own pre-conceptions.
- It is not about finding those things that you personally think should or should not be in a medical record
- Your own view of what should and should not be in a record may be very well founded and thought through, but it may be different to the view of the guidelines.
- We are interested in consistent annotation based on a single, written down set of instructions.
- We are not trying to collect annotations based on lots of different viewpoints, however expert they may be.
- Please try to justify every annotation against the guidelines.
- If you find yourself puzzling over whether something should or should not be annotated, and trying to squeeze something into the guidelines, then it is probably best not annotated.
  - You may find that it is a complex phrase - perhaps you can annotate just the core part of it. See the examples in the previous section.

B.10.6 Overlapping and containment of annotations

- Mentions cannot overlap with other mentions, or be contained within them
- Signals cannot overlap with other signals, or be contained within them
- Mentions and signals cannot overlap each other, or be contained within each other

B.10.7 Breaking down phrases

- Note: Loci, sub-locations, and laterality are often combined in a complex way. Please use theLocus and sub-location recipeto deal with these.
- A key question when annotating entity mentions, is: what is the textual extent of a mention? What does it include, and what does it exclude?
- For example, “mild left groin pain” can be annotated in many ways:
  - as “mild left groin pain”, a mention of a *pain* condition
  - as “left groin pain”, a mention of a *pain*, with “mild” left unannotated
  - as “left”, a laterality, and “groin pain”, a mention of a condition
  - as a laterality, a condition “groin pain” and a locus “groin”
  - as a laterality, a locus, and a condition
  - etc...
- The general rule will be to break phrases apart into their component entities. Modifiers that are not commonly treated as part of an entity will be ignored.
- So the above example will be annotated as a laterality, a locus, and a condition. The word “mild” will be ignored.
- There are, however, many medical terms that commonly include modifiers as part of the term.
- For example, “full blood count” could be annotated as:
  - a *count* intervention with locus *blood*, the modifier “full” left unnanotated
  - an intervention with mention “full blood count”
- In these cases, the term will not be split. It will be annotated as a single mention.

- The decision as to whether a term should be split is left to the judgement of annotators
- General tests that are suggestive of a term that should not be split are:
  - "Would the mention be found in a medical dictionary?"
  - "Is the mention something that has an acronym in wide use?"
  - Can the mention be rearranged syntactically, e.g. by switching words or by introducing a prepositional "of"?
    * If it can't then it may be a term (but not vice versa)
- Annotators should not attempt to assign annotations to every word in complex phrases. Words that are not clearly one of the required annotations can be safely ignored. If there is any doubt, do not annotate a word.
  - For example,
    * "moderately differentiated adenocarcinoma"
    * Only the word "adenocarcinoma" should be annotated
  - For example,
    * "partial nephrectomy" would not appear in a dictionary, though "nephrectomy" would.
    * Just "nephrectomy" should be annotated.
- For the purposes of the dictionary test, the final arbiter will be:
  1. Stedman's 27 edition. This is available online at `http://www.stedmans.com/section.cfm/45`
  2. For terms that may be British English specific, the UK CancerWEB Online Medical Dictionary. This is available online at `http://cancerweb.ncl.ac.uk/omd/`
  3. If you are unable to use an online dictionary, a paper one may be provided. (For example, if network connections are restricted for confidentiality reasons)
- Some examples:
  - "Full blood count" would be found
    * Also, "count of full blood" makes no sense
  - Myocardial infarction would be found

        * Also, swithcing words makes no sense without changing the syntactic category of the words (infarct of myocardium)
    - "mild left groin pain" would not
    - "left groin pain" would not
    - "groin pain" would not
    - "pain" would be found, as would "groin"

B.10.8 Implied entities

- Entities must have at least one mention.
- Only entities that are explicitly mentioned in the text should be annotated.
- Inference using domain knowledge should not be needed to create an entity. If an entity can only be inferred using domain knowledge, then that entity shall not be created.
- Every mention must refer to a piece of text.
- For example:
    - "Histology shows ..." implies that the patient had a biopsy
    - If the text nowhere mentions that the patient had a biopsy, then an intervention entity for this biopsy must not be created
- Conversely, entities must not be ignored, because although the annotator recognises an entity, they think that it is unimportant to the narrative. All entities that appear in the text should be annotated, whether or not the reader thinks they are clinically important.

B.10.9 Relationships and domain knowledge

- In many cases, relationships are explicitly stated in the text.
    - For example:
        * "Paracetamol was prescribed for his pain"
        * An has_indication relationship between "paracetamol" and "pain" must be annotated: it is clearly stated to exist in the text.
- There are lots of other common patterns that signify relationships:

  - "Problem in the Locus"
  - "Investigation showed Problem"
  - "Problem seen on the x-ray"
  - "Problem found on examination"

- Occasionally, however, some level of domain knowledge is required to infer that a relationship exists between two entities. These relationships will be annotated.
  - For example:
    * "He is in pain. Paracetamol was prescribed"
    * A has_indication relationship between "paracetamol" and "pain" exists
    * It requires (minimal) domain knowledge to infer this, but can also be guessed.
    * The relationship will be annotated.
  - For example:
    * "He is suffering from nausea and severe headaches. Dolasteron was prescribed"
    * There is a has_indication relationship between "dolasteron" and "nausea".
    * This is not obvious without domain knowledge. But with domain knowledge, it is quite clear.
    * The relationship will be annotated.

- Please try to only annotate those relationships that the text is telling you about. Often, such relationships are clearly stated. Sometimes, the text is saying something, but it needs some clinical knowledge interpret this and to decide on the relationship. This should not mean, however, that you try to deduce every single relationship between every single entity, regardless of whether the text is saying something about it. We are only interested in what the text is telling us.

- The guidelines on relationships should not imply that you should go on a hunt for tenuous and conjectured relationships holding between two entities mentioned several paragraphs apart. Relationships should be intentionally stated in the text, although in practice this might be hard to judge. If a relationship is not obvious with your clinical kowledge, please do not annotate it.

- Relationships should be annotated between the particular spans of text that seem relevant to your reading of a particular paragraph or section, i.e. spans of text that are "in the focus" of your attention.

B.10.10 Signals: modifying entities

Signals are additional words that *modify* an entity, to provide extra information about it. For eample, "_left_ leg", "_no_ meatstases", "_upper_ back". Signals always modify an entity that is closely associated with them. Signals are related to their main entity with a "modifies" relationship. So we might create annotations that say "left *modifies* leg". The *modifies* relationship is not like other relationships. It is saying something about the linguistic structure of a phrase, and is much less about clinical (domain) knowledge than other relationships.

- Signals are always in the same phrase as the word they are modifying. Never mark a signal as something modifying an entity in some other sentence or phrase.
  - For example,
    - "Fragmented core and blood clot together. Histological examination shows bone marrow with extensive necrosis"
    - Do not annotate "core" as a sublocation modifying "bone marrow". "Core" is not signalling anything about any word in its immediate phrase surroundings.
- Signals are almost always before the word they are modifying. Think hard before using a signal to modify a word in front of it.
  - For example,
    - "lower back": "lower" modifies "back"
    - "head of the pancreas": "head" should be marked as modifying "pancreas"
- Signals are often adjectives
- Every signal that is annotated should be related to at least one main entity. Signals should not be created that do not modify any entity.
  - For example, please do not mark every occurence of the word "no" as a negation signal. Only those examples that are clearly referring to some condition are signals about conditions.
  - For example,
    - "no referral has been made"
    - There is no need to mark "no" as a negation signal. It is not describing the absence of a condition.

- Signals may modify more than one main entity.
  - For example:
    - “no consolidation and collapse”
    - Mark “no” as negating both “consolidation” and “collapse”

B.10.11 Metonymy

- Metonymy is where a feature of something is used to stand for that thing.
- Entities and interventions that depend on metonymy will not be annotated.
- For example,
  - “we shall see him again in 6 weeks” implies an appointment.
  - An intervention for this appointment will not be annotated.

B.10.12 Cross-document inference

- All annotation will be of a single document in isolation, and should not consider other documents for the same patient. In particular, any inference required should only make use of information within the document being annotated.

B.10.13 Plurals, conjunctions and sets

- A single term may appear to refer to more than one entity. For example,
  - Two or more lesions: “lytic lesions in the spine and abdomen”
  - Two x-rays: “x-ray of the leg and chest”
  - One scan: “CT scan of her abdomen and thorax”
  - Two scans: “CT scan of her head and neck”
- In all of these cases, a single mention for a single entity will be annotated.
- Sometimes, a plural or a set of things will be mentioned, and then a little later, a single member of that set. For example,
  - “Her finger nails show onycholysis. The nail of the left index is bleeding from the bed”

  - In such cases, the set (finger "nails"), and the indivudual ("nail" of the left index), should be annotated as entities.
  - They should not, however, be coreferred - see Coreference

B.10.14 Spelling and other mistakes

- Misspellings should be annotated
- For example,
  - "lumbar punction"
  - is a misspelling of "lumbar puncture"
  - it should be annotated as mention of an investigation entity
- The mention should be recorded as a spelling mistake
- Other changes made after the letter has been dictated and typed, and that alter the way the letter reads, should also be marked as spelling mistakes.
  - For example, computer processing may introduce unintentional changes, such as:
    * "r******otherapy", where some process has obliterated part of a word
    * The word "r******otherapy" should be marked as a spelling mistake.
- Another mistake is where the typist misses a space between two words, running two different metnions together
  - For example,
    * "T1 G1 adenocarcinoma of the prostate. Presenting PSA8.5."
    * In this case, the space has been accidentally ommited between an investigation, "PSA", and its result, "8.5"
  - Where this has clearly happened, tha annotator should mark those parts of the non-spaced "word" that correspond to each mention type, regardless of the missing space, and additionally mark both as misspelt.
  - In the example, the characters "PSA" in "PSA8.5" would be marked as an Investigation, and the characters"8.5" marked as a Result. Both would be marked as misspelt.

B.11 Annotating entities in text

The basic annotation unit within the CLEF corpus is the *entity*. Entities refer to real-world objects that are a part of a patient's care and treatment: conditions, drugs, investigations etc. Entities are grounded in the text of CLEF documents. The span of text that refers to an entity is a *mention* of that entity. An entity may appear several times in the same document. Different mentions may refer to the same entity: "Mr. Jone's tumour... his melanoma... the lump".

Just as the entity is the basic unit of annotation, so marking up entities and mentions is the basic sub-task of the annotation process. In this sub-task, stretches of text are marked as being mentions of an entity of a particular type. A co-reference link may be created between these mentions. This section describes, for each of the entity types, how annotators should map from the surface text to annotation:

- Which bits of text should be annotated?
- How should spans of text be mapped to mentions: which text should be included and excluded?
- How should special cases be dealt with?
- What information should be recorded for different entities?

B.12 Condition

B.12.1 What is a condition?

B.12.1.1 Problems

Conditions are typically the sorts of things that some clinicians record in the "problem list" at the start of a patient document. Conditions include:

- symptoms
- diagnoses
- complications
- pathological functions
- processes
- injuries

B.12.1.2 Normal function

Conditions are not neccesarily pathological. For example, a piece of text may be commenting on some normal function.

- For example,
  - "bowel sounds were normal"
  - "sound" should be annotated as a condition.

B.12.1.3 Social and general life issues

Conditions may also include social and general life issues, that the writer has considered it important to mention

- For example,
  - Smoking
  - Drinking
  - Frail
  - Ederley

B.12.1.4 Psychological problems

Conditions may include psychological problems.

- For example,
  - "devastated by the results of surgery"
  - "devastated" should be annotated as a condition.
- For example,
  - "distressed by the bulge"
  - "distressed" should be annotated as a condition.

B.12.1.5 Physical and physiological processes

Conditions may also include processes. This is often associated with a locus.

- For example, here are several process conditions, each with an associated locus:

  - distended abdomen
  - swollen feet
  - painful joints

- For example,
  - "the mass is causing compression "
  - "the mass is compressing her trachea"
  - "his trachea is compressed"
  - In each of these, compression / compressing / compressed should be annotated as a condition.

- Cellular Processes such as "transformation" and "proliferation" should also be annotated as Conditions. See:Histopathology reports

B.12.1.6 General terms for problems and diseases

Sometimes, a patient's disease will be referred to as "the disease", or in phrases such as "no disease found". This will be annotated as a mention of the relevant condition entity

- For example,
  - "Mr. X has five to six in-transit deposits of melanoma around his groin. I have explained the nature of the disease to him."
  - "disease" will be annotated as a mention of the *melanoma* entity.

Similarly, a condition may be referred to using a general term such as "symptoms", "difficulties", "problems", "abnormality" etc. This will also be annotated.

- For example,
  - "No chest symptoms found"
  - "symptoms" will be marked as a condition, with locus "chest" and negation of "absent".

- For example,
  - "difficulty with breathing"
  - "difficulty" is a condition

- For example,

  - "The problem persists"
  - "problem" is a condition - probably coreferring with some other problem.

- For example,

  - "Sections show large bowel mucosa with no significant histological abnormality"
  - "abnormality" is a Condition, modified by the negation signal "no signficant"

B.12.1.7 Other people's conditions (e.g. a relative)

Even if a condition belongs to a person other than the patient, it should still be annotated. It is still a condition. Deciding who it belongs to is a separate process.

- For example,

  - "She attended outpatients today accompanied by her husband, who has CLL".
  - "CLL" should be annotated as a condition, even though it is not directly related to the patient.

B.12.1.8 Conditions as the findings of examinations

Examinations, as investigations (see below), often have lists of conditions that are related to the examination (by has_finding relationships: see below).

- For example

  - "On abdominal examination she had a scar consistent with the surgery, bruising over the abdominal wall. She had pitting oedema of both limbs."
  - There are three conditions, all related to the "examination" investigation.

B.12.2 What is not a condition?

B.12.2.1 Doubts and wonderings

General doubts and wonderings of the clinician will not be annotated as conditions.

- For example,

  - "My main concern was her femoral nerve neuropraxia"
  - "concern" will not be annotated as a condition

B.12.2.2 Progress, recurrence, change

Statements of the progress of a condition will not be annotated as standing for the condition.

- For example,
  - "She has noticed a change in her voice"
  - "change" will not be anotated as a condition.
- For example,
  - "I discussed the potential for recurrence"
  - "recurrence" will not be anotated as a condition.
- For example,
  - "He is free of tumour recurrence"
  - "recurrence" will not be anotated as a condition.

The loss or change of functional conditions should not be annotated.

- For example,
  - "She has lost strength in her adductors"
  - The loss of strength will not be annotated.

B.12.2.3 Results of an investigation

An investigation may find the absence of any condition. This will be marked as a result: see section on Result entity below.

- For example,
  - "Plain X-rays of these areas were all normal"
  - "normal" is a result, not a condition.

B.12.3 Conditions modified by other words: complex condition terms.

