AUTOMATIC SUMMARIZATION OF CLINICAL ABSTRACTS FOR EVIDENCE-BASED MEDICINE

BY

RODNEY L. SUMMERSCALES

Submitted in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Computer Science
in the Graduate College of the
Illinois Institute of Technology

Approved _____
Advisor

Chicago, Illinois
December 2013

ACKNOWLEDGMENT

I thank my advisor, Dr. Shlomo Argamon. He provided the original inspiration for this project. I am grateful for his guidance and time spent reviewing my papers and presentations. I thank Dr. David Grossman and Dr. Jahna Otterbacher for serving on my thesis proposal committee. I thank Dr. Mustafa Bilgic, Dr. Boris Glavic and Dr. Libby Hemphill for serving on my dissertation committee. I appreciate their supervision and participation in our Machine Learning reading group meetings.

Several individuals made significant contributions to my thesis work. I thank Dr. Jordan Hupert and Dr. Alan Schwartz from the University of Illinois at Chicago Medical School. Not only did they contribute their experience and knowledge of Evidence-based medicine which guided the direction of the project, but they also sacrificed their time to review and evaluate the summary results from the system. Fellow graduate students Shangda Bai and Nandhi Prabhu Mohan contributed to my thesis work by annotating medical abstracts and developing a web-based system for evaluating summaries. These were tedious and time-consuming tasks. I am extremely grateful to both Shangda and Prabhu for their assistance. I thank fellow lab-mate Dr. Ken Bloom for our conversations in the lab.

I am truly grateful to the Chicago chapter of the ARCS Foundation and the Illinois Institute of Technology for their financial support. I value the encouragement and moral support that I received from members of both organizations.

Finally, I thank my family for their love and support, particularly my grandparents who had unreasonable confidence in my abilities. Most of all, I thank my wife, Tiffany, for her patience, love and emotional support.

TABLE OF CONTENTS

Page

ACKNOWLEDGEMENT iii

LIST OF TABLES x

LIST OF FIGURES xii

ABSTRACT xiii

CHAPTER

1. INTRODUCTION 1

1.1. Summary statistics 4
1.2. EBM-oriented summaries 5
1.3. Generating summaries 7
1.4. Claims 9
1.5. Application to other domains 9
1.6. Contributions 10

2. BACKGROUND AND PRIOR WORK 11

2.1. Online trial registries 11
2.2. PICO queries 11
2.3. PubMed and MEDLINE 12
2.4. UMLS Metathesaurus 12
2.5. Clinical question answering systems 13
2.6. Semantic MEDLINE 15
2.7. Semi-automatic creation of clinical trial databases 16
2.8. Contributions 16

3. SYSTEM ARCHITECTURE 18

3.1. Problem overview 18
3.2. Overview of ACRES 24
3.3. Pre-processing 25
3.4. Extract key elements 26
3.5. Associate elements 27

4. DATA 31

4.1. Corpora annotations 31
4.2. Structured vs. Unstructured abstracts 32
4.3. Corpora construction 33
4.4. Corpora characteristics 35

4.5. Contributions 38

5. EXTRACTING KEY ELEMENTS 41

5.1. Pre-processing stage 42
5.2. Rule-based extraction 55
5.3. Classifier-based extraction 62
5.4. Re-ranking classifier output 68
5.5. Post-processing classifier output 70
5.6. Contributions and related work 71

6. SUMMARY CONSTRUCTION 76

6.1. Element associations 76
6.2. Clustering mentions 77
6.3. Associating mentions and values 82
6.4. Calculating summary statistics 93
6.5. Compiling summaries 96
6.6. Contributions and prior work 97

7. EVALUATION 98

7.1. Methodology 98
7.2. Element extraction 99
7.3. Mention clustering 107
7.4. Value association 109
7.5. Summary evaluation 114
7.6. Exact match criteria 120
7.7. Ceiling analysis 123
7.8. Boosting outcomes 128
7.9. Expert evaluations 134
7.10. Contributions 141

8. SUMMARY AND CONCLUSION 143

8.1. Contributions 143
8.2. Summarization in other domains 143
8.3. Future work 145
8.4. Conclusion 147

APPENDIX 148

A. ARTICLE ANNOTATION SCHEME 149
A.1. Overview 150

A.2. Annotating Abstracts 151

B. EBM SUMMARY STRUCTURE 160

B.1. Study element 160
B.2. Sample summary 163

BIBLIOGRAPHY 166

LIST OF TABLES

Table		Page
1.1	Sample PICO query	3
4.1	Basic characteristics of each corpus: the total number of abstracts in each corpus; the number and percentage of abstracts that have section labels; the number and percentage of abstracts containing at least one group size, outcome number or event rate; the average number of sentences in an abstract; the average number of tokens in a sentence; and the average number of acronym occurrences that appear in an abstract.	38
4.2	The total number of annotated numeric values for each corpus as well as the average number of annotated values in an abstract.	39
4.3	The total number of annotated condition, group and outcome mentions for each corpus as well as the average number of annotated mentions in each abstract and the average length (number of tokens) of each mention type.	39
4.4	The total number of *unique* condition, group and outcome *entities* for each corpus as well as the average number of unique entities in each abstract and the average number of mentions that refer to a unique entity for each type.	40
4.5	Statistics regarding the number absolute risk reduction (ARR) calculations that should be computed for all of the abstracts in each corpus based on information given in the abstract: the total number of ARR that could be computed for each corpus and average per abstract; the total and average number of ARRs that can only be computed from outcome numbers and group sizes that appear in abstract (*computed event rates*); and the total and average number of ARRs that can only be computed from event rates that appear in the abstract text (*textual event rates*).	40
5.1	Comparison token patterns and their normalized version.	47
5.2	The special phrases that are identified and chunked.	48
5.3	The numeric patterns that are recognized before parsing along with the parsed form that is inserted back into the parse tree.	48
5.4	Words for common statistics used in clinical research.	53

5.5 Common terms identified by Xu et al.[52] that are often used to describe trial participants. 54

5.6 Time unit strings used to identify time values. 54

5.7 Units of measure used to identify measurement values. 54

5.8 Special values that can be identified using rule-based approach. . . 57

5.9 Patterns used for identifying special values values. 58

5.10 Patterns used for parsing age phrases and recognizing age values. . 60

5.11 Negation words used by the system. 70

6.1 A description of the associations that need to be made. 78

6.2 Common words that are ignored when comparing mention to see if they match. 81

6.3 Common terms that often indicate the role of a treatment group in a study. An experimental group mention cannot contain any control terms. 82

6.4 Common patterns used when reporting both the number of outcomes and the event rate for an outcome. 85

6.5 Common lemmas that indicate a problem or recovery. 95

7.1 Recall, precision and F-score for the summarization system and baseline system for extracted condition, group and outcome mentions. . 101

7.2 Recall, precision and F-score for condition, group and outcome mention extractors with different feature sets. 104

7.3 Recall, precision and F-score for the summarization system, baseline system and system variants with different feature sets for extracted group size, outcome numbers and event rates. 106

7.4 Recall, precision and F-score for the summarization system and baseline system for clustering detected condition, group and outcome mentions. 109

7.5 Recall, precision and F-score for the summarization system and baseline system for associating detected group sizes with detected group mentions and detected outcome measurements with detected group and outcome mentions. 114

7.6 Recall, precision and F-score for finding age phrases and the resulting age values that appear in the summary. 119

7.7 Recall, precision and F-score for the summarization system and baseline system for summary elements. 119

7.8 Correctly computing ARR values. Results reported for qualitatively correct ARR values interpreted as false positives and true positives. 120

7.9 Finding summaries that contain at least one correct ARR value (Any correct); at least one correct and no incorrect ARR values (Correct only); and all correct ARR values and no errors (Exact). Results reported for qualitatively correct ARR values interpreted as false positives. 120

7.10 Comparison of mention extraction performance using partial match and exact match criteria. 121

7.11 Comparison of summary element performance using partial match and exact match criteria. 122

7.12 Comparison of ARR value performance using partial match and exact match criteria for group and outcome mentions associated with the values. 122

7.13 Finding summaries that contain at least one correct ARR value (Any correct); at least one correct and no incorrect ARR values (Correct only); and all correct ARR values and no errors (Exact). A comparison of performance using partial match and exact match criteria for mentions associated with ARR values. 122

7.14 Recall, precision and F-score for ACRES and baseline system for clustering detected condition, group and outcome mentions. 125

7.15 Recall, precision and F-score for ACRES and baseline system for associating true group sizes with true group mentions and true outcome measurements with true group and outcome clusters. 125

7.16 Ceiling analysis results for condition, group and outcome summary elements when there is perfect mention and number extraction and perfect extraction followed by perfect clustering. 127

7.17 The effect of perfect performance at each stage in the system on computing correct ARR values. 127

7.18 The effect of perfect performance at each stage in the system on generating summaries with correct ARR values. 128

7.19 Outcome complementarity for the system using alternate CRF labels and the system using an ensemble approach when trained on BMJ, Cardio and BMJCardio corpora. 130

7.20 A comparison of summary element results achieved with different training sets: BMJCardio, Cardio and random subsets of 42 BMJ abstracts. This table shows recall, precision and F-score for outcome mentions, outcome summary elements and ARR values for the system without any boosting; the system using alternate CRF labels; and the system using an ensemble approach. 133

7.21 A comparison of summary results achieved with different training sets: BMJCardio, Cardio and random subsets of 42 BMJ abstracts. Recall, precision and F-score for summaries with correct ARR values for the system without any boosting; the system using alternate CRF labels; and the system using an ensemble approach. 134

7.22 Summary statistic accuracy as determined by EBM researchers. . . 135

7.23 The number of correct, qualitatively correct, incorrect, duplicate ratings for each type of summary element. Recall, precision and F-score are calculated from the ratings with qualitatively correct treated as false positives and as true positives. 140

7.24 Comparision of correct, qualitatively correct, incorrect and duplicate element ratings for the expert (R1) and the automatic evaluations performed by the system. 141

7.25 The number of summaries that each expert determined to be very helpful, somewhat helpful, not helpful, somewhat misleading or very misleading. 141