B.12.3.1 Conditions modified with loci

Conditions are often combined with a locus as a modifier

- For example,
  - Bony metastases
  - Cerebral atrophy
- An important questions in these examples is: should the whole phrase be annotated as a condition, or should it be split and annotated as a condition and a locus?
- The decision whether to split the phrase, or to leave as one, will follow the general guidelines on breaking down terms.
- Usually, one part of the phrase will be annotated as a locus, and one part as a condition. A relationship will be created between the locus and condition (see the has_location relationship)
  - For example,
    - "joint pain"
    - "pain" will be annotated as a condition, and "joint" as a locus
- However, where a condition is commonly modified by a locus, and the combined term is accepted as the name of a condition, it will be annotated in its entirety as a condition. The locus will not be annotated separately.
  - For example,
    - "myocardial infarction"
    - the entire term is valid as a condition entity
    - "myocardial" will not be annotated as a locus

B.12.3.2 Loci modified with conditions

As well as loci qualifying conditons, conditions may appear qualifying loci.

- For example,
  - "broken bone"
  - "fractured" will be annotated as a condition (a fracture), and "bone" as a locus

B.12.3.3 Other modifiers: detail of the condition

In additon to loci being added to condition words, many other words are combined to give more detail. These cannot and should not all be annotated, Where the modifier is commonly accepted as part of the condition name, it will retained and annotated with the condition. See the general discussion of entity annotation above, for further explanation of this guideline.

- For example,
  - "malignant melanoma"
  - the entire term is valid as a condition entity and might be found in a dictionary
  - "malignant" will be included in the annotation
- For example,
  - "multinodal goitre"
  - You would not find the entire term in a dictionary, and so only goitre will be annotated as a condition. "multinodal" will not be annotated.
- For example,
  - "moderately differentiated adenocarcinoma"
  - "adenocarcinoma" will be annotated as a condition, and the additional qualifying words ignored.

B.12.4 Multiple conditions appearing together

Some phrases give complex combinations of conditions that are associated with, or subparts of, each other. Each separate condition should be annotated. There is no need to combine them in any way.

- For example,
  - "a moderately differentiated adenocarcinoma with lymphatic invasion"
  - Two problems should be annotated: "adenocarsinoma" and "lymphatic invasion".

B.13 Intervention

B.13.1 What is an intervention?

B.13.1.1 General definition

Interventions are:

- Some technical act, such as an operation
- Some administrative act, such as an admission
- Some patient self-treatment, such as exercise

Interventions are usually intended to treat a condition, as opposed to investigations, which are usually aimed at diagnosing a condition.

- Note that this distinction is blurred and imprecise
  - For example, a diagnosis may be made as a result of some failed or succesful treatment. Or a procedure may both investigate and treat some condition (as in endoscopy).
  - For example, staging may involve both an intervention and an investigation.
- The distinction is, however, still felt useful for the purposes of annotation.
- Where the distinction is unclear, annotators should err on the side of annotating mentions as an "investigation"

B.13.1.2 Patient administration and movement

Patient administration and management events are interventions. Stock phrases include:

- admission
- discharge
- referral

B.13.1.3 Therapeutic acts

Therapeutic acts that do not involve drugs are considered to be interventions.

- For example, radiotherapy

B.13.1.4 Interventions as verbs

Interventions may often be expressed in verbal form. For example,

- "it was excised with clear margins"
- implies an excision intervention

Such interventions will be annotated. In the example, "excised" will be annotated as an intervention.

B.13.1.5 Patient self-treatment

Some interventions are patient self-treatments, such as exercise.

- For example,
  - "I have given him advice with regard to exercises to help with this"
  - "exercises" should be annotated as an intervention.

B.13.1.6 Advice given to patients

Advice given to a patient may also be considered an intervention.

- See the above example, where "advice" should be annotated as an intervention.

B.13.1.7 Operations and sub-procedures

Complex descriptions of interventions, such as operation notes, may mention sub-parts of a procedure. These will be annotated.

- For example, a single note may mention:
  - Bladder opened and a cuff of bladder removed. Rectum divided just above the peritoneal resection.
  - Three interventions will be annotated: "opened", "removed", and "divided"
- Such sub-processes include:
  - resection
  - division
  - closure
  - suture
  - and so on...

B.13.2 What is not an intervention?

B.13.2.1 Seeing a patient

The verb "see" can sometimes stand in for some unspecified intervention on patient and clinician. The verb "see" will not be annotated as an intervention.

- For example,
  - "I have asked by a consultant psychologist colleague to see this patient"
  - "see" will not be annotated as an intervention.

B.13.2.2 Care and treatment

General statements about care and treatment, as opposed to specific acts and events, will not be annotated as interventions

- For example,
  - "Thank you for involving us with her care"
  - "care" will not be annotated
- Examples:
  - "treated with atenolol", "his hypercalcaemia was treated"
  - "treated" will not be annotated
- Example:
  - "He presented in September"
  - "presented" will not be annotated

B.13.2.3 Section and paragraph headings

General statements of intervention in section and paragraph headings will not be annotated.

- Example:
  - "operation note and discharge summary:"
  - Neither operation nor discharge will be annotated

B.13.2.4 Changes to drugs

Starting and stopping of drug treatments will not be annotated as interventions.

- For example, the italicised words in the following will not be annotated as interventions:
  - "In 1987, she was c_commenced_ on Tamoxifen"
  - "Tamoxifen was _discontinued_"

B.13.3 Interventions modified by other words

Don't forget to use the dictionary test when considering interventions. Some interventions may be referred to by complex phrases, with additional words being added to the actual intervention to describe it further.

- For example,
  - "partial nephrectomy" would not appear in a dictionary, though "nephrectomy" would.
  - Just "nephrectomy" should be annotated.

B.14 Investigation

B.14.1 What is an investigation?

B.14.1.1 General definition

An investigation is some act intended to aid in diagnosis. It can include:

- Technical acts such as imaging and laboratory tests
- Acts carried out by the clinician, such as examinations

Investigations are usually aimed at diagnosing a condition, as opposed to interventions which are usually intended to treat a condition.

- Note that this distinction is blurred and imprecise
  - For example, a diagnosis may be made as a result of some failed or succesful treatment. Or a procedure may both investigate and treat some condition (as in endoscopy).

  - For example, staging may involve both an intervention and an investigation.

- The distinction is, however, still felt useful for the purposes of annotation.
- Where the distinction is unclear, annotators should err on the side of annotating mentions as an “investigation”

B.14.1.2 Examination

Examination is an especially important form of investigation, and should be annotated.

- For example,
  - “On examination, there was a fullness in the right supraclavicular fossa and a well healed scar.”
  - “Examination” should be annotated as an investigation.

B.14.2 What is not an investigation?

B.14.2.1 General, unspecific acts

General acts such as “identified” and “investigated” will not be annotated

B.14.2.2 Seeing a patient

The verb “see” often stands in for some unspecified appointment between patient and clinician. The verb “see” will not be annotated as an investigation.

B.14.2.3 Admistrative acts

Other general terms about appointments and hospital stays will not be annotated. For example, the following will not be annotated:

- – review
  - appointment
  - meeting / met

B.15 Result

- A result is the numeric or qualitative finding of an investigation, where that finding is not a condition.

- Results include:
  - the numeric values of tests (including units, if given)
  - references to normality and abnormality
  - qualities such as colours
- For example,
  - "potassium 3.8 mmol/l": "3.8 mmol/l" is a result of the potassium investigation
  - "INR of 1.9": "1.9" is a result of the "INR" investigation
  - "FBC was normal": "normal" is a result
  - "Chest x-ray showed abnormalities": "abnormalities" is a result
- Results and conditions are both findings of investigations. It is important to distinguish between them. A rule to apply is:
  - A condition can stand alone in its own right. It can be touched, excised, or discussed as a whole.
    - "Hb showed anaemia": we can talk about the patients anameia.
  - A result has no meaning away from the context of the investigation.
    - "Hb was 7.5": it makes no sense to talk about the 7.5 in isolation from Hb. The patient does not have a 7.5
- Some investigations are really banks of tests or give multiple parameters, such as an FBC.
  - Sometimes the entire investigation is reported with a single result.
    - For example, "FBC normal"
    - In this case, the "FBC" investigation should be linked to its "normal" result
  - On other occasions, however, the results of the individual parameters making up the test are reported
    - For example, "FBC. Hb 12.5, WBC 5.2, Plt 150"
    - In this case, each individual investigation (such as Hb, WBC, Plt) should be linked to their individual results

- An investigation may find the absence of any condition. This will be marked as a result.
  - For example,
    - "Plain X-rays of these areas were all normal"
    - "normal" is a result

B.15.1 Staging codes

Sometimes, cancer staging codes, such as TNM codes, are given. If you think these are the result of some investigation (such as "staging"), then please mark the code as a Result.

- For example,
  - "he had an oesophageal primary, staged locally as T4"
  - "T4" can be marked as a Result, being the finding of a "staging" Investigation

B.16 Drug or device

B.16.1 What is a drug?

B.16.1.1 General definition

- A drug is some medicine taken to reduce, cure or prevent some condition
- All generic and trade name drugs will be annotated
- Specific references to drugs used as nouns will be annotated

B.16.1.2 Classes of drugs

Where a class of drug is used to refer to a specific treatment, it will be annotated. For example,

- "responded to intravenous antibiotics"
- "antibiotics" will be annotated as a drug
- "Opioids were retitrated"
- "opioids" will be annotated
- "He has now completed his induction chemotherapy"

- "chemotherapy" will be annotated

Where a drug class is mentioned, and later individual drugs that are members of that class are also mentioned, you should be careful not to co-refer the class of drug with the individual drugs. See the guideline on Co-reference

B.16.1.3 "Treatment" and general words for drugs

The definite use of general words that clearly co-refer back to an earlier mentioned drug, will be annotated and co-referred.

- For example,
  - "We decided to try again with Tamoxifen. She is unlikely to respond to the treatment."
  - "Treatment" will be marked as a drug and co-referred to tamoxifen.

B.16.2 What is not a drug?

B.16.2.1 Drugs in headings

General references to "drugs", e.g. in headings, will not be annotated. For example, in:

- "Drugs on discharge:"
- The word "drugs" will not be annotated

B.16.2.2 Drugs as modifiers of conditions

Where a drug or class of drug is used as a modifier, it will not be annotated. For example,

- "There was a query about whether he had become opioid toxic"
- "opioid" will not be annotated (*opiate toxicity*, is however, a valid condition entity)

B.16.2.3 Dosage and route

It is not neccesary to annotate either the dosage or the route of administration: just the drug itself.

B.16.2.4 Body substances

The patients's own body substances should not be annotated as drugs or devices.

- For example,
  - "Masson Fontana stain for melanin is negative"
  - "melanin" should not be annotated as a drug.
  - There is no need to annotate it.

B.16.3 Chemotherapy

B.16.3.1 The word "chemotherapy"

Where the word "chemotherapy" is used to refer to a specific treatment using some broad class of drugs, it will be annotated. For example,

- "She is starting her induction chemotherapy on the LSA2L2 regime"
- "chemotherapy" will be annotated as a drug

B.16.3.2 Individual chemotherapy drugs

Where a specific drug is mentioned as part of chemotherapy, it should be annotated. For example,

- "He received 400mg carboplatin on chemotherapy"
- "carboplatin" should be annotated as a Drug.

B.16.3.3 Protocol names

Please do not annotate specific chemotherapy trial names and protocols as drugs: we do not want to collect them.

- For example,
  - "She is starting her induction chemotherapy on the LSA2L2 regime"
  - "LSA2L2" will not be annotated as a drug
- For example,
  - "We discussed the REAL trial which compares ECF and MCF chemotherapy"
  - "REAL", "ECF", and "MCF" will not be annotated as drugs.

B.16.4 Co-reference

Co-reference that requires a knowledge of pharmaceuticals will be annotated. For example, in:

- "I recommended he used a common NSAID, such as Ibuprofen"
- *NSAID* and *Ibuprofen* will be marked as separate entities, and co-refered.

B.16.5 Devices

A device is some manufactured object or substance used in the treatment or investigation of a patient. For example,

- suture material
- drains
- prostheses
- tubes

Medical devices will be annotated as drugs / devices.

- For example,
  - "closed with chromic catgut"
  - "chromic catgut" will be marked as a drug
  - "Robinson drain to pelvis"
    - "Robinson drain" will be marked as a drug

B.17 Locus

B.17.1 What is a locus?

B.17.1.1 General definition

- A locus may be:
  - an anatomical structure or location
  - a body substance
  - a physiologic function

  - within histopathology reports, this can include things at the tissue, cellular and sub-cellular level. See the notes on histopathology reports below.

- Loci, sub-locations, and laterality are often combined in a complex way. Please use the Locus and sub-location recipe to deal with these.

- All conditions should be checked, to see if a locus is mentioned.

B.17.1.2 Non-anatomical loci

It is important to note that loci are not just anatomical locations. Loci are the sites of conditions, and the targets of investigations and interventions. Loci can therefore include body substances, fluids, and functions.

- For example,

  - "cytology of her ascitic fluid"
  - "ascitic fluid" should be marked as a locus.

- For example,

  - "his hearing is impaired"
  - "hearing" should be marked as a locus.

- For example,

  - "there is some loss of voice"
  - "voice" should be marked as a locus.

B.17.1.3 Non-normative loci

Occasionally, a non-normative body structure may refer to a locus.

- For example,

  - "The chest wall flap caused her distress"
  - "chest wall flap" should be annotated as a locus.

B.17.2 What is not a locus?

B.17.2.1 Non-specific loci

- General and unspecified locations will not be annotated. These often occur when describing diffuse conditions.
  - For example,
    - "widespread metastatic disease"
    - "widespread" will not be annotated as a locus
    - "occult primaries"
    - "occult" will not be annotated as a locus
- Unspecified locations may also occur when describing vague areas. These will also not be annotated.
  - For example,
    - "we took a swab of the area"
    - "area" will not be annotated as a locus

B.17.2.2 General area words that refer to more than one locus

Loci that refer to a conjunction of several other areas will not be annotated.

- For example,
  - "She complained of pain in her upper back and right hip. X-rays of these areas were normal"
  - "areas" will not be annotated

B.17.2.3 Ordinal numbers and loci

Ordinals that modify a locus will not be included in that locus.

- For example,
  - "third toe"
  - "toe" will be annotated as a locus, not "third toe".