LIST OF FIGURES

Figure		Page
1.1	Sample abstract	7
1.2	Desired EBM-oriented summary	8
3.1	Main tasks to be performed in order to generate an EBM oriented summary of a medical research paper.	19
3.2	System input: abstract text	20
3.3	ACRES desired output: EBM-oriented summary	21
3.4	Overview of main processing stages in ACRES.	25
3.5	Overview of preprocessing performed before key numbers and mentions are found.	26
3.6	Overview of rule-based extraction step. This part of the system labels TIMEs, PRIMARY_OUTCOME phrases and AGE phrases.	28
3.7	Overview of classifier-based extraction step. This part of the system labels EVENT_RATEs, GROUP_SIZEs, OUTCOME_NUMBERs, GROUPs, OUTCOMEs and CONDITIONs.	28
3.8	Overview of the association stage. This part of the system is responsible for distilling extracted elements down into an EBM oriented summary.	30
4.1	Structured abstract	34
4.2	Unstructured abstract	34
5.1	The process of identifying all key trial information. Each stage performs an operation on the text and passes the updated text to the next stage.	43
5.2	Algorithm for converting numbers in word form to number form in a string of text.	44
5.3	Phrase-structure parse tree produced by the Stanford Parser for a sample sentence.	50
5.4	Dependency graph based on the collapsed typed dependencies produced by the Stanford Parser.	51
5.5	Collapsed typed dependencies produced by the Stanford Parser.	52

5.6 Algorithm for re-ranking the top-k labelings of a sentence 69

6.1 Overview of stages that take extracted elements, identify relationships between them and compile the resulting data into EBM oriented summaries. 77

6.2 Algorithm for linking outcome numbers (ON) and event rates (ER) that report the same outcome measurement for the same group. . 85

8.1 Example experimental physics abstract. 146

8.2 Desired summary for example physics abstract. 146

ABSTRACT

The practice of evidence-based medicine (EBM) encourages health professionals to make informed treatment decisions based on a careful analysis of current research. However, after caring for their patients, medical practitioners have little time to spend reading even a small fraction of the rapidly growing body of medical research literature. As a result, physicians must often rely on potentially outdated knowledge acquired in medical school. Systematic reviews of the literature exist for specific clinical questions, but these must be manually created and updated as new research is published.

Abstracts from well-written clinical research papers contain key information regarding the design and results of clinical trials. Unfortunately, the free text nature of abstracts makes it difficult for computer systems to use and time consuming for humans to read. I present a software system that reads abstracts from randomized controlled trials, extracts key clinical entities, computes the effectiveness of the proposed interventions and compiles this information into machine readable and human readable summaries.

This system uses machine learning and natural language processing techniques to extract the key clinical information describing the trial and its results. It extracts the names and sizes of treatment groups, population demographics, outcome measured in the trial and outcome results for each treatment group. Using the extracted outcome measurements, the system calculates key summary measures used by physicians when evaluating the effectiveness of treatments. It computes absolute risk reduction (ARR) and number needed to treat (NNT) values complete with confidence intervals. The extracted information and computed statistics are automatically compiled into XML and HTML summaries that describe the details and results of the trial.

Extracting the necessary information needed to calculate these measures is not trivial. While there have been various approaches to generating summaries of medical research, this work has mostly focused on extracting trial characteristics (e.g. population demographics, intervention/outcome information). No one has attempted to extract all of the information needed, nor has anyone attempted to solve many of the tasks needed to reliably calculate the summary statistics.

CHAPTER 1

INTRODUCTION

Quantitative research tests a given hypothesis and measures the results of its predictions. Abstracts that report the results of quantitative experiments contain elements describing the hypothesis and the results. However, abstracts also contain additional text to introduce the hypothesis and place the results in context. Although useful, this additional text increases the time needed to grasp the research results. Furthermore, abstract text is not convenient for information retrieval systems to index. Summaries that contain the key elements describing the hypotheses and experimental results from quantitative research papers, promise time-savings to researchers who are trying to keep up with the latest advances.

This dissertation describes a system that reads and automatically summarizes the clinical results reported in the abstracts of medical research papers. Clinical research is a quantitative science that compares the effectiveness of treatments for a given set of outcomes. Summaries generated by the system include the characteristics of the trial as well as statistics evaluating the effectiveness of the treatments involved in the study. The statistics are computed from outcome results reported in the abstract which are extracted by the system. Producing summaries of this nature is novel and requires solutions to multiple unstudied tasks.

This work addresses a problem faced by those who wish to adopt the the evidence-based medicine (EBM) paradigm. EBM refers to the practice of making treatment decisions based on a careful analysis of current research. Practitioners first construct focused treatment questions, research and analyze the evidence, make a decision, then evaluate the result. The problem is that there is a lot of research to

search through. As of November 2013, PubMed[1], the foremost database of biomedical abstracts in the world, contains over 23 million abstracts. One million were added in 2012 and the number increases every year. As a result, EBM is difficult to implement in practice. Ubbink et al. [49] report that only 52% of doctors consider their practice to be "evidence-based." Often physicians must rely on knowledge learned in medical school, which may be potentially obsolete and their experience which may be incomplete or biased.

One solution is to rely on teams of medical experts to compile *systematic reviews*, extensive reviews of the medical literature on various topics. Examples include Cochrane Collaboration[2], Evidence Based Medicine[3], ACP Journal Club[4] and BMJ Clinical Evidence[5]. Although useful, these reviews must be manually researched and continually updated as new research is published. Furthermore, these reviews are generic, aimed at the broadest set of the population possible. For this reason it may not be clear how the conclusions of the review apply to a specific patient (e.g. a 62 year old woman who is recovering from a stroke and has a family history of heart disease).

When physicians search the literature, they are encouraged to formulate focused questions so their searches return results that are relevant to their patients. The PICO query framework [40] is the commonly recommended approach for building focused queries; it contains descriptions of the patient or problem in question,

[1] `http://www.ncbi.nlm.nih.gov/pubmed`

[2] `http://www.cochrane.org`

[3] `http://ebm.bmjjournals.com`

[4] `http://www.acpjp.org`

[5] `http://www.clinicalevidence.com`

Table 1.1. Sample PICO query

Patient or Problem	Intervention	Comparison	Outcome
54 year old woman with exacerbation of periodontal disease	doxycycline	no treatment	less gum bleeding; stop recession

the intervention under consideration, a comparison intervention (when relevant), and clinical outcome(s) of interest. For example, suppose a dentist has a patient who is suffering from gum disease[6]. Normally a course of antibiotics are recommended for this condition. However, the patient is concerned about the overuse of antibiotics. The specific question that the dentist would like to research is the following.

> For a 54 year old woman with periodontal disease, what is the therapeutic efficacy of doxycycline compared to no treatment on decreasing gum bleeding and recession?

The resulting PICO query can be seen in Table 1.1. Once the PICO components are identified for a specific patient, the physician can use these to find research that is appropriate for this particular patient.

When reviewing the literature, physicians need to critically analyze the effectiveness of the treatments used in randomized studies and the significance of the results. Treatment effectiveness is captured with the summary statistics *absolute risk reduction* (ARR), which is the percentage of control patients (those with the standard treatment) who would benefit from taking the new treatment (the experimental treatment), and the *number needed to treat* (NNT) with the new treatment

[6]Example from `http://libguides.hsl.washington.edu/content.php?pid=231619&sid=1931590`

to prevent one bad outcome that would happen with the control. The significance of reported results is evaluated by examining confidence intervals for ARR and NNT values. While these statistics sometimes appear in papers, they typically do not [34], which means that physicians must calculate them. There are online tools such as the Risk Reduction Calculator [42] that will calculate these statistics, however information from the study must be manually entered into the calculator before this can happen. If a physician reviews even a few papers, calculating these statistics can become a time-consuming process. Therefore, a system that can automatically calculate these summary measures for a given paper, would help physicians evaluate the latest research more efficiently and find the best treatment options for their patients. This dissertation describes the first system to automatically extract outcome results and compute summary statistics.

1.1 Summary statistics

Absolute risk reduction (ARR) and the number needed to treat (NNT) are key measures that physicians use when searching the medical research literature for treatment strategies. They were first described by Laupacis et al[23]. ARR is the difference between the Control Event Rate (CER) and the Experiment Event Rate (EER), where the control and experiment are the control and experimental therapies that are evaluated in a randomized controlled trial (RCT). CER and EER are the rates of bad outcomes for participants in the control and experiment groups.

In order to calculate ARR for a paper, we need to identify the number of bad outcomes for the control ($N^{bad}_{control}$) and experimental treatments (N^{bad}_{exp}) along with the sizes of the treatment groups ($N_{control}$ and N_{exp}). With this information we can calculate ARR.

$$\text{ARR} = \text{CER} - \text{EER} = \frac{N_{control}^{bad}}{N_{control}} - \frac{N_{exp}^{bad}}{N_{exp}} \tag{1.1}$$

Once ARR has been calculated, we can also calculate NNT. This is the number of people that need to be given the experimental treatment in order to prevent one bad outcome. The NNT is simply the inverse of ARR, rounded up to the nearest integer.

$$\text{NNT} = \lceil 1/\text{ARR} \rceil \tag{1.2}$$

If ARR is negative, the measure is negated and it then describes the Absolute Risk Increase (ARI) of the experimental therapy. Similarly NNT becomes the number needed to harm (NNH). 95% confidence intervals for ARR are calculated using

$$\text{ARR} \pm 1.96\sqrt{\frac{\text{CER}(1-\text{CER})}{N_{control}} + \frac{\text{EER}(1-\text{EER})}{N_{exp}}}. \tag{1.3}$$

A system that automatically calculates these summary stats needs to find all the relevant information, interpret it, and then perform the ARR and NNT calculations.

1.2 EBM-oriented summaries

This dissertation presents ACRES (Automatic Clinical Result Extraction and Summarization), a system that scans an abstract, identifies the key components relevant to a PICO query and calculates summary measures for the outcomes reported in the article. Summaries contain the following elements:

- Participant information. The age ranges and common medical conditions.
- Treatment groups. The names and sizes of the treatment groups in the study.

- Outcomes. The outcomes measured in the study along with ARR calculations comparing results from the treatment groups.

These elements comprise the essential information from the clinical trial. Summaries containing just these elements require less time and effort to read the original abstract. They could also be used in physician support systems and medical information retrieval systems.

To illustrate the function of the system, consider the abstract for [16] which appears in Figure 1.1. Given this abstract, the system should generate the summary in Figure 1.2. In this abstract, all of the information needed to calculate summary statistics for the outcome *mortality* can be found in a single sentence.

> Mortality was higher in the quinine group than in the artemether group (10/52 v 6/51; relative risk 1.29 , 95% confidence interval 0.84 to 2.01)

From this sentence, the system can determine the following information.

- Outcome: *Mortality*
- Control: *quinine group*
  - Number of bad outcomes: *10*
  - Number of participants in group: *52*
- Experiment: *artemether group*
  - Number of bad outcomes: *6*
  - Number of participants in group: *51*

Although it is common for all of the information needed for calculating summary measures to appear in the same sentence, there are many papers where this is not

Rectal artemether versus intravenous quinine for the treatment of cerebral malaria in children in Uganda: randomised clinical trial.