B.17.3 Loci modified by other words

B.17.3.1 Complex loci terms

1. Body locations are often used as modifiers of other loci.
    - For example,
        - "inguinal lymph node"
    - The decision whether to split the term into two loci, or to leave as one, will follow the general guidelines on breaking down terms, and will make use of the dictionary test.
2. Loci mentions that are formed by adding an adjective form of one locus to another locus, will be marked as a single locus in their entirety.
    - For example, all of the following will be annotated as a single locus:
        - "pelvic wall"
        - "axilliary tissue"
        - "abdominal aorta"
3. Sometimes, complex chains of locations are used to modify each other. The annotation needs to mark each separate component, using laterality and sub-location signals where appropriate.
    - For example,
        - Right lobe of the lower pole of the thyroid
        - Right upper quadrant of the abdomen
    - See the guidelines on Laterality and Sub-location for further detail.

B.17.3.2 A recipe for dealing with complex loci and sub-locations

It is often difficult to decide how a complex locus term should be annotated. This recipe should be applied in these cases, to decide which part of the term or phrase should be annotated as what entity and what signal. It applies guidelines that are discussed elsewhere in a fixed order.

Step	Description	Examples
1	Identify a phrase that is about some Locus	"Right upper quadrant of the abdomen"
		"surface of the large intestine"
2	Find the major anatomical site. What is the most general, or most whole, bit of the phrase	In "Right upper quadrant of the abdomen" this would be "abdomen".
		In "surface of the large intestine" this would be "intestine"
3	Does the dictionary test apply? Is there a multiword part of the phrase that could be considered a term in a dictionary? If so, this becomes our major anatomical site.	In "Right upper quadrant of the abdomen", there is none, so "abdomen" is the major anatomical site.
		In "surface of the large intestine", "large intestine" would be found in a medical dictionary, so is the major anatomical site.
4	Are adjacent words anatomical or general?	In "Right upper quadrant of the abdomen", none of the other words are anatomical terms, they are all general language.
		In "section of the inguinal lymph node", the word "inguinal" is another anatomical locus - and so joins "lymph node" to become part of the anatomical site
5	Mark the main locus	For "Right upper quadrant of the abdomen" this would be "abdomen".
		For "surface of the large intestine", this would be "large intestine".
		For "section of the inguinal lymph node", this would be "inguinal lymph node"

Continued on next page

Continued from previous page

Step	Description	Examples
6	Mark other general location words as a single sub-location, excluding laterality	For “Right upper quadrant of the abdomen”, this would be “upper quadrant”
7	Mark any laterality	For “Right upper quadrant of the abdomen”, this would be “right”

B.17.3.3 Loci modifying conditions

Where a Locus is used to modify a condition, they should be annotated separately - taking into account the general guideline “dicitionary test” (i.e. if it appears in a medical dicitionary, annotate it)

- For example,
  - “bony metastases”
  - “cerebral atrophy”

B.17.3.4 Loci modifying investigations and interventions

Where a Locus is used to modify investigations and inteventions, they should be annotated separately - taking into account the general guideline “dicitionary test” (i.e. if it appears in a medical dicitionary, annotate it)

- For example, “bone scan”

B.18 Annotating signals in text

A signal is a span of text that gives extra information about an entity. For example, the laterality of an entity. Once the mention of an entity has been annotated, other information signalled by the text can also be annotated and attached to the entity. This section describes how annotators should map from the surface text of signals to a signal annotation:

- Which bits of text should be annotated as signals?
- How should spans of text be mapped to signals?

- How should special cases be dealt with?
- What information should be recorded for a signal text?

B.19 Negation

B.19.1 What is negation?

B.19.1.1 General description

- Negation describes the absence, or uncertainty about the presence of, a condition entity.
- The purpose of the negation signal is to capture negation and uncertainty
- The starting point for negation is that we need to mark conditions that a patient does not have. This is partly so that the computerised annotation can avoid reporting them as conditions of the patient. If we were to report all conditions that appear in a document, often we would be in the embarrasing position of reporting all patients with "no evidence of melanoma" as having melanoma.
- It is therefore important not only to annotate conditions, but also to mark those that are mentioned because the patient does *not* have them.
- Absence is always explicitly stated.
  - For example
    - "There was no evidence of lymphadenopathy"
    - "no evidence" will be annotated as a negation signal with value *absent*.
- Negation will only ever be annotated when it relates to a condition.
- Note that we are not concerned that "absence of evidence is no evidence of absence". We are interested in capturing negative signals in the text, and will assume, for the purposes of annotation, that "no evidence" and other such signals are the same as absence.

B.19.1.2 For every negation, annotate a modifier relation

For every negation signal, at least one modifier relation must also be annotated, relating it to the associated condition.

B.19.1.3 For every negation, select a value

Each Negation signal that is annotated will be assigned a value from the enumeration: {*absent, uncertain*}.

- For example:
  - "There was no evidence of skin metastases"
  - The signal "no evidence" signals that "metastases" do not exist.
  - "no evidence" will be annotated as a negation entity with value of *absent*.
- For example:
  - "? METASTASIS"
  - The signal "?" signals that the existence of "metastases" is uncertain.
  - "?" will be annotated as a negation entity with value of *uncertain*.
- For example:
  - "The diagnosis of myelodysplasia is uncertain"
  - The signal "uncertain" signals that "myelodysplasia" might not exist.
  - "uncertain" will be annotated as a negation entity with value of *uncertain*.

B.19.1.4 Annotate the entire phrase

- Absence may be signalled by a single word, such as "no", but is often signalled by an entire phrase. The entire phrase that signals the absence should be marked.
  - For example,
    - "failed to indicate a definitive diagnosis of malignancy"
    - The "malignancy" is being negated by a phrase, "failed to indicate".
    - The entire phrase should be annotated.
  - For example,
    - "no evidence on this swallow of progressive disease."
    - the disease is being negated by the entire phrase "no evidence", which should be annotated.

B.19.1.5 Examples of stock phrases

Other stock phrases that indicate negation entities include:

- "free from"
- "absent"
- "no"
- "not present"
- "not seen"
- "uncertain"
- "unsure"
- "no evidence"
- "?" (as in query) - a negation with value of uncertain.

B.19.2 What is not negation?

B.19.2.1 Entities other than conditions are not negated

Only conditions will be negated. If a phrase describes the existence of something other than a condition, then it will not be annotated.

- – For example,
  - * "no evidence of surgery"
  - * "no evidence" will not be annotated as a negation entity

B.19.2.2 Conditions that are already expressed in the negative

Negation is not about marking conditions that are already negatives

- For example,
  - "afebrile"
  - "Afebrile" should not be marked as a negation. It is already a negative.

B.20 Laterality

B.20.1 What is a laterality?

B.20.1.1 General description

- Laterality is associated with either loci or intervention.
- It descibes the sidedness of the loci or intervention, in relation to the patient's body.
- Lateralities that modify other types of entity will not be annotated.

B.20.1.2 For every laterality, annotate a modifier relation

For every laterality signal, at least one modifier relation must also be annotated, relating it to the associated locus.

B.20.1.3 For every laterality, select a value

Each laterality signal that is annotated will be assigned a value from the enumeration: {*left, right, bilateral*}.

B.20.1.4 Examples of stock phrases

- There are a limited number of words that may be annotated as a laterality.
- These laterality stock words and phrases include:
  - right
  - left
  - rightmost
  - leftmost
  - bilateral
- Bilaterality may sometimes be indicated by other words implying involvement of all sides.
  - For example,
    - "Oedema of both limbs"
    - The "both" should be annotated as a laterality, related to the locus "limb".
- There may be others not included in this list.

B.20.2 What is not a laterality?

A laterality will only ever be a signal modifying either a locus or an intervention.

B.20.3 Lateralities in complex terms

- Loci, sub-locations, and laterality are often combined in a complex way. Please use the Locus and sub-location recipe to deal with these.
- Where either a locus or an intervention is qualified with its laterality, the laterality will always be split from the locus and annotated separately.
- Sometimes, complex combinations of loci and lateralities are used to modify each other. The annotation needs to mark the laterality as applying to the main locus, not the sub-locations.
  - For example,
    - Right lobe of the lower pole of the thyroid
      - "Right" should be annotated as a laterality of "thyroid"
    - Right upper quadrant of the abdomen
      - "Right" should be annotated as a laterality of "abdomen"
  - See the guideline on Sub-location for further details.

B.21 Sub-location

B.21.1 What is a sub-location?

B.21.1.1 General description

- A sub-location further divides and describes a locus.
- Loci, sub-locations, and laterality are often combined in a complex way. Please use the Locus and sub-location recipe to deal with these.
- A sub-location describes sub-areas, opposites, and negative areas.
- For example,
  - "lower back": "lower" modifies "back"
  - "extra pelvic disease": "extra" modifies "pelvic"

- Anatomical locations with no clear boundary will be annotated using this signal.
  - For example,
    - “the lower part of the right femur”
    - “lower part” will not be annotated as a sub-location related to “femur”.
  - For example,
    - “chest x-ray showed left lower lobe infection”
    - “lower lobe” refers to a sub-location, and will be annotated

B.21.1.2 For every sub-location, annotate a modifier relation

- For every sub-location signal, at least one modifier relation must also be annotated, relating it to the associated locus.

B.21.1.3 Examples of sub-location words and stock phrases

- Sub-location words are not always obvious “area” words. All sort of metaphors and analogies are used when describing location.
  - For example,
    - “head of the pancreas”
      - “head” should be marked as a sub-location
    - “Cardiomediastinal contour ”
      - “contour” is a sub-location
    - “lower pole of the thyroid”
      - “lower pole” is a sub-location
- Other sub-location words refer to the association of one structure with another.
  - For example,
    - “common iliac node”
    - “common” is referring to the location of the iliac node near to the common iliac vein, and should be annotated as a sub-location
- Other stock phrases that may provide sub-location signals include:
  - upper
  - outer

  - inner
  - top
  - bottom
  - high (e.g. high small bowel)
  - deep
  - superficial
  - lobe
  - quadrant
  - pole
  - contour
  - angle
  - common
  - medial
  - lateral
  - proximal
  - distal
  - midline
  - median

B.21.2 What is not a sub-location?

Sub-location should be used to annotate general language division and area words. It should not be used to annotate other anatomical terms that are qualifying a locus.

- For example,
  - “abdominal aorta”
  - “abdominal” is clearly an anatomical term.
  - It should not be annotated as a sub-location.
- For example
  - “hepatic parenchyma”
  - “hepatic” is clearly an anatomical term, as is “parenchyma”.

  - Neither should be annotated as a sub-location

- For example

  - "pulmonary parenchymal metastases"
  - Both "pulmonary" and "parenchymal" are anatmical terms
  - Neither should be annotated as a sub-location

- In these cases, the entire phrase should be annotated as a Locus, as described in the Locus guidelines.

B.21.3 Sub-locations in complex terms

- Loci, sub-locations, and laterality are often combined in a complex way. Please use the Locus and sub-location recipe to deal with these.

- Where more than a single word signals a sub-location, all words will be annotated.

  - For example, in "the upper part of his arm", "upper part" will be annotated
  - For example, in "the lower region of his abdomen", "lower region" will be annotated

- Sometimes, complex chains of locations are used to modify each other. The annotation needs to mark each separate component, using laterality and sub-location signals where appropriate.

  - For example,
    * Right lobe of the lower pole of the thyroid
      · "Right" should be annotated as a laterality of "thyroid"
      · "lobe of the lower pole" should be annotated as a sub-location of "thyroid"
    * Right upper quadrant of the abdomen
      · "Right" should be annotated as a laterality of "abdomen"
      · "upper quadrant" should be annotated as a sub-location of "abdomen"

B.22 Annotating relationships in text

The previous two sections looked at the annotation of entities and signals. These relate to specific spans of text: mentions of the entity, and signals that modify the entities. A further type of annotation describes how these latter two relate to each other. Which signal modifies which entity? Which drug is for which condition? These annotations are slightly different, in that they are not directly attached to spans of text. Instead, they describe how two spans of text interrelate. The guidelines in this section describes how to add these relational annotations:

- Which relationships can be used to relate which entities and signals?
- When should annotations describing relationships be created?
- How should special cases be dealt with?
- What information should be recorded for a relation?

B.23 has_target

B.23.1 Arguments

A has_target relationship associates either an intervention or an investigation to a locus.

First argument type	Second argument type	Relationship type
Investigation	Locus	has_target
Intervention	Locus	has_target

B.23.2 Entities do not have to have relationships

Interventions and investigations are not required to take part in a has_target relationship: some will have no locus specified:

- For example,
  - "He also had a PET scan."
  - No locus is ever mentioned for the PET scan, it being a whole-body scan. No has_target relationship will be created.
- For example,
  - "This 42 year old smoker presented with a severe cough and weight loss. An x-ray has been requested."

  - Although we might guess a locus of "chest" for the "x-ray", it is not mentioned in the text. No has_target relation will be created.

B.23.3 Inferring relations with clinical knowledge

If the locus of an intervention or investigations can only be inferred using clinical knowledge, then it will still be annotated

- For example,
  - "There was evidence of neurological involvement: she has become forgetful, and at times confused. A CT scan showed atrophy. where was no evidence of brain metastases."
  - We might infer that it was the scan that showed metastases, and therefore it was a scan of the brain.
  - a relation will be created associating "CT scan" and "brain"

B.23.4 has_target and multiple Loci

An intervention or investigation may have several loci. One has_target relation will be created for each locus.

- For example,
  - "A CT scan of her abdomen and thorax"
  - A has_target relationship will be created associating "CT scan" with "abdomen"
  - A second has_target relationship will be created associating "CT scan" with "thorax"
  - (Note that the general guideline on sets and conjunctions requires "CT scan" to be annotated as a single investigation)

B.24 has_finding

B.24.1 Arguments

A has_finding relationship associates an investigation with a condition or a result.

First argument type	Second argument type	Relationship type
Investigation	Condition	has_finding
Investigation	Result	has_finding

B.24.2 Entities do not have to have relationships

- Investigations are not required to take part in has_finding relationships.
- An investigation may have no finding or the finding may not yet be reported.
  - For example, a letter might state that "FBC, LFT and U&E were requested", but no mention ever be made of the results.

B.24.3 Inferring relations with clinical knowledge

A has_finding relationship may be created based on clinical knowledge:

- For example,
  - "FBC was requested, and showed severe anaemia."
  - We can infer that "anaemia" and "FBC" have a has_finding relationship without any specialist knowledge.
  - The has_finding relationship will be annotated.
- For example,
  - "FBC, U&E and LFTs were requested. She was severely anaemic."
  - With background knowledge, we know that the finding of "FBC" was "anaemic".
  - The has_finding relationship will be annotated.

B.24.4 has_finding and multiple Conditions and Results

An investigation may show several conditions and /or results. One has_finding relation will be created for each condition and result.

- For example,
  - "FBC showed thrombocytopaenia and neutropaenia"
  - A has_finding relationship will be created associating "FBC" with "thrombocytopaenia"

  - A second has_finding relationship will be created associating “FBC” with “neutropaenia”

- For example,
  - “platelet count of 20 showed severe thrombocytopaenia”
  - A has_finding relationship will be created associating “platelet count” with the result “20”.
  - A second has_finding relationship will be created associating “platelet count” with “thrombocytopaenia”

B.24.5 Examinations and has_finding

Examinations, as investigations, often have lists of conditions that are related to the examination by has_finding relationships.