Aceng JR, Byarugaba JS, Tumwine JK.

OBJECTIVE: To compare the efficacy and safety of rectal artemether with intravenous quinine in the treatment of cerebral malaria in children.
DESIGN: Randomised, single blind, clinical trial.
SETTING: Acute care unit at Mulago Hospital, Uganda's national referral and teaching hospital in Kampala.
PARTICIPANTS: 103 children aged 6 months to 5 years with cerebral malaria.
INTERVENTION: Patients were randomised to either intravenous quinine or rectal artemether for seven days.
MAIN OUTCOME MEASURES: Time to clearance of parasites and fever; time to regaining consciousness, starting oral intake, and sitting unaided; and adverse effects.
RESULTS: The difference in parasitological and clinical outcomes between rectal artemether and intravenous quinine did not reach significance (parasite clearance time 54.2 (SD 33.6) hours v 55.0 (SD 24.3) hours, P = 0.90; fever clearance time 33.2 (SD 21.9) hours v 24.1(SD 18.9 hours, P = 0.08; time to regaining consciousness 30.1 (SD 24.1) hours v 22.67 (SD 18.5) hours, P = 0.10; time to starting oral intake 37.9 (SD 27.0) hours v 30.3 (SD 21.1) hours, P = 0.14). Mortality was higher in the quinine group than in the artemether group (10/52 v 6/51; relative risk 1.29, 95% confidence interval 0.84 to 2.01). No serious immediate adverse effects occurred.
CONCLUSION: Rectal artemether is effective and well tolerated and could be used as treatment for cerebral malaria.

Figure 1.1. Sample abstract

the case and this information, such as group sizes, must be gathered from multiple sentences.

1.3 Generating summaries

There are many tasks that must be performed in order to generate EBM-oriented summaries for a given article. ACRES identifies the parts of the text that refer to entities such as treatment groups and outcomes (*mentions*) and *numbers* such as the sizes of the treatment groups, the number of good or bad outcomes and the outcome event rates. It finds relationships between the detected mentions and numbers and determines what can be calculated from the detected data. While there has been previous work aimed at finding some of the mention types used by the system, most of the tasks performed by the system have not been studied.

Title: Rectal artemether versus intravenous quinine for the treatment of cerebral malaria in children in Uganda: randomised clinical trial
Age:

- **min:** 6 months
- **max:** 5 years

Condition: cerebral malaria
Groups:

- rectal artemether
- intravenous quinine

Outcomes:

- mortality
  More effective: artemether group, 11.8% (6/51)
  Less effective: quinine group, 19.2% (10/52)
  ARR: 7.4%, 95% confidence interval [-6.5%, 21.3%]
  NNT: 14, 95% confidence interval [5, ∞]
- parasite clearance time
- fever clearance time
- time to regaining consciousness
- time to starting oral intake
- adverse effects

Figure 1.2. Desired EBM-oriented summary

1.4 Claims

In this thesis I set out to prove the following claims:

1. It is possible to develop a system that can read an abstract and automatically generate EBM-oriented summaries that include automatically calculated summary statistics.
   (a) Finding all of the relevant information such a summary is possible.
   (b) Correctly interpreting this information is possible.
   (c) Computing summary statistics with high precision is possible.
2. A system that produces EBM-oriented summaries is useful to physicians.

I support these claims by first describing similar projects that have attempted to summarize or identify key information in medical research papers. Then I describe the novel approaches that I use to extract and interpret the key information needed to calculate summary measures. Finally, I provide results demonstrating that ACRES is able to automatically calculate summary measures with reasonable precision.

1.5 Application to other domains

The ACRES framework for summarizing clinical abstracts consists of a sequence of methods to extract the key elements describing the clinical trial and its results; identify the relationships that exist between to the elements; and fill slots in a summary template to produce a summary that describes the experiment and its results. Although the focus of this work is summarizing clinical abstracts, the summarization framework employed in ACRES may be applied to abstracts from other types of quantitative research. In Chapter 8, I describe how to adapt the ACRES framework to other domains and provide an example from experimental physics.

1.6 Contributions

This thesis describes the first known system that can read an abstract, extract the clinical results and calculate summary measures from the extracted information. Building such a system required solutions to several novel problems. As a result, this thesis work makes the following contributions.

1. New data sets. The first corpora of randomized controlled trial abstracts containing annotations for conditions, population age values, treatment groups, group sizes, outcome descriptions, number of good or bad outcomes, and outcome event rates.

2. Novel approach for extracting population age information.

3. Novel approach for extracting condition, group and outcome mentions that leverages alternate conditional random field labelings.

4. The first approach for extracting group size, outcome number and event rate values.

5. The first approach for associating group and outcome mentions with group size, outcome number and event rates.

6. The first approach for calculating ARR and NNT from automatically extracted information.

CHAPTER 2

BACKGROUND AND PRIOR WORK

This chapter describes the various systems and resources that have been developed to help physicians find answers to their clinical questions. Some of these resources may be used directly by physicians, such as PubMed, other resources, such as MetaMap, are used to construct question answering systems for physicians.