- For example
  - “On abdominal examination she had a scar consistent with the surgery, bruising over the abdominal wall. She had pitting oedema of both limbs.”
  - The “examination” investigation is related to three conditions, by three has_finding relationships.

B.25 has_indication

B.25.1 Arguments

A has_indication relationship associates a intervention, drug or investigation to a condition that indicated the need for that entity.

First argument type	Second argument type	Relationship type
Investigation	Condition	has_indication
Intervention	Condition	has_indication
Drug or device	Condition	has_indication

B.25.2 Entities do not have to have relationships

Interventions, investigations and drugs are not required to take part in a has_indication relationship: some will have no condition specified.

- For example,
  - "Drugs on discharge: co-danthramer 10mls bd; Milpar 10-20mls bd ..."
  - Neither "co-danthramer" nor "Milpar" have an indication specified in this discharge summary.
  - If the whole discharge summary were read by a knowledgeable reader, indications could probably be inferred, but if they are not explicitly stated in the text, they will not be annotated.

B.25.3 Inferring relations with clinical knowledge

Indications that require clinical knowlege will be annotated.

- For example,
  - "Ibuprofen was prescribed for her pain."
  - There is an explicit association between "ibuprofen" and "pain"
  - A has_indication annotation will be created, relating "ibuprofen" and "pain".
- For example,
  - "He is in pain. Ibuprofen was prescribed."
  - Domain knowledge is required to infer that the prescription of "ibuprofen" and the "pain" are probably related, and not just coincidentally in adjacent sentences.
  - A has_indication relation will be annotated.

B.25.4 has_indication and multiple conditions

An intervention or drug may have several indications. One has_indication relationship will be created for each condition.

- For example,
  - "Dihydrocodeine and paracetamol were prescribed for the pain"
  - A has_indication relationship will be created associating "pain" with "dihydrocodeine"
  - A second has_indication relationship will be created associating "pain" with "paracetamol"

B.25.5 Investigations and has_indication

Investigations will not generally have an explicitly stated has_indication relationship. Occasionally, however, the indication for an investigation is stated in the text. Often, these are stated as prepositional phrases.

- For example,
  - "histology of the lung tumour showed..."
    - "histology" should be marked as an investigation with has_indication "tumour".
  - "Prostate cancer with normal PSA" * "PSA" should be marked as an investigation entity, with has_indication "cancer"

B.26 has_location

B.26.1 Arguments

A has_location relationship associates a condition to a locus.

First argument type	Second argument type	Relationship type
Condition	Locus	has_location

B.26.2 Entities do not have to have relationships

Conditions are not required to take part in has_location relationships.

A condition may have no locus specified.

- For example,
  - "He was forgetful and confused."
  - The locus of "Confusion" is not stated.
  - A relationship with the locus will not be annotated.

B.26.3 Inferring relations with clinical knowledge

A locus for a condition may only be implicit in the text, and require domain knowledge to infer. In this case, it will be annotated.

- For example,

  - "There was evidence of neurological involvement: she has become forgetful, and at times confused. A CT scan showed atrophy. There was no evidence of brain metastases."
  - We might infer that "atrophy" was located in the "brain".
  - The has_location relationship will be annotated.

B.26.4 has_location and multiple Loci

A condition may have several loci. One location has_relationship will be created for each locus.

- For example,
  - "pain in his left buttock and hip"
  - A has_location relationship will be created associating "pain" with "buttock"
  - A second has_location relationship will be created associating "pain" with "hip"
  - (Note that the general guideline on sets and conjunctions requires "pain" to be annotated as a single investigation)

B.27 Modifies: negation

- At least one negation modifier relationship will be created for every negation annotation.
- The relationship will associate the negation annotation with a condition annotation.
- A negation annotation may refer to several conditions. In this case, one relationship will be annotated for each condition annotation. For example,
  - "Crackles and wheezing were absent"
  - A relationship will be annotated to associate "crackles" with "absent"
  - A second relationship will be created to associate "wheezing" with "absent"

B.28 Modifies: laterality

- At least one laterality modifier relationship will be created for every laterality annotation.

- The relationship will associate the laterality annotation with a locus annotation.
- A laterality annotation may refer to several loci. In this case, one relationship will be annotated for each locus annotation. For example,
  - "pain in his left buttock and hip"
  - A relationship will be annotated to associate "buttock" with "left"
  - A second relationship will be created to associate "hip" with "left"
- A laterality may appear in front of both an intervention and a locus. It should modify the closest.
  - For example,
    - "right mastectomy of a breast carcinoma"
    - "right" should modify "mastectomy"
  - For example,
    - "right breast mastectomy"
    - "right" should modify "breast"

B.29 Modifies: sub-location

- At least one sub-location modifier relationship will be created for every sub-location annotation.
- The relationship will associate the sub-location annotation with a locus annotation.
- A sub-location annotation may refer to several loci. In this case, one relationship will be annotated for each locus annotation. For example,
  - "The cardiac and mediastinal contour are unchanged"
  - A relationship will be annotated to associate "mediastinal" with "contour"
  - A second relationship will be created to associate "cardiac" with "contour"

B.30 Histopatholgy reports

B.30.1 Introduction

In general, histopathology reports should be annotated in the same way as any other document. There are, however, some additional points to bear in mind. The way in which

some entities are used is slightly different, to allow for the annotation of the investigations, results, and loci (and some addtional conditions) that predominate in histopathology reports. This section details the differences.

B.30.2 Investigations

- Histopathology reports may contain large lists of histochemistry and immunocytochemistry tests.
- These should be annotated as investigations.
  - For example,
    - “Mucin stains are focally positive.”
    - “Mucin stains” should be annotated as an investigation
    - “Masson Fontana stain for melanin is negative”
    - “Masson Fontana stain” should be annotated as an investigation
- Many tests are looking for the presence of specific surface antigens and markers. It can be unclear whether the text is referring to the antigen or the test. Where this is the case, the text should be marked as the test - i.e. an investigation.
  - For example
    - “SMA and desmin and MNF116 are negative.”
    - SMA, desmin and MNF116 should all be marked as investigations.
- Sometimes, these markers will be reported as part of a general histochemistry test
- For example
  - “Immunohistochemistry shows these cells to be CD68 positive and MNF116 negative”
  - “Immunohistochemistry”, “CD68” and “MNF116” should all be annotated as investigations.

B.30.3 Results

- Histopathology reports may contain indications of the presence or absence of markers and stains.
- These should be annotated as results.

  - For example,
    * “Mucin stains are focally positive.”
    * “positive” should be annotated as a result
    * “Masson Fontana stain for melanin is negative”
    * “negative” should be annotated as a result

- In the case of surface antigen tests, the presence or absence of the specific surface antigens and markers should be annotated as results.
  - For example
    * “SMA and desmin and MNF116 are negative.”
    * “negative” should be marked as a result, linked to the three investigations.
  - For example
    * “Immunohistochemistry shows these cells to be CD68 positive and MNF116 negative”
    * “positve” and “negative” should be annotated as results, linked to “CD68” and “MNF116” investigations respectively.

- There are many other ways of expressing positivity, which should also be marked as results of their repsective tests.
  - For example
    * “P53 stains some large cells” - “stains” should be marked as a result of the Investigation “P53”
    * “large lymphoid cells immunoreactive for CD5” – “immunoreactive” should be marked as a result of the Investigation “CD5”

B.30.4 Locus

- In addition to the loci used in other reports, histopathology reports contain several other kinds of loci that should be annotated.

- Tissues
  - For example
    * “The subepithelial tissues are inflamed”
    * “subepithelial tissue” should be annotated as a locus
  - For example

        - "endometrial stromal sarcoma"
        - "endometrial stromal" should be annotated as a locus

- Cells
    - For example,
        - "HMB-45 immunostains are positive in these cells"
        - "cells" should be annotated as a locus
    - For example
        - "many plump fibroblasts were seen"
        - "fibroblasts" should be annotated as a locus
    - For example
        - "Mitotic figures are scanty"
        - "Mitotic figures" should be annotated as a locus

- Sub-cellular components
    - For example,
        - "The nucleoli are prominent"
        - "nucleoli" should be marked as a locus

B.30.5 Condition

- Histopathology reports sometimes mention cellular processes. These should be annotated as Conditions.
    - For example,
        - "There is a moderate proliferation fraction"
        - "proliferation" should be annotated as a condition
    - For example,
        - "consistent with transformation of previous B-CLL"
        - "transformation" should be annotated as a condition

B.30.6 has_finding relationship

- Histopathology reports may contain large lists of histochemistry and immunocytochemistry tests and results, indicating the presence or absence of markers and stains.

- These should be annotated as Investigations and Results, and related with a has_finding relation.
  - For example,
    - "Mucin stains are focally positive."
    - A has_finding relation should link "Mucin stains" and "positive"
    - "Masson Fontana stain for melanin is negative"
    - A has_finding relation should link "Masson Fontana stain" and "negative"
- In the case of surface antigen tests, the presence or absence of the specific surface antigens and markers should be linked to results.
  - For example
    - "SMA and desmin and MNF116 are negative."
    - "negative" should be linked to all three investigations, with has_finding relationships.
  - For example
    - "Immunohistochemistry shows these cells to be CD68 positive and MNF116 negative"
    - "positve" and "negative" should be linked with has_finding relationships to "CD68" and "MNF116" investigations respectively.

B.31 Radiology reports

B.31.1 Introduction

In general, radiology reports should be annotated in the same way as any other document. There are, however, some additional points to bear in mind. This section details these points.

B.31.2 Results

- Radiology reports may contain apparently free-standing results, which are the result of the entire procedure. These should be annotated as results.
  - For example,
    - "The cardiomediastinal contour is normal."
    - "normal" should be annotated as a result.

B.31.3 Drug or device

- Radiology reports may contain more references to devices
- These may be some manufactured object, or some substance used for the investigation.
  - For example,
    - “very little barium was seen to actually pass through the prosthetic tube”
    - Both “barium” and “prosthetic tube” should be annotated as drugs / devices

Appendix C

Consensus Guidelines

Foreword

This appendix reproduces the CLEF consensus annotation guidelines. The CLEF gold standard was first doubly annotated by two independent annotators. Any differences between these two sets of independent annotations were then resolved by a third *consensus* annotator, following the guidelines given below.

In several places, the guidelines make reference to the Knowtator plugin for Protégé (Ogren, 2006). This was the tool used by annotators, and the consensus guidelines are written with this in mind. They contain a mixture of instructions for how to make decisions on double annotation differences, and instructions for how annotators should implement these decisions in Knowtator.

The guidelines include examples of text. None of these are taken from the CLEF corpus, although many were written in the style of similar examples found in the corpus. In examples, the convention used to show entities is to surround them in square brackets, with the entity type immediately after the closing bracket. For example, "on x-ray, a tumour in the [right lobe]locus was visible", means that the actual text is "on x-ray, a tumour in the right lobe was visible", and that it contains a mention of a locus type entity, and that the mention is "right lobe".

C.1 Building a consensus annotation set

C.1.1 Using Knowtator for consensus annotating

In Knowtator, merged annotations will be highlighted in light blue. Unmerged will remain black on white. The aim of the consensus review is to check and resolve all unmerged annotations, leaving those highlighted in blue in place. Several actions can be performed

in Knowtator, to aid with this:

- Remove an annotation and consolidate with some other
  - On removing an annotation, the annotator will be given the chance to consolidate it with some other annotation. This should be done where there are two similar annotations, and one is being kept. Consolidation means that wherever the removed annotation appeared as a slot filler, the consolidation choice will be used in its place. The consolidation choice will be given an annotator the "consensus annotation team" and be highlighted in blue
- Remove an annotation with no consolidation
  - When given the chance to consolidate, press cancel. This should be done when deleting an annotation that exists in one set only, and where it has been decided that it should not be in the consensus set.
- Alter the annotator of a single annotation to the "consensus annotation team". This will highlight it in blue.
  - This should be done when deciding to keep an annotation that only appears in one set
- Delete and add relationships
- Change the type of an entity

C.1.2 Workflow

There is no big need for a strict workflow. Knowtator presents work completed on a blue background, and all work remaining as annotations listed on a white background. It is therefore easy to see what is left to do. Neither is order of consolidation important, as the consequences of each action are visible in Kowtator. It can, however, be a bit daunting to be presented with lots of complex decisions. It may be best to do the easy ones first.

A useful technique is to first go through the document merging and deleting where the decision is straightforward. Harder decisions often involve altering slot fillers, some of which might later be deleted or changed as part of some other decision. By resolving easy decisions first, slot fillers are taken out of play before we move on to these harder decisions. This technique has a useful side-effect of familiarising the reviewer with the document.

This is summarised in the following workflow, which should be used with the table of decision cases in the next section.

1. Go through all disagreements
2. Check for annotations that are of the same type and span, and that have slot fillers that are also of the same type and span. There may be several of these - Knowtator does not always succesfully merge annotations that may seem obvious (this seems to be when a chain of relations does not match).
3. Check all annotations where one annotator has marked a span but the other has not marked either that span or any overlapping span. The decision here is fairly easy: is the annotation correct? Is the typing correct?
4. Check all others, which will be harder cases involving relations.

C.1.3 Consensus decision cases

This table lists different cases that a reviewer may come across, describes the actions that may be taken, and gives examples where neccesary. Note that not all decisions are mutually exclusive: especially when dealing with relationships, more than one decision needs to be taken. Also, one decision may lead to the need for others. These cases should cover the majority of annotation differences, but there could be others. Please let Angus know if you find something that is not covered, or where the possible decisions described seem to fit uneasily.

	Decision case	**Possible actions**	**Description, examples**
1	Annotations merged by Knowtator	Nothing should be done	The two annotators were in agreement. The annotation will be highlighted in blue.
2	Span of text not annotated	Nothing should be done	The two annotators were in agreement, that the text should not be annotated (unless applying case **8**)

Continued on next page

Continued from previous page

	Decision case	**Possible actions**	**Description, examples**
3	Two annotations in exact agreement in class, span, and all slots	One should be deleted, and consolidated with the other.	Knowtator will not merge annotations where this is the case, if the slot fillers do not themselves have the same slot fillers. However, chains of relations like this will eventually get resolved through the complete review process.
4	An annotation exists in one set and not another	1. Delete, not consolodating with any other 2. Keep, changing the annotator to the consensus team annotator 3. Keep, changing the type and changing the annotator to the consensus team annotator	There is a simple choice: was one annotator right? In this case, you may, however, change the type if keeping. If you do this, the annotation will still be double annotated.