2.1 Online trial registries

Since 2008, clinical trials that compare interventions regulated by the U.S. FDA must be registered on clinicaltrials.gov. The registry entries contain eligibility criteria, intervention and outcome details for the trial, in XML form. However, there are many studies, especially older trials and those conducted outside the U.S., that are not registered. Furthermore, outcome results are rarely posted once the study has been completed. As of Nov 2013, only 10,267 of 106,426 (10%) closed trials have results posted. Hence, results for many studies are only available in publications and natural language processing solutions are needed to extract the key trial information from the text.

2.2 PICO queries

When physicians begin their investigation of treatment options for a patient, they are encouraged to identify four pieces of information that will characterize their search[40]. These four information elements are:

- Patient or problem: Characteristics that describe the patient (e.g. sex or age range) and their current physical condition.
- Intervention: The proposed treatment for the patient. This may be a drug, surgical procedure, or even a nonstandard medical activity such as playing the

didgeridoo or swimming with dolphins.

- Comparison: The standard or control treatment for such a patient.
- Outcome: The result that the treatment is supposed to affect in someway. This may be something that they want to happen (*good outcome*) such as recovering from a disease or something that they do not want to happen (*bad outcome*) such as death.

A query that includes some or all of these elements is often referred to as a PICO query. Although alternative structures have been proposed[10], the PICO structure still appears to be the standard.

2.3 PubMed and MEDLINE

A common resource for both physicians and those developing systems to help them in their search for treatment options is PubMed[7]. It is a web site that allows users to search MEDLINE[8], the National Library of Medicine's (NLM) database of abstracts and citations in the fields of biology and medicine, as well as its own database of articles that fall outside the scope of MEDLINE or have not yet been indexed by MEDLINE. Records in MEDLINE are indexed using NLM's Medical Subject Headings (MeSH)[9] thesaurus. A set of MeSH terms are associated with each citation.

2.4 UMLS Metathesaurus

The Unified Medical Language System (UMLS)[26] is a collection of resources

[7]http://www.ncbi.nlm.nih.gov/pubmed/

[8]http://www.nlm.nih.gov/pubs/factsheets/medline.html

[9]http://www.nlm.nih.gov/mesh/

developed by NLM to help people create computer systems that act as if they "understand" the language of biomedicine and health. UMLS includes three databases: the Metathesaurus, the Semantic Network, and the SPECIALIST Lexicon. The Metathesuarus is a database of biomedical terms, their alternate versions, and the relationships between the terms. The Semantic Network consists of a set of subject categories (Semantic types) and a set of relationships that exists between the types. The SPECIALIST Lexicon is a collection of medical terms and common English words.

The Metathesaurus includes terms from many existing collections of terms such as MeSH and the Systematized Nomenclature of Medicine - Clinical Terms (SNOMED CT). SNOMED CT is a collection of terms used in health and health care. It is currently maintained by The International Health Terminology Standards Development Organization (IHTSDO). Each term is associated with a concept code and there may be multiple terms associated with the same code if they are alternate ways of referring to the same concept. SNOMED CT also defines relationships that exist between terms. Relationships include hyponym-hypernym (e.g. tuberculous pneumonia is a kind of lung infection) and causal (e.g. tuberculous pneumonia is caused by mycobacterium tuberculosis).

MetaMap[1] is a tool that identifies segments of text that correspond to concepts in the UMLS Metathesaurus. It parses a given sentence into noun phrases and finds the Metathesaurus concept that best matches each noun phrase. MetaMap was originally developed for help find relevant MEDLINE citations for a given query.

2.5 Clinical question answering systems

Various intelligent software-based solutions have been developed as an alternative to manually searching the literature or relying on human experts to summarize the literature. Clinical question answering systems automate the search process for

the user. They take a PICO query as input and look for studies that contain text segments (phrases or entire sentences) which match elements in a user's query. The resulting matches are returned for the user to review.

Niu et al.[33][32] describe a system that is part of the EpoCare project (Evidence at Point of Care) whose goal to develop fully automatic methods for answering clinical queries by searching Clinical Evidence[10], a journal of manually compiled, systematic reviews published by BMJ. Their system takes a PICO query and retrieves text segments that contain all of the elements from the query.

The medical digital library PERSIVAL [29] uses information from patient records to re-rank search results, giving preference to articles that are the best match for individual patients. Profiles for patient records and articles are built using finite state grammars which extract noun phrase describing medical terms and any related values. Textual summaries of the top search results are created by combining key phrases extracted from the abstract with pre-written slotted sentences.

Demner-Fushman and Lin[12] present a system that takes a PICO query and retrieves a list of MEDLINE citations from PubMed, which their system ranks according their relevance, and forms an answer to the query from these citations. Their system scores the relevance of citations by first applying various knowledge extractors to the abstracts of each citation. The knowledge extractors search for PICO elements. They extract short phrases for patient, population, and interventions. For outcomes, they extract complete sentences containing the outcome. The outcome extractor finds the most likely outcome sentences by applying an ensemble of classifiers to each sentence in the abstract and combining the results. The knowledge extractors rely heavily on MetaMap. A citation is scored based on how well extracted PICO

[10] `http://clinicalevidence.bmj.com`

elements match the original PICO query. The authors also calculate a relevance score which is based on the journal that article appears in, type of study, and date of publication. Finally, their system calculates a task score, which is based on the presence of MeSH terms that indicate certain clinical tasks (therapy tasks, diagnosis tasks, prognosis tasks, and etiology tasks). These three citation scores are then combined to determine the overall relevance of a citation. The citations are re-ranked by relevance and the final answer to the original query consists of the title and top three outcome sentences for each of the top citations.

The question answering system askHERMES takes a natural language question as input, then retrieves and summarizes relevant passages from multiple online sources including PubMed and Wikipedia. It uses a novel scoring measure to identify passages that best match the original query [5].

Although question answering systems look for text segments containing PICO elements, none of the existing methods extract outcome results, nor do they compute summary measures that can be used to compare the effectiveness of proposed treatments.

2.6 Semantic MEDLINE

Semantic MEDLINE is a tool that visualizes the clinical entity relationships found in MEDLINE citations retrieved by queries on a given topic. It uses the semantic processor SemRep to process the title and abstract text to identify clinically relevant *predictions*, relationships such as "Aspirin-TREATS-Headache," and automatically identify the point-of-view focus of the text (e.g. treatment of disease or interaction of substances) [14][50].

2.7 Semi-automatic creation of clinical trial databases

Databases of clinical trials promise more efficient searching of past trials than manually searching databases of abstracts. Since voluntary entry of trial results in existing databases such as ClinicalTrials.gov is inconsistent, tools for automatically constructing trial databases from published reports are desirable. ExaCT is a tool to help human reviewers compile a database of clinical trials and their characteristics. It was developed by Kiritchenko et al. [19] and it builds on earlier work by Sim et al. [44] and de Bruijn et al. [11]. The system automatically searches an clinical journal article for text fragments that best describe the trial characteristics. A human reviewer assesses and modifies the suggested selections. The information found by ExaCT consists of 21 different elements that describe the trial participants, the interventions assigned to them, the outcomes measured in the trial, and information about the article (e.g. authors, data of publication). However, this system does not attempt to extract the number of good or bad outcomes, nor does it try to calculate any summary statistics. ExaCT uses a sentence classifier to first find sentences most likely to contain desired information elements. Elements are then extracted from the candidate sentences using hand-crafted rules.

2.8 Contributions

Machine readable summaries that describe trial results and quantify the effectiveness of proposed treatments have multiple uses. They can populate databases of clinical trials. Literature searches can retrieve and filter articles based on their summary elements; articles may be ranked by the significance of the results. Summaries are an efficient alternative to abstracts for reviewing results from a collection of studies.

ACRES is the first system to generate machine readable summaries from ab-

stract text that contain essential trial characteristics, outcome results and computed summary measures. These measures quantify the effectiveness of proposed treatments, a key step in the EBM paradigm. Correctly identifying and interpreting all of the information needed to calculate *absolute risk reduction* and *number needed to treat* values challenging and has not been previously attempted.

CHAPTER 3

SYSTEM ARCHITECTURE

This chapter provides an overview of ACRES (Automatic Clinical Result Extraction and Summarization), the summarization system presented in this document. It describes the function of each component and how they all work together to summarize a given abstract.

3.1 Problem overview

This section provides an overview of the tasks that must be solved in order to generate EBM-oriented summaries that include automatically calculated summary measures. A diagram of the main tasks involved in generating a summary for a given paper can be seen in Figure 3.1. ACRES takes an abstract text as input and produces a machine readable XML summary as output. Figures 3.2 and 3.3 give an example of an abstract and its desired summary (formatted without XML tags for easier reading). A complete description of the XML summary format is given in Appendix B.

3.1.1 Extracting key elements. In order to create informative summaries of abstracts that describe the results of clinical trials, the system needs to first identify the text segments that refer to the key elements of the trial. The system needs to identify text that describes following elements:

- treatment groups
- outcomes
- population age information
- conditions common to all study participants
- group sizes

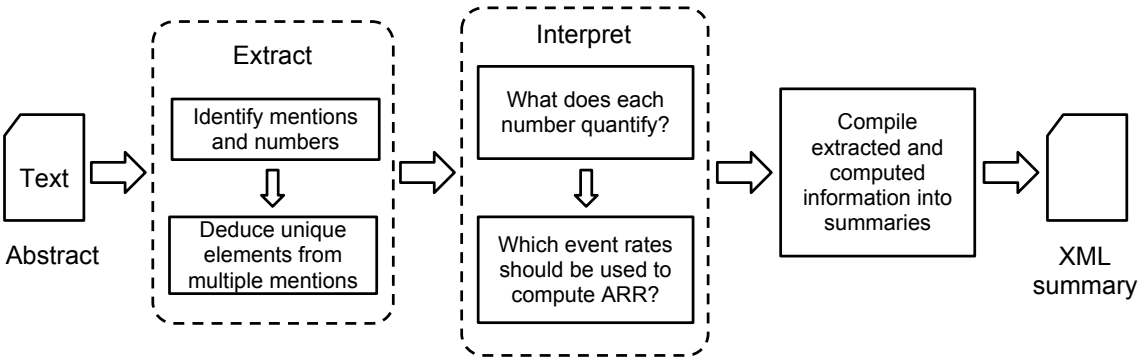

Figure 3.1. Main tasks to be performed in order to generate an EBM oriented summary of a medical research paper.

- number of good or bad outcome
- outcome event rates

Typically, there are multiple textual references to the same treatment group, outcome or condition entity in an abstract. For example the abstract in Figure 3.2 contains multiple references to the quinine treatment group. I refer to each individual reference as a *mention*. In addition to identifying mentions, the system needs to recognize when multiple mentions refer to the same *unique entity*. Again, the quinine group is referred to as both "the quinine group" and "intravenous quinine."