Continued on next page

Continued from previous page

	Decision case	Possible actions	Description, examples
5	Two annotations of the same span but different types (see also **6** and **7**)	1. Delete the annotation with the first type, consolidate into the second 2. Delete the annotation with the second type, consolidate into the first 3. Change the type of one annotation, and delete the other, consolidating into the changed one 4. Delete both annotations	The decision is: was one of the types wrong, or does the disagreement reflect that there should be no annotation here, or did they both create an annotation in the correct place but both mis-type? For example, Annotator 1: "A [CT scan]investigation has been scheduled for Tuesday". Annotator 2: "A [CT scan]intervention has been scheduled for Tuesday". The reviewer may choose to use: invetigation, intervention, some other type,remove the annotation.
6	Where an annotator removes an entity annotation in step **5**, they should first look at all relationships in the entity being removed.	1. Relationships in both the annotation being removed, and the one being kept, should be kept. 2. Other relationships in the annotation being removed may be added to the annotation being kept.	

Continued on next page

Continued from previous page

	Decision case	**Possible actions**	**Description, examples**
7	Where an annotator changes an entity type in step 5, they should be aware that this will also alter the relationships that it may take part in.	A choice must be made as to whether any relationships in the original annotation type can be added to the new type.	
8	Overlaps	The annotator may choose either span, or some other	For example, Annotator 1: "superficial inguinal [lymph node]locus" Annotator 2: "[superficial inguinal lymph node]locus" The reviewer may choose: "superficial inguinal [lymph node]locus" or "[superficial inguinal lymph node]locus" or "superficial [inguinal lymph node]locus" etc.

Continued on next page

Continued from previous page

	Decision case	**Possible actions**	**Description, examples**
9	Deleting one of two entity annotations with different relationships	1. All relationships that are in both entities must be kept 2. A relationships that is in one entity but not the other may be kept or removed 3. A relationships that is in neither entity may not be added	Often, the easiest action is to keep the entity with the most relationships, and add to and delete from this.
10	Same two entities related via different relationships	1. Keep the first relationship, and remove the second 2. Keep the second relationship, and remove the first 3. Remove both relationships	This is unlikely to happen with our annotation schema.

Bibliography

S. Abney. *Semisupervised Learning for Computational Linguistics*. Chapman & Hall/CRC, 1st edition, 2007. ISBN 1584885599, 9781584885597.

C. C. Aggarwal and C. Zhai, editors. *Biomedical Text Mining: A Survey of Recent Progress.* Springer, 2012. ISBN 978-1-4419-8462-3.

C. B. Ahlers, M. Fiszman, D. Demner-Fushman, F.-M. Lang, and T. C. Rindflesch. Extracting semantic predications from medline citations for pharmacogenomics. In *Pac Symp Biocomput*, pages 209–220, Hawaii, USA, January 2007.

B. Alex, C. Grover, B. Haddow, M. Kabadjov, E. Klein, M. Matthews, S. Roebuck, R. Tobin, and X. Wang. The ITI TXM Corpora: Tissue Expressions and Protein-Protein Interactions. In *Proceedings of Building and evaluating resources for biomedical text mining: Workshop at Sixth International Conference on Language Resources and Evaluation, LREC 2008*, pages 11–18, Marrakech, Morocco, 2008.

S. Ananiadou and J. Mcnaught, editors. *Text Mining for Biology And Biomedicine*. Artech House, Inc., 2005.

S. Ananiadou and G. Nenadic. Automatic terminology management in biomedicine. In S. Ananiadou and J. McNaught, editors, *Text Mining for Biology and Biomedicine*, chapter 4, pages 67–97. Artech House, 2006.

Apache OpenNLP Development Community. *Apache OpenNLP Developer Documentation*. The Apache Software Foundation, 1.5.2-incubating edition, 2011. `http://opennlp.apache.org/documentation/1.5.2-incubating/manual/opennlp.html` [Accessed 21 October 2012].

Apache UIMA. Welcome to the apache uima project. The Apache Software Foundation. `http://uima.apache.org/`, 2012. [Accessed 20 October 2012].

A. Aronson. Filtering the UMLS metathesaurus for MetaMap. Technical report, U.S National Library of Medicine, Lister Hill National Center for Biomedical Communications, Cognitive Science Branch, 2005.

A. R. Aronson. Metamap technical notes. Technical report, United States National Library of Medicine, Bethesda, MD, February 1996.

A. R. Aronson. Effective mapping of biomedical text to the UMLS metathesaurus: The metamap program. In S. Bakken, editor, *Proceedings of the 2001 AMIA Annual Symposium*, pages 17–21, Bethesda, MD, 2001. American Medical Informatics Association.

M. Bada, M. Eckert, D. Evans, K. Garcia, K. Shipley, D. Sitnikov, W. A. Baumgartner, K. B. Cohen, K. Verspoor, J. A. Blake, and L. E. Hunter. Concept annotation in the CRAFT corpus. *BMC Bioinformatics*, 13:161, 2012.

R. H. Baud, A. M. Rassinoux, and J. R. Scherrer. Natural language processing and semantical representation of medical texts. *Methods Inf Med*, 31(2):117–125, Jun 1992.

BioCreAtIvE. BioCreAtIvE II - protein-protein interaction task. SourceForge. `http://biocreative.sourceforge.net/biocreative_2_ppi.html`, 2006. [Accessed 29 October 2012].

BioNLP mailing list. Trends in clinical nlp. BioNLP. `https://lists.ccs.neu.edu/pipermail/bionlp/2012-October/002912.html`, 2012. [Accessed 20 October 2012].

BioNotate. BioNotate. SourceForge. `http://sourceforge.net/projects/bionotate/`, 2008. [Accessed 2 October 2008].

S. Bird, E. Klein, and E. Loper. *Natural Language Processing with Python – Analyzing Text with the Natural Language Toolkit*. O'Reilly Media, Sebastapol, CA, June 2009.

C. Blaschke, M. A. Andrade, C. Ouzonis, and A. Valencia. Automatic extraction of biological information from scientific text: protein-protein interactions. In *Proc Int COnf Intell Syst Mol Biol*, pages 60–67, Heidelberg, Germany, August 1999.

S. E. Bleeker, G. Derksen-Lubsen, A. M. van Ginneken, J. van der Lei, and H. A. Moll. Structured data entry for narrative data in a broad specialty: patient history and physical examination in pediatrics. *BMC Med Inform Decis Mak*, 6:29, 2006.

S. Boisen, M. R. Crystal, R. Schwartz, R. Stone, and R. Weischedel. Annotating resources for information extraction. In *Proceedings of the Second Language Resources and Evaluation, LREC 2000*, pages 1211–1214, 2000.

K. Bontcheva, V. Tablan, D. Maynard, and H. Cunningham. Evolving GATE to Meet New Challenges in Language Engineering. *Natural Language Engineering*, 10(3/4): 349—373, 2004.

M. Bundschus, M. Dejori, M. Stetter, V. Tresp, and H.-P. Kriegel. Extraction of semantic biomedical relations from text using conditional random fields. *BMC Bioinformatics*, 9(207), 2008.

R. Bunescu, R. Ge, R. J. Kate, E. M. Marcotte, R. J. Mooney, A. K. Ramani, and Y. W. Wong. Comparative experiments on learning information extractors for proteins and their interactions. *Artif Intell in Med*, 33:139–155, 2005.

R. C. Bunescu and R. J. Mooney. A shortest path dependency kernel for relation extraction. In *Proceedings of the Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 724–731, October 2005.

B. Carpenter and B. Baldwin. *Natural Language Processing with LingPipe 4*. LingPipe Publishing, New York, 0.5 edition, June 2011. `http://alias-i.com/lingpipe-book/index.html` [Accessed 21 October 2012].

J. Carvel. NHS risks £20bn white elephant, say auditors. The Guardian. `http://www.guardian.co.uk/technology/2006/jun/16/egovernment.politics`, 2006. [Accessed 29 October 2012].

W. W. Chapman and J. N. Dowling. Inductive creation of an annotation schema for manually indexing clinical conditions from emergency department reports. *J Biomed Inform*, 39(2):196–208, Apr 2006.

W. W. Chapman, W. Bridewell, P. Hanbury, G. F. Cooper, and B. G. Buchanan. A simple algorithm for identifying negated findings and diseases in discharge summaries. *J Biomed Inform*, 34(5):301–310, 2001.

W. W. Chapman, J. N. Dowling, and G. Hripcsak. Evaluation of training with an annotation schema for manual annotation of clinical conditions from emergency department reports. *Int J Med Inform*, 77(2):107–113, Feb 2008.

W. W. Chapman, P. M. Nadkarni, L. Hirschman, L. W. D'Avolio, G. K. Savova, and O. Uzuner. Overcoming barriers to NLP for clinical text: the role of shared tasks and the need for additional creative solutions. *J Am Med Inform Assoc*, 18(5):540–543, 2011.

W. W. Chapman, G. K. Savova, J. Zheng, M. Tharp, and R. Crowley. Anaphoric reference in clinical reports: characteristics of an annotated corpus. *J Biomed Inform*, 45(3):507–521, Jun 2012.

N. Chinchor. Overview of MUC-7/MET-2. In *Proceedings of the Seventh Message Understanding Conference (MUC-7)*, Fairfax, VA, USA, April 1998.

N. Chinchor and E. Marsh. MUC-7 Information Extraction Task Definition (version 5.1). In *Proceedings of the Seventh Message Understanding Conference (MUC-7)*, Fairfax, Virginia, April 1998. National Institute of Standards and Technology.

L. M. Christensen, P. J. Haug, and M. Fiszman. Mplus: a probabilistic medical language understanding system. In *Proceedings of the ACL-02 workshop on Natural language processing in the biomedical domain - Volume 3*, BioMed '02, pages 29–36, Stroudsburg, PA, USA, 2002. Association for Computational Linguistics. doi: 10.3115/1118149.1118154.

A. Coden, G. Savova, I. Sominsky, M. Tanenblatt, J. Masanz, K. Schuler, J. Cooper, W. Guan, and P. C. de Groen. Automatically extracting cancer disease characteristics from pathology reports into a Disease Knowledge Representation Model. *J Biomed Inform*, 42(5):937–949, Oct 2009.

K. B. Cohen, P. V. Ogren, L. Fox, and L. Hunter. Corpus design for biomedical natural language processing. In *Proceedings of the ACL-ISMB Workshop on Linking Biological Literature, Ontologies and Databases: Mining Biological Semantics*, ISMB '05, pages 38–45, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics. URL `http://dl.acm.org/citation.cfm?id=1641484.1641490`.

J. Cowie and W. Lehnert. Information extraction. *Communications of the ACM special issue on Natural Language Processing*, 39(1):80–91, January 1996.

R. Crowley. Progress on the ODIE Toolkit. The National Center For Biomedical Ontology. `http://www.bioontology.org/odie-update`, 2010. [Accessed 23 October 2012].

R. S. Crowley, M. Castine, K. Mitchell, G. Chavan, T. McSherry, and M. Feldman. caTIES: a grid based system for coding and retrieval of surgical pathology reports and tissue specimens in support of translational research. *J Am Med Inform Assoc*, 17(3): 253–264, 2010.

H. Cunningham. Information Extraction, Automatic. *Encyclopedia of Language and Linguistics, 2nd Edition*, pages 665–677, 2005.

H. Cunningham. SAFE, the Semantic Annotation Factory Environment. University of Sheffield. http://gate.ac.uk/safe/, 2008. [Accessed 2 October 2008].

H. Cunningham, K. Humphreys, R. Gaizauskas, and Y. Wilks. Software Infrastructure for Natural Language Processing. In *Proceedings of the 5th Conference on Applied Natural Language Processing (ANLP-97)*, Mar. 1997. ftp://ftp.dcs.shef.ac.uk/home/hamish/auto_papers/Cun97a.ps.gz.

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*, pages 168–175, Philadelphia, PA, USA, 2002.

H. Cunningham, D. Maynard, K. Bontcheva, V. Tablan, N. Aswani, I. Roberts, G. Gorrell, A. Funk, A. Roberts, D. Damljanovic, T. Heitz, M. A. Greenwood, H. Saggion, J. Petrak, Y. Li, and W. Peters. *Text Processing with GATE (Version 6)*. University of Sheffield Department of Computer Science, 2011. ISBN 978-0956599315.

D. Dalan. Clinical data mining and research in the allergy office. *Curr Opin Allergy Clin Immunol*, 10(3):171–177, Jun 2010.

G. Demetriou, R. Gaizauskas, H. Sun, and A. Roberts. Annalist – annotation alignment and scoring tool. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation, LREC 2008*, Marrakech, Morocco, May 2008. ELRA.

D. Demner-Fushman, W. W. Chapman, and C. J. McDonald. What can natural language processing do for clinical decision support? *J Biomed Inform*, 42(5):760–772, Oct 2009.

J. C. Denny, J. D. Smithers, R. A. Miller, and A. Spickard. "understanding" medical school curriculum content using KnowledgeMap. *Journal of the American Medical Informatics Association*, 10(4):351–362, 2003.

G. Doddington, A. Mitchell, M. Przybocki, L. Ramshaw, S. Strassel, and R. Weischedel. The automatic content extraction (ACE) program - tasks, data, & evaluation. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC 2004)*, pages 837–840, Lisbon, Portugal, May 2004.

eClinicalWorks. eclinicalworks. eClinicalWorks. http://www.eclinicalworks.com/, 2012a. [Accessed 10 October 2012].

eClinicalWorks. 2012 Summer eCW Product Innovation. eClinicalWorks. http://www.eclinicalworks.com/2012-summer-product-innovation.htm, 2012b. [Accessed 20 October 2012].

P. L. Elkin, S. H. Brown, B. A. Bauer, C. S. Husser, W. Carruth, L. R. Bergstrom, and D. L. Wahner-Roedler. A controlled trial of automated classification of negation from clinical notes. *BMC Medical Informatics and Decision Making*, 5(13), 2005.

Elsevier. Rights and Responsibilities. Elsevier B.V. http://www.elsevier.com/wps/find/authorsview.authors/rights, 2012. [Accessed 16 October 2012].

European Language Resources Association. Authors' kit. ELRA. http://www.lrec-conf.org/lrec2012/?Authors-Kit, 2012. [Accessed 16 October 2012].

M. Finkelstein-Landau and E. Morin. Extracting semantic relationships between terms: Supervised vs. unsupervised methods. In V. R. Benjamins, D. Fensel, and A. G. Pérez, editors, *Proceedings of the International Workshop on Ontological Engineering on the Global Information Infrastructure*, pages 71–80, Dagstuhl Castle, Germany, May 1999.

J. M. Fisk, P. Mutalik, F. W. Levin, J. Erdos, C. Taylor, and P. Nadkarni. Integrating query of relational and textual data in clinical databases: a case study. *J Am Med Inform Assoc*, 10(1):21–38, 2003.

K. Franzén, Gunnar, Eriksson, F. Olsson, L. Asker, P. Lidén, and J. Cöster. Protein names and how to find them. *Int J Med Inform*, 67(1–3):49–61, 2002.

C. Friedman and G. Hripcsak. Evaluating natural language processors in the clinical domain. *Methods of Information in Medicine*, 37(4-5):334–44, 1998.

C. Friedman, P. Alderson, J. Austin, J. Cimino, and S. Johnson. A general natural-language text processor for clinical radiology. *J Am Med Inform Assoc*, 1(2):161–174, March 1994.

K. Fundel, R. Küffner, and R. Zimmer. Relex–relation extraction using dependency parse trees. *Bioinformatics*, 23(3):365–71, 2007.

R. Gaizauskas and K. Humphreys. XI: A simple Prolog-based language for cross-classification and inheritance. In *Proceedings of the 7th International Conference on Artificial Intelligence: Methodology, Systems, Applications (AIMSA96)*, pages 86–95, Sozopol, Bulgaria, 1996.

R. Gaizauskas and Y. Wilks. Information extraction: Beyond document retrieval. *Journal of Documentation*, 54(1):70 – 105, 1998.

R. Gaizauskas, L. Cahill, and R. Evans. Sussex University: Description of the Sussex system used for MUC-5. In *Proceedings of the Fifth Message Undersanding Conference (MUC-5)*, pages 321–335. ARPA, Morgan Kaufmann, 1993.

R. Gaizauskas, T. Wakao, K. Humphreys, H. Cunningham, and Y. Wilks. University of Sheffield: description of the LaSIE system as used for MUC-6. In *Proceedings of the Sixth Message Understanding Conference (MUC-6)*. Morgan Kaufmann, 1995.

R. Gaizauskas, G. Demetriou, and K. Humphreys. Term recognition and classification in biological science journal articles. In *Proceedings of the Computational Terminology for Medical and Biological Applications Workshop of the 2nd International Conference on Natural Language Processing (NLP-2000)*, pages 37–44, Patras, Greece, June 2000.

R. Gaizauskas, G. Demetriou, P. J. Artymiuk, and P. Willett. Protein structures and information extraction from biological texts: The PASTA system. *Bioinformatics*, 19(1): 135–143, 2003.

R. Gaizauskas, H. Harkema, M. Hepple, and A. Setzer. Task-oriented extraction of temporal information: The case of clinical narratives. In *Proceedings of the 13th International Symposium on Temporal Representation and Reasoning (TIME2006)*, pages 188–195, 2006.

A. Gangemi, D. M. Pisanelli, and G. Steve. Understanding systematic conceptual structures in polysemous medical terms. *Proc AMIA Symp*, pages 285–289, 2000.

J. H. Gennari, M. A. Musen, R. W. Fergerson, W. E. Grosso, M. Crubézy, H. Eriksson, N. F. Noy, and S. W. Tu. The evolution of protégé: an environment for knowledge-based systems development. *International Journal Human-Computer Studies*, 58(1): 89–123, 2003. ISSN 1071-5819.

R. Ghani, R. Jones, T. Mitchell, and E. Riloff. Active learning for information extraction with multiple view feature sets. In *Proceedings of the 20th International Conference on Machine Learning (ICML 2003) Workshop on Adaptive Text Extraction and Mining*, 2003.

C. Giuliano, A. Lavelli, and L. Romano. Exploiting shallow linguistic information for relation extraction from biomedical literature. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL06)*, pages 401–408, Trento, Italy, April 2006.

M. Goadrich, L. Oliphant, and J. Shavlik. Learning to extract genic interactions using gleaner. In *Proceedings of the ICML05 workshop: Learning Language in Logic (LLL05)*, pages 62–68, Bonn, Germany, August 2005.

C. Goble, C. Wroe, and R. Stevens. The myGrid project: services, architecture and demonstrator. In S. Cox, editor, *Proceedings of UK e-Science All Hands Meeting 2003, Nottingham, UK*, pages 595–603, 2003.

T. Greenhalgh, H. W. Potts, G. Wong, P. Bark, and D. Swinglehurst. Tensions and paradoxes in electronic patient record research: a systematic literature review using the meta-narrative method. *Milbank Q*, 87(4):729–788, Dec 2009.

G. Grefenstette. *Explorations in Automatic Thesaurus Discovery*. The Kluwer International Series in Engineering and Computer Science. Kluwer Academic Publishers, Boston, 1994.

R. Grishman. Information extraction. In R. Mitkov, editor, *The Oxford Handbook of Computational Linguistics*. Oxford University Press, 2003. Chapter 30.

R. Grishman and B. Sundheim. Message understanding conference-6: A brief history. In *Proceedings of the 16th International Conference on Computational Linguistics (COLING)*, pages 466–471, Copenhagen, 1996.

C. Grover, B. Haddow, E. Klein, M. Matthews, L. Nielsen, R. Tobin, and X. Wang. Adapting a relation extraction pipeline for the BioCreAtIvE II task. In *Proceedings of the BioCreAtIvE II Workshop 2007*, Madrid, Spain, 2007.

H. Gurulingappa, A. M. Rajput, A. Roberts, J. Fluck, M. Hofmann-Apitius, and L. Toldo. Development of a benchmark corpus to support the automatic extraction of drug-related adverse effects from medical case reports. *Journal of Biomedical Informatics*, 45(5): 885 – 892, 2012. ISSN 1532-0464. doi: 10.1016/j.jbi.2012.04.008. Text Mining and Natural Language Processing in Pharmacogenomics.

U. Hahn and S. Schulz. Towards a broad-coverage biomedical ontology based on description logics. In *Proceedings of the Pacific Symposium on Biocomputing 2003*, volume 8, pages 577–588, Kauai, Hawaii, 2003.

U. Hahn, M. Romacker, and S. Schulz. MEDSYNDIKATE — a natural language system for the extraction of medical information from findings reports. *Int J Med Inform*, 67(1–3): 63–74, December 2002.

U. Hahn, E. Beisswanger, E. Buyko, E. Faessler, J. Traumller, S. Schrder, and K. Hornbostel. Iterative refinement and quality checking of annotation guidelines: How to deal effectively with semantically sloppy named entity types, such as pathological phenomena. In N. C. C. Chair), K. Choukri, T. Declerck, M. U. Doan, B. Maegaard, J. Mariani, J. Odijk, and S. Piperidis, editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may 2012. European Language Resources Association (ELRA). ISBN 978-2-9517408-7-7.

C. Hallett, R. Power, and D. Scott. Summarisation and visualisation of e-health data repositories. In *Proceedings of the UK e-Science All Hands Meeting*, pages 69–77, Nottingham, UK, 2006.

H. Harkema, R. Gaizauskas, M. Hepple, N. Davis, Y. Guo, A. Roberts, and I. Roberts. A Large-Scale Resource for Storing and Recognizing Technical Terminology. In *Proceedings of 4th International Conference on Language Resources and Evaluation*, pages 83–86, Lisbon, Portugal, 2004a.

H. Harkema, R. Gaizauskas, M. Hepple, A. Roberts, I. Roberts, N. Davis, and Y. Guo. A Large Scale Terminology Resource for Biomedical Text Processing. In L. Hirschman and J. Pustejovsky, editors, *HLT-NAACL 2004 Workshop: BioLINK 2004, Linking Biological Literature, Ontologies and Databases*, pages 53–60, Boston, MA, USA, May 2004b. ACL.

H. Harkema, I. Roberts, R. Gaizauskas, and M. Hepple. Information Extraction from Clinical Records. In S. J. Cox and D. W. Walker, editors, *Proceedings of the UK e-Science All Hands Meeting 2005*, pages 254–258, Nottingham, UK, September 2005.

Z. Harris. Discourse and sublanguage. In R. Kittredge and J. Lehrberger, editors, *Sublanguage — Studies of Language in Restricted Semantic Domains*, Foundations of Communication, chapter 11, pages 231–236. Walter de Gruyter, Berlin, 1982.

P. Haug, S. Koehler, L. M. Lau, P. Wang, R. Rocha, and S. Huff. A natural language understanding system combining syntactic and semantic techniques. *Proc Annu Symp Comput Appl Med Care*, pages 247–251, 1994.

K. Hayrinen, K. Saranto, and P. Nykanen. Definition, structure, content, use and impacts of electronic health records: a review of the research literature. *Int J Med Inform*, 77 (5):291–304, May 2008.

W. R. Hersh, H. Muller, J. R. Jensen, J. Yang, P. N. Gorman, and P. Ruch. Advancing

Biomedical Image Retrieval: Development and Analysis of a Test Collection. *J Am Med Inform Assoc*, 13(5):488–496, 2006. doi: 10.1197/jamia.M2082.

L. Hirschman, R. Grishman, and N. Sager. From text to structured information: Automatic processing of medical reports. In *AFIPS Conference Proceedings 45*, pages 267–275, Montvale, NJ, 1976. AFIPS Press.

L. Hirschman, A. Morgan, and A. Yeh. Rutabaga by any other name: extracting biological names. *Journal of Biomedical Informatics*, 35(4):247–259, 2002.

G. Hripcsak and A. Rothschild. Agreement, F-measure and reliability in information retrieval. *Journal of the American Medical Informatics Association*, 12(3):296–298, 2005.

G. Hripcsak, G. J. Kuperman, and C. Friedman. Extracting findings from narrative reports: software transferability and sources of physician disagreement. *Methods Inf Med*, 37(1):1–7, Jan 1998.

G. Hripcsak, G. J. Kuperman, C. Friedman, and D. F. Heitjan. A reliability study for evaluating information extraction from radiology reports. *J Am Med Inform Assoc*, 6 (2):143–150, 1999.

B. L. Humphreys, D. A. Lindberg, H. M. Schoolman, and G. O. Barnett. The unified medical language system: an informatics research collaboration. *J Am Med Inform Assoc*, 5(1), Jan–Feb 1998a.

K. Humphreys, R. Gaizauskas, S. Azzam, C. Huyck, B. Mitchell, H. Cunningham, and Y. Wilks. University of Sheffield: description of the LaSIE-II system as used for MUC-7. In *Proceedings of the Seventh Message Understanding Conference (MUC-7)*, Fairfax, Virginia, April 1998b. National Institute of Standards and Technology.

K. Humphreys, G. Demetriou, and R. Gaizauskas. Two applications of information extraction to biological science journal articles: Enzyme interactions and protein structures. In *Proc. of Pacific Symposium on Biocomputing (PSB-2000)*, Honolulu, Hawaii, 2000a.

K. Humphreys, G. Demetriou, and R. Gaizauskas. Bioinformatics applications of information extraction from journal articles. *Journal of Information Science*, 26(2):75–85, 2000b.

I2B2. Nlp research data sets. i2b2. `https://www.i2b2.org/NLP/DataSets/Main.php`, 2007. [Accessed 6 June 2008].

I2B2. Announcement of Data Release and Call for Participation: 2012 i2b2 Shared-Tasks and Workshop on Challenges in Natural Language Processing for Clinical Data. State of New York University at Albany. https://www.i2b2.org/NLP/TemporalRelations/Call.php, 2012. [Accessed 29 October 2012].

IBM. Mayo Clinic and IBM Host Medical Language Initiative. IBM. http://www-03.ibm.com/press/us/en/pressrelease/27140.wss, 2009. [Accessed 23 October 2012].

J. Y. Irwin, H. Harkema, L. M. Christensen, T. Schleyer, P. J. Haug, and W. W. Chapman. Methodology to develop and evaluate a semantic representation for NLP. *AMIA Annu Symp Proc*, 2009:271–275, 2009.

A. K. Jha. The promise of electronic records: Around the corner or down the road? *Journal of the American Medical Association*, 306(8):880–881, August 2011.

M. Johansson, A. Roberts, D. Chen, Y. Li, M. Delahaye-Sourdeix, N. Aswani, M. A. Greenwood, S. Benhamou, P. Lagiou, I. Holcatova, L. Richiardi, K. Kjaerheim, A. Agudo, X. Castellsague, T. V. Macfarlane, L. Barzan, C. Canova, N. S. Thakker, D. I. Conway, A. Znaor, C. M. Healy, W. Ahrens, D. Zaridze, N. Szeszenia-Dabrowska, J. Lissowska, E. Fabianova, I. N. Mates, V. Bencko, L. Foretova, V. Janout, M. P. Curado, S. Koifman, A. Menezes, V. Wunsch-Filho, J. Eluf-Neto, P. Boffetta, S. Franceschi, R. Herrero, L. Fernandez Garrote, R. Talamini, S. Boccia, P. Galan, L. Vatten, P. Thomson, D. Zelenika, M. Lathrop, G. Byrnes, H. Cunningham, P. Brennan, J. Wakefield, and J. D. McKay. Using prior information from the medical literature in GWAS of oral cancer identifies novel susceptibility variant on chromosome 4–the AdAPT method. *PLoS ONE*, 7(5):e36888, 2012.

D. Kalra, P. Singleton, D. Ingram, J. Milan, J. MacKay, D. Detmer, and A. Rector. Security and confidentiality approach for the Clinical E-Science Framework (CLEF). In S. Cox, editor, *Proceedings of UK e-Science All Hands Meeting 2003*, pages 825–832, Nottingham, UK, September 2003.

S. Katrenko and P. Adriaans. Learning relations from biomedical corpora using dependency trees. In *Knowledge Discovery and Emergent Complexity in Bioinformatics*, number 4366 in Lecture Notes in Computer Science, pages 61–80. Springer, 2007.

J.-D. Kim, T. Ohta, Y. Tateisi, and J. Tsujii. GENIA corpus — a semantically annotated corpus for bio-textmining. *Bioinformatics*, 19(1):i180–i182, 2003.

J.-D. Kim, T. Ohta, and J. Tsujii. Corpus annotation for mining biomedical events from literature. *BMC Bioinformatics*, 9(10), 2008. doi: 10.1186/1471-2105-9-10.

R. Kittredge and J. Lehrberger, editors. *Sublanguage — Studies of Language in Restricted Semantic Domains*. Foundations of Communication. Walter de Gruyter, Berlin, 1982.

D. Klein and C. D. Manning. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, pages 423–430, Morristown, NJ, USA, 2003. Association for Computational Linguistics.

I. Kohane and O. Uzuner. Letters to the editor: No structure before its time. *J Am Med Inform Assoc*, 15(5):708, 2008.

S. Kulick, A. Bies, M. Liberman, M. Mandel, R. McDonald, M. Palmer, A. Schein, L. Ungar, S. Winters, and P. White. Integrated annotation for biomedical information extraction. In L. Hirschman and J. Pustejovsky, editors, *HLT-NAACL 2004 Workshop: BioLINK 2004, Linking Biological Literature, Ontologies and Databases*, pages 61–68, Boston, Massachusetts, USA, May 6 2004. Association for Computational Linguistics.

Y. Li, K. Bontcheva, and H. Cunningham. SVM based learning system for information extraction. In *Deterministic and statistical methods in machine learning: first international workshop*, number 3635 in Lecture Notes in Computer Science, pages 319–339. Springer, 2005.

E. D. Liddy. *Encyclopedia of Library and Information Science*, chapter Natural Language Processing. Marcel Decker, Inc., NY, USA, 2 edition, 2003.

D. A. Lindberg, B. L. Humphreys, and A. T. McCray. The unified medical language system. *Methods of Information in Medicine*, 32(4):281–291, August 1993.

H. Liu, Y. Lussier, and C. Friedman. Disambiguating ambiguous biomedical terms in biomedical narrative text: an unsupervised method. *Journal of Biomedical Informatics*, 34(4):249–261, August 2001.

K. Liu, K. J. Mitchell, W. W. Chapman, and R. S. Crowley. Automating tissue bank annotation from pathology reports - comparison to a gold standard expert annotation set. *AMIA Annu Symp Proc*, pages 460–464, 2005.

R. K. Los, A. M. van Ginneken, and J. van der Lei. OpenSDE: a strategy for expressive and flexible structured data entry. *Int J Med Inform*, 74(6):481–490, Jul 2005.

Y. Lussier, T. Borlawsky, D. Rappaport, Y. Liu, and C. Friedman. PhenoGO: Assigning phenotypic context to Gene Ontology annotations with natural language processing. In *Pac Symp Biocomput*, pages 64–75, Hawaii, USA, January 2006.

I. Mani and G. Wilson. Robust temporal processing of news. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL 2000)*, pages 69–76, New Brunswick, New Jersey, 2000.

J. Mayer, S. Shen, B. R. South, S. Meystre, F. J. Friedlin, W. R. Ray, and M. Samore. Inductive creation of an annotation schema and a reference standard for de-identification of va electronic clinical notes. In *AMIA Annu Symp Proc.*, pages 416–420. AMIA, November 2009.

A. McCray, O. Bodenreider, J. Malley, and A. Browne. Evaluating UMLS Strings for Natural Language Processing. In *Proceedings of the 2001 American Medical Informatics Association Annual Symposium*, pages 448–452, Portland, OR, USA, 2001.

A. McCray, A. Browne, and O. Bodenreider. The lexical properties of the gene ontology (go). In *Proceedings of the 2002 American Medical Informatics Association Annual Symposium*, pages 504–508, San Antonio, TX, USA, 2002.

A. T. McCray. Extending a natural language parser with UMLS knowledge. *Proc Annu Symp Comput Appl Med Care*, pages 194–198, 1991.

A. T. McCray and S. J. Nelson. The representation of meaning in the umls. *Methods of Information in Medicine*, 34(1–2):193–201, March 1995.

A. T. McCray, S. Srinivasan, and A. C. Browne. Lexical methods for managing variation in biomedical terminologies. In J. G. Ozbolt, editor, *Proceedings of the 18th Annual Symposium on Computer Applications in Medical Care*, pages 235–239, Philadelphia, 1994. American Medical Informatics Association, Hanley and Belfus.

T. McEnery and A. Wilson. *Corpus Linguistics*. Edinburgh Textbooks in Empirical Lingusistics. Edinburgh University Press, Edinburgh, 1996.

Medical Research Council. Personal Information in Medical Research, 2000.

N. Menachemi, D. L. Ettel, R. G. Brooks, and L. Simpson. Charting the use of electronic health records and other information technologies among child health providers. *BMC Pediatr*, 6:21, 2006.

S. Meystre and P. J. Haug. Natural language processing to extract medical problems from electronic clinical documents: Performance evaluation. *Journal of Biomedical Informatics*, 39(6):589–599, 2006.

S. Meystre, G. Savova, K. Kipper-Schuler, and J. Hurdle. Extracting information from textual documents in the electronic health record: A review of recent research. *IMIA Yearbook of Medical Informatics*, pages 128–144, 2008.

S. M. Meystre, J. Thibault, S. Shen, J. F. Hurdle, and B. R. South. Textractor: a hybrid system for medications and reason for their prescription extraction from clinical text documents. *J Am Med Inform Assoc*, 17(5):559–562, 2010a.

S. M. Meystre, J. Thibault, S. Shen, J. F. Hurdle, and B. R. South. Automatically detecting medications and the reason for their prescription in clinical narrative text documents. *Stud Health Technol Inform*, 160(Pt 2):944–948, 2010b.

S. Mika and B. Rost. Protein names precisely peeled off free text. *Bioinformatics*, 20 (Supplement 1):i241–i247, 2004.

A. Mikheev, M. Moens, and C. Grover. Named Entity recognition without gazetteers. In *Proceedings of the ninth conference of the European chapter of the Association for Computational Linguistics (EACL'99)*, pages 1–8, Bergen, Norway, 1999.

S. Miller, M. Crystal, H. Fox, L. Ramshaw, R. Schwartz, R. Stone, and R. Weischedel. Algorithms that learn to extract information: Bbn: Description of the SIFT system as used for MUC-7. In *Proceedings of the Seventh Message Understanding Conference (MUC-7)*, Fairfax, VA, USA, April 1998.

MRC. Medical Research Council. MRC. `http://www.mrc.ac.uk`, 2012. [Accessed 23 October 2012].

H. Müller, T. Deselaers, T. M. Lehmann, P. D. Clough, and W. Hersh. Overview of the imageclefmed 2006 medical retrieval and annotation tasks. In *Cross Language Evaluation Forum (CLEF) Workshop 2006*, volume 4730, pages 595–608, Alicante, Spain, 2007. Springer.

P. M. Nadkarni, L. Ohno-Machado, and W. W. Chapman. Natural language processing: an introduction. *J Am Med Inform Assoc*, 18(5):544–551, 2011.

NIST. *Proceedings of the 4th Conference on Message Understanding*, McLean, Virginia, USA, June 1992a. National Institute of Standards and Technology, Morgan Kaufmann.

NIST. *Proceedings Fifth Message Understanding Conference (MUC-5)*, Baltimore, Maryland, August 1993b. National Institute of Standards and Technology, Morgan Kaufmann.

NIST. *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, 1995c. National Institute of Standards and Technology, Morgan Kaufmann.

NIST. *Proceedings of the Seventh Message Understanding Conference (MUC-7)*, Fairfax, Virginia, April 1998d. National Institute of Standards and Technology, National Institute of Standards and Technology.

NLM. *UMLS Knowledge Sources, 2007AB*. National Library of Medicine, July 2007.

C. Nédellec. Learning language in logic - genic interaction extraction challenge. In *Proceedings of the ICML05 workshop: Learning Language in Logic (LLL05)*, pages 31–37, Bonn, Germany, August 2005a.

C. Nédellec. Learning language in logic - genic interaction extraction challenge. In *Proceedings of the ICML05 Workshop on Learning Language in Logic*, pages 31–37, Bonn, Germany, 2005b.

NHS. NHS Connecting for Health. . `http://www.connectingforhealth.nhs.uk/`, 2012. [Accessed 29 October 2012].

NLM. Training and educational resources for UMLS users. U.S. National Library of Medicine. `http://www.nlm.nih.gov/research/umls/user_education/index.html`, 2012. [Accessed 23 October 2012].

W. A. Nowlan, A. L. Rector, S. Kay, B. Horan, and A. Wilson. A patient care workstation based on user centred design and a formal theory of medical terminology: PEN&PAD and the SMK formalism. *Proc Annu Symp Comput Appl Med Care*, pages 855–857, 1991.

P. V. Ogren. Knowtator: a Protégé plug-in for annotated corpus construction. In *Proceedings of the 2006 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 273–275, Morristown, NJ, USA, 2006. Association for Computational Linguistics.

P. V. Ogren, G. Savova, J. D. Buntrock, and C. G. Chute. Building and evaluating annotated corpora for medical nlp systems. In *Proc AMIA Symp*, page 1050, 2006.

OHNLP. Open Health Natural Language Processing (OHNLP). OHNLP. `http://ohnlp.org`, 2012a. [Accessed 23 October 2012].

OHNLP. Clinical Text Analysis and Knowledge Extraction System (cTAKES). National Cancer Institute. https://wiki.nci.nih.gov/pages/viewpage.action?pageId=65733244, 2012b. [Accessed 23 October 2012].

OHNLP. The MedKAT Pipeline. SourceForge. http://ohnlp.sourceforge.net/MedKATp/, 2012c. [Accessed 23 October 2012].

S. Pakhomov, J. Buntrock, and P. Duffy. High throughput modularized NLP system for clinical text. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05), interactive poster and demonstration sessions*, pages 25–28, Ann Arbor, MI, USA, 2005.

S. Pakhomov, A. Coden, and C. Chute. Developing a corpus of clinical notes manually annotated for part-of-speech. *Int J Med Inform*, 75(6):418–429, Jun 2006.

J. Park and J. Kim. Named entity recognition. In S. Ananiadou and J. McNaught, editors, *Text Mining for Biology and Biomedicine*, chapter 6, pages 121–142. Artech House, 2006.

J. Patrick and M. Sabbagh. An active learning process for extraction and standardisation of medical measurements by a trainable fsa. In A. Gelbukh, editor, *CICLing 2011*, number 6609 in Lecture Notes in Computer Science, pages 151–162, Heideliberg, 2011. Springer-Verlag.

J. P. Pestian, C. Brew, P. Matykiewicz, D. J. Hovermale, N. Johnson, K. B. Cohen, and W. Duch. A shared task involving multi-label classification of clinical free text. In *Proceedings of the Workshop on BioNLP 2007: Biological, Translational, and Clinical Language Processing*, BioNLP '07, pages 97–104, Stroudsburg, PA, USA, 2007. Association for Computational Linguistics.

J. P. Pestian, P. Matykiewicz, M. Linn-Gust, B. South, O. Uzuner, J. Wiebe, K. B. Cohen, J. Hurdle, and C. Brew. Sentiment Analysis of Suicide Notes: A Shared Task. *Biomed Inform Insights*, 5(Suppl 1):3–16, Jan 2012.

D. M. Pisanelli, A. Gangemi, M. Battaglia, and C. Catenacci. Coping with medical polysemy in the semantic web: the role of ontologies. *Stud Health Technol Inform*, 107(Pt 1):416–419, 2004.

S. M. Powsner, J. C. Wyatt, and P. Wright. Opportunities for and challenges of computerisation. *Lancet*, 352(9140):1617–1622, Nov 1998.

D. Proux, F. Rechenmann, L. Julliard, V. Pillet, and B. Jacq. Detecting gene symbols and names in biological texts: a first step toward pertinent information extraction. *Genome Informatics*, 9:72–80, 1998.

J. Pustejovsky, J. C. no, J. Zhang, M. Kotecki, and B. Cochran. Robust relational parsing over biomedical literature: Extracting inhibit relations. In *Pac Symp Biocomput*, pages 362–373, Hawaii, USA, January 2002.

J. Pustejovsky, J. C. no, R. Ingria, R. Saurí, R. Gaizauskas, A. Setzer, and G. Katz. TimeML: Robust specification of event and temporal expressions in text. In *Proceedings of the Fifth International Workshop on Computational Semantics (IWCS-5)*, Tilburg, January 2003.

D. L. Ranum. Knowledge-based understanding of radiology text. *Comput Methods Programs Biomed*, 30(2-3):209–215, 1989.

D. Rebholz-Schuhmann, A. J. Yepes, E. M. Van Mulligen, N. Kang, J. Kors, D. Milward, P. Corbett, E. Buyko, E. Beisswanger, and U. Hahn. CALBC silver standard corpus. *J Bioinform Comput Biol*, 8(1):163–179, Feb 2010.

A. Rector. Knowledge driven software and fractal tailoring: Ontologies in development environments for clinical systems. In *Proceedings of the 2010 conference on Formal Ontology in Information Systems: Proceedings of the Sixth International Conference (FOIS 2010)*, pages 17–28, Amsterdam, The Netherlands, The Netherlands, 2010. IOS Press. ISBN 978-1-60750-534-1.

A. Rector, J. Rogers, A. Taweel, D. Ingram, D. Kalra, J. Milan, P. Singleton, R. Gaizauskas, M. Hepple, D. Scott, and R. Power. CLEF — Joining up Healthcare with Clinical and Post-Genomic Research. In *Proceedings of UK e-Science All Hands Meeting 2003*, pages 264–267, Nottingham, UK, 2003.

A. L. Rector. Thesauri and formal classifications: terminologies for people and machines. *Methods Inf Med*, 37(4-5):501–509, Nov 1998.

A. L. Rector. Clinical terminology: why is it so hard? *Methods Inf Med*, 38(4-5):239–252, Dec 1999.

A. L. Rector and J. Rogers. Ontological and practical issues in using a description logic to represent medical concept systems: Experience from galen. In P. Barahona, F. Bry, E. Franconi, N. Henze, and U. Sattler, editors, *Reasoning Web*, volume 4126 of *Lecture Notes in Computer Science*, pages 197–231. Springer, 2006. ISBN 3-540-38409-X.

A. L. Rector, A. J. Glowinski, W. A. Nowlan, and A. Rossi-Mori. Medical-concept models and medical records: an approach based on GALEN and PEN&PAD. *J Am Med Inform Assoc*, 2(1):19–35, 1995.

M.-L. Reinberger, P. Spyns, W. Daelemans, and R. Meersman. Mining for lexons: applying unsupervised learning methods to create ontology bases. In *Proceedings of the International Conference on Ontologies, Databases and Applications of Semantics (ODBASE 2003)*, Catania, Sicily, November 2003.

E. Riloff. Automatically generating extraction patterns from untagged text. In *AAAI/IAAI, Vol. 2*, pages 1044–1049, 1996.

T. Rindflesch and M. Fiszman. The interaction of domain knowledge and linguistic structure in natural language processing: interpreting hypernymic propositions in biomedical text. *J Biomed Inform*, 36(6):462–477, 2003.

T. C. Rindflesch, B. Libbus, D. Hristovski, A. R. Aronson, and H. Kilicoglu. Semantic relations asserting the etiology of genetic diseases. In *Proc AMIA Annu Fall Symp*, pages 554–558, Washington, DC, USA, November 2003.

A. Roberts, R. Gaizauskas, M. Hepple, N. Davis, G. Demetriou, Y. Guo, J. Kola, I. Roberts, A. Setzer, A. Tapuria, and B. Wheeldin. The CLEF Corpus: Semantic Annotation of Clinical Text. In *Proceedings of the 2007 American Medical Informatics Association Annual Symposium*, pages 625–629, Chicago, IL, USA, 2007.

A. Roberts, R. Gaizauskas, and M. Hepple. Extracting clinical relationships from patient narratives. In *Proceedings of the Workshop on BioNLP 2008*, Columbus, OH, USA, June 2008a. Association for Computational Linguistics.

A. Roberts, R. Gaizauskas, M. Hepple, G. Demetriou, Y. Guo, A. Setzer, and I. Roberts. Semantic annotation of clinical text: The CLEF corpus. In *Proceedings of Building and evaluating resources for biomedical text mining: workshop at LREC 2008*, Marrakech, Morocco, May 2008b.

A. Roberts, R. Gaizauskas, M. Hepple, and Y. Guo. Combining terminology resources and statistical methods for entity recognition: an evaluation. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation, LREC 2008*, Marrakech, Morocco, May 2008c.

A. Roberts, R. Gaizauskas, M. Hepple, and Y. Guo. Mining clinical relationships from patient narratives. *BMC Bioinformatics*, 9 Suppl 11(S3), November 2008d.

A. Roberts, R. Gaizauskas, M. Hepple, G.Demetriou, Y. Guo, I. Roberts, and A. Setzer. Building a semantically annotated corpus of clinical texts. *Journal of Biomedical Informatics*, 42(5):950–66, October 2009.

J. Rogers, C. Puleston, and A. Rector. The CLEF chronicle: Patient histories derived from electronic health records. *Data Engineering Workshops, 2006. Proceedings. 22nd International Conference on*, pages x109–x109, 2006.

B. Rosario and M. A. Hearst. Classifying semantic relations in bioscience texts. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, page 430, Morristown, NJ, USA, 2004. Association for Computational Linguistics.

B. Rosario and M. A. Hearst. Multi-way relation classification: application to protein-protein interactions. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 732–739, Morristown, NJ, USA, 2005. Association for Computational Linguistics.

S. T. Rosenbloom, R. A. Miller, K. B. Johnson, P. L. Elkin, and S. H. Brown. Interface terminologies: facilitating direct entry of clinical data into electronic health record systems. *J Am Med Inform Assoc*, 13(3):277–288, 2006.

S. T. Rosenbloom, J. C. Denny, H. Xu, N. Lorenzi, W. W. Stead, and K. B. Johnson. Data from clinical notes: a perspective on the tension between structure and flexible documentation. *J Am Med Inform Assoc*, 18(2):181–186, 2011.

N. Sager. Syntactic analysis of natural language. In *Advances in Computers*, volume 8, pages 153–188. Academic Press, NY, 1978a.

N. Sager. Natural language information formatting: The automatic conversion of texts to a structured data base. In M. Yovits, editor, *Advances in Computers*, volume 17, pages 89–162. Academic Press, NY, 1978b.

N. Sager, M. Lyman, C. Bucknall, N. Nhan, and L. Tick. Natural language processing and the representation of clinical data. *J Am Med Inform Assoc*, 1(2):142–160, March-April 1994.

T. G. Savel and S. Foldy. The role of public health informatics in enhancing public health surveillance. *Centers for Disease Control and Prevention Morbidity and Mortality Weekly Reports*, 61:20–24, July 2012.

G. K. Savova, K. Kipper-Schuler, J. D. Buntrock, and C. G. Chute. UIMA-based clinical information extraction system. In *Towards Enhanced Interoperability for Large HLT*

Systems: UIMA for NLP, held in conjunction with LREC 2008, Marrakech, Morocco, May 2008.

G. K. Savova, J. J. Masanz, P. V. Ogren, J. Zheng, S. Sohn, K. C. Kipper-Schuler, and C. G. Chute. Mayo clinical Text Analysis and Knowledge Extraction System (cTAKES): architecture, component evaluation and applications. *J Am Med Inform Assoc*, 17(5):507–513, 2010.

G. K. Savova, W. W. Chapman, J. Zheng, and R. S. Crowley. Anaphoric relations in the clinical narrative: corpus creation. *J Am Med Inform Assoc*, 18(4):459–465, 2011.

T. Schleyer. A salient problem in informatics? *J Am Med Inform Assoc*, 15(5):707; author reply 708, 2008.

M. Schuemie, J. Kors, and B. Mons. Word sense disambiguation in the biomedical domain: An overview. *Journal of Computational Biology*, 12(5):554–565, 2005.

C. Scott, M. DeRouen, and L. Crawley. The language of hope: Therapeutic intent in stem-cell clinical trials. *AJOB Primary Research*, 2010.

D. Scott, R. Barone, and R. Koeling. Corpus annotation as a scientific task. In N. C. C. Chair), K. Choukri, T. Declerck, M. U. Doan, B. Maegaard, J. Mariani, J. Odijk, and S. Piperidis, editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may 2012. European Language Resources Association (ELRA). ISBN 978-2-9517408-7-7.

C. R. Selden and B. L. Humphreys. Unified medical language system (UMLS): January 1986 through december 1996. *Current Bibliographies in Medicine*, 96(8), 1997.

B. Settles. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin-Madison, 2010.

S. R. Simon, C. S. Soran, R. Kaushal, C. A. Jenter, L. A. Volk, E. Burdick, P. D. Cleary, E. J. Orav, E. G. Poon, and D. W. Bates. Physicians' use of key functions in electronic health records from 2005 to 2007: a statewide survey. *J Am Med Inform Assoc*, 16(4): 465–470, 2009.

R. Snow, B. O'Connor, D. Jurafsky, and A. Y. Ng. Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '08, pages 254–263, Stroudsburg, PA, USA, 2008. Association for Computational Linguistics.

B. R. South, S. Shen, M. Jones, J. Garvin, M. H. Samore, W. W. Chapman, and A. V. Gundlapalli. Developing a manually annotated clinical document corpus to identify phenotypic information for inflammatory bowel disease. *BMC Bioinformatics*, 10 Suppl 9:S12, 2009.

B. R. South, S. Shen, W. W. Chapman, S. Delisle, M. H. Samore, and A. V. Gundlapalli. Analysis of False Positive Errors of an Acute Respiratory Infection Text Classifier due to Contextual Features. *AMIA Summits Transl Sci Proc*, 2010:56–60, 2010.

B. R. South, S. Shen, R. Barrus, S. L. DuVall, O. Uzuner, and C. Weir. Qualitative analysis of workflow modifications used to generate the reference standard for the 2010 i2b2/VA challenge. *AMIA Annu Symp Proc*, 2011:1243–1251, 2011.

J. F. Sowa. *Knowledge Representation: Logical, Philosophical and Computational Foundations*. Brooks/Cole Thomson Learning, Pacific Grove, CA, 2000.

P. Spyns. Natural language processing in medicine: an overview. *Methods of Information in Medicine*, 35(4–5):285–301, December 1996.

M. H. Stanfill, M. Williams, S. H. Fenton, R. A. Jenders, and W. R. Hersh. A systematic literature review of automated clinical coding and classification systems. *J Am Med Inform Assoc*, 17(6):646–651, 2010.

M. Stevenson and R. Gaizauskas. Using Corpus-derived Name Lists for Named Entity Recognition. In *Proceedings of the Sixth Conference on Applied Natural Language Processing (ANLP-2000)*, pages 84–89, Seattle, Washington, USA, 2000.

R. Stewart, M. Soremekun, G. Perera, M. Broadbent, F. Callard, M. Denis, M. Hotopf, G. Thornicroft, and S. Lovestone. The South London and Maudsley NHS Foundation Trust Biomedical Research Centre (SLAM BRC) case register: development and descriptive data. *BMC Psychiatry*, 9:51, 2009.

L. Tanabe and W. Wilbur. Tagging gene and protein names in biomedical text. *Bioinformatics*, 18(8):1124–1132, 2002.

L. Tanabe, N. Xie, L. H. Thom, W. Matten, and W. J. Wilbur. GENETAG: a tagged corpus for gene/protein named entity recognition. *BMC Bioinformatics*, 6(Suppl 1)(S3), 2005.

J. Thomas, D. Milward, C. Ouzounis, S. Pulman, and M. Carroll. Automatic extraction of protein interactions from scientific abstracts. In *Pac Symp Biocomput*, pages 541–551, Hawaii, USA, January 2000.

C. A. Thompson, M. E. Califf, and R. J. Mooney. Active learning for natural language parsing and information extraction. In *Proc. 16th International Conf. on Machine Learning*, pages 406–414. Morgan Kaufmann, San Francisco, CA, 1999.

TREC. TREC genomics track. Oregon Health and Science University. http://ir.ohsu.edu/genomics, 2008. [Accessed 6 June 2008].

TREC. TREC 2011 medical track. National Institute of Standards and Technology http://trec.nist.gov/data/medical2011.html, 2011. [Accessed 29 October 2012].

TREC. Call for participation: Text Retrieval Conference (TREC) 2012. National Institute of Standards and Technology http://trec.nist.gov/pubs/call2012.html, 2012. [Accessed 29 October 2012].

United States Government. Public Law 111-5 - American Recovery and Reinvestment Act. United States Government Printing Office. http://www.gpo.gov/fdsys/pkg/PLAW-111publ5/pdf/PLAW-111publ5.pdf, 2009. [Accessed 29 October 2012].

University of Pittsburgh Department of Biomedical Informatics. University of Pittsburgh NLP Repository. University of Pittsburgh. http://www.dbmi.pitt.edu/nlpfront, 2012. [Accessed 29 October 2012].

University of Sheffield. LaSIE - design of the Sheffield NLP group MUC6 system. Technical report, Department of Computer Science, University of Sheffield, June 1996.

University of Sheffield. Clinical e-science framework. university of sheffield. University of Sheffield. http://nlp.shef.ac.uk/clef/, 2008. [Accessed 2 October 2008].

University of Sheffield. GATE – General Architecture for Text Engineering. University of Sheffield. http://gate.ac.uk, 2012. [Accessed 20 October 2012].

O. Uzuner. Recognizing obesity and comorbidities in sparse data. *J Am Med Inform Assoc*, 16(4):561–570, 2009.

O. Uzuner, Y. Luo, and P. Szolovits. Evaluating the state-of-the-art in automatic de-identification. *J Am Med Inform Assoc*, 14(5):550–563, 2007.

O. Uzuner, I. Goldstein, Y. Luo, and I. Kohane. Identifying patient smoking status from medical discharge records. *J Am Med Inform Assoc*, 15(1):14–24, 2008.

O. Uzuner, J. Mailoa, R. Ryan, and T. Sibanda. Semantic relations for problem-oriented medical records. *Artificial Intelligence in Medicine*, 50(2):63 – 73, 2010a. ISSN 0933-3657. doi: 10.1016/j.artmed.2010.05.006.

O. Uzuner, I. Solti, and E. Cadag. Extracting medication information from clinical text. *J Am Med Inform Assoc*, 17(5):514–518, 2010b.

O. Uzuner, I. Solti, F. Xia, and E. Cadag. Community annotation experiment for ground truth generation for the i2b2 medication challenge. *J Am Med Inform Assoc*, 17(5): 519–523, 2010c.

O. Uzuner, B. R. South, S. Shen, and S. L. DuVall. 2010 i2b2/VA challenge on concepts, assertions, and relations in clinical text. *J Am Med Inform Assoc*, 18(5):552–556, 2011.

O. Uzuner, A. Bodnari, S. Shen, T. Forbush, J. Pestian, and B. South. Evaluating the state of the art in coreference resolution for electronic medical records. *J Am Med Inform Assoc.*, 1(19):786–791, Sep 2012.

J. van der Lei. Closing the loop between clinical practice, research, and education: The potential of electronic patient records. *Methods of Information in Medicine*, 41(1): 51–54, 2002.

M. Verhagen, R. Gaizauskas, F. Schilder, M. Hepple, G. Katz, and J. Pustejovsky. SemEval-2007 task 15: TempEval temporal relation identification. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 75–80, Prague, 2007.

T. Wang, Y. Li, K. Bontcheva, H. Cunningham, and J. Wang. Automatic extraction of hierarchical relations from text. In *The Semantic Web: Research and Applications. 3rd European Semantic Web Conference, ESWC 2006*, number 4011 in Lecture Notes in Computer Science, pages 215–229. Springer, 2006.

WHO. International Classification of Diseases (ICD). World Health Organisation. `http://www.who.int/classifications/icd`, 2008. [Accessed 6 June 2008].

I. H. Witten and E. Frank. *Data Mining: Practical machine learning tools and techniques with Java implementations*. The Morgan Kaufmann Series in Data Management Systems. Morgan Kaufmann, San Fransisco, second edition, 2005.

F. Xia and M. Yetisgen-Yildiz. Clinical corpus annotation: challenges and strategies. In *Proceedings of Third Workshop on Building and Evaluating Resources for Biomedical Text Mining (BioTxtM'2012) of the International Conference on Language Resources and Evaluation (LREC)*, Istanbul, May 2012.

H. Xu, P. D. Stetson, and C. Friedman. Methods for building sense inventories of abbreviations in clinical notes. *J Am Med Inform Assoc*, 16(1):103–108, 2009.

K. Yamamoto, T. Kudo, A. Konagaya, and Y. Matsumoto. Protein name tagging for biomedical annotation in text. In *Proceedings of the ACL 2003 Workshop on Natural Language Processing in Biomedicine*, pages 65–72, Sapporo, Japan, 2003.

D. Zelenko, C. Aone, and A. Richardella. Kernel methods for relation extraction. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 71–78, Philadelphia, PA, USA, July 2002.

Q. T. Zeng, S. Goryachev, S. Weiss, M. Sordo, S. N. Murphy, and R. Lazarus. Extracting principal diagnosis, co-morbidity and smoking status for asthma research: evaluation of a natural language processing system. *BMC Med Inform Decis Mak*, 6:30, 2006.

J. Zheng, W. Chapman, R. Crowley, and G. Savova. Coreference resolution: a review of general methodologies and applications in the clinical domain. *J Biomed Inform*, 44 (6):1113–1122, December 2011.

J. Zheng, W. W. Chapman, T. A. Miller, C. Lin, R. S. Crowley, and G. K. Savova. A system for coreference resolution for the clinical narrative. *J Am Med Inform Assoc*, 19:660–667, 2012. doi:10.1136/amiajnl-2011-000599.

G. Zhou, J. Su, J. Zhang, and M. Zhang. Exploring various knowledge in relation extraction. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, pages 427–434, Ann Arbor, MI, USA, June 2005.

P. Zweigenbaum. MENELAS: an access system for medical records using natural language. *Computer Methods and Programs in Biomedicine*, 45:117–120, 1994.

P. Zweigenbaum, B. Bachimont, J. Bouaud, J. Charlet, and J.-F. Boisvieux. A multilingual architecture for building a normalised conceptual representation from medical language. In *Proc Annu Symp Comput Appl Med Care*, pages 357–3g61, New York, NY, USA, 1995.

P. Zweigenbaum, D. Demner-Fushman, H. Yu, and K. B. Cohen. Frontiers of biomedical text mining: current progress. *Brief. Bioinformatics*, 8(5):358–375, Sep 2007.

Citation Index