Group mentions are the names of the treatment groups involved in the clinical trial that is documented by the article. Their names often consist of the name of the treatment that they are given followed by the word "group" or "arm" (e.g. "quinine group" or "placebo arm"). However, there are many cases where these terms are omitted and the treatment groups are referred to implicitly using only the treatment name, such as in Figure 3.2 which contains several references to "intravenous quinine" and "rectal artemether". Furthermore, there are many cases where the treatment group is referred to using names that describe their *role* in the trial (e.g. "control group" or "intervention group") and do not contain a description of the treatment.

Rectal artemether versus intravenous quinine for the treatment of cerebral malaria in children in Uganda: randomised clinical trial.

Aceng JR, Byarugaba JS, Tumwine JK.

OBJECTIVE: To compare the efficacy and safety of rectal artemether with intravenous quinine in the treatment of cerebral malaria in children.
DESIGN: Randomised, single blind, clinical trial.
SETTING: Acute care unit at Mulago Hospital, Uganda's national referral and teaching hospital in Kampala.
PARTICIPANTS: 103 children aged 6 months to 5 years with cerebral malaria.
INTERVENTION: Patients were randomised to either intravenous quinine or rectal artemether for seven days.
MAIN OUTCOME MEASURES: Time to clearance of parasites and fever; time to regaining consciousness, starting oral intake, and sitting unaided; and adverse effects.
RESULTS: The difference in parasitological and clinical outcomes between rectal artemether and intravenous quinine did not reach significance (parasite clearance time 54.2 (SD 33.6) hours v 55.0 (SD 24.3) hours, P = 0.90; fever clearance time 33.2 (SD 21.9) hours v 24.1(SD 18.9 hours, P = 0.08; time to regaining consciousness 30.1 (SD 24.1) hours v 22.67 (SD 18.5) hours, P = 0.10; time to starting oral intake 37.9 (SD 27.0) hours v 30.3 (SD 21.1) hours, P = 0.14). Mortality was higher in the quinine group than in the artemether group (10/52 v 6/51; relative risk 1.29, 95% confidence interval 0.84 to 2.01). No serious immediate adverse effects occurred.
CONCLUSION: Rectal artemether is effective and well tolerated and could be used as treatment for cerebral malaria.

Figure 3.2. System input: abstract text

Title: Rectal artemether versus intravenous quinine for the treatment of cerebral malaria in children in Uganda: randomised clinical trial
Age:

- **min:** 6 months
- **max:** 5 years

Condition: cerebral malaria
Groups:

- rectal artemether
- intravenous quinine

Outcomes:

- mortality
  More effective: artemether group, 11.8% (6/51)
  Less effective: quinine group, 19.2% (10/52)
  ARR: 7.4%, 95% confidence interval [-6.5%, 21.3%]
  NNT: 14, 95% confidence interval [5, ∞]
- parasite clearance time
- fever clearance time
- time to regaining consciousness
- time to starting oral intake
- adverse effects

Figure 3.3. ACRES desired output: EBM-oriented summary

In these cases, the system must make connections between the treatments mentioned in the paper and the more generic control or experimental group name references that may be used when reporting the results. This task can be challenging since the paper may not always state which treatment is the experiment or control.

Outcome mentions describe an event or condition that the experimental treatment is supposed to affect for each person in the trial. For example, "mortality" and "parasite clearance time" are both outcomes mentioned in Figure 3.2. Outcomes may be considered *good* or *bad*. Good outcomes are something that the experimental treatment should increase such as quitting smoking, recovering from or not developing a disease. Bad outcomes are events that the treatment should reduce such as mortality or developing a disease. Besides finding outcome mentions, the system must also determine each mention's *polarity*, whether the mention is good or bad. This is important for determining whether the related outcome number is the number of good outcomes or bad outcomes for a group, which affects the calculation of summary statistics.

Population demographics are the sections of text that describe some common aspect of the subjects involved in a study (e.g. children under 5 years, women over 50, or people with diabetes). While this information is not necessary for calculating summary statistics, it is important for helping physicians determine whether certain studies are relevant for their patients. The demographic information currently extracted by the population age statistics and common medical conditions that describe a participants eligibility for the trial. The *age* values that the system looks for are *minimum*, *maximum*, *mean* and *median* population ages. In Figure 3.2, the minimum age is "6 months" and the maximum age is "5 years." *Condition* mentions describe common characteristics of the trial population (e.g. "patients with acute myocardial infarction", "people needing rehabilitation" or "with type 1 or type 2

diabetes but no symptomatic cardiovascular disease"). In Figure 3.2, the condition common to all patients is "cerebral malaria." Distinguishing between conditions and outcomes is a challenge as there is often overlap between the two. In our example, the patients all have cerebral malaria and one of the outcomes is clearance of the malaria parasites.

A *group size* is the number of people in a particular treatment group. In our example, there are 52 subjects in the quinine group and 51 subjects in the artemether group. Group sizes are important for calculating outcome event rates for a group. Besides simply identifying a value as a group size, the system must identify the correct size of the group when an outcome is measured. This is not trivial. A study may report multiple sizes for the same group. Some participants may drop out of the study before outcomes are measured. These are referred to as the number *lost to follow-up*. Sometimes this value is explicitly mentioned. Other times, those lost are simply reflected in a smaller group size reported with the outcome number for the group at a given follow-up time.

An *outcome number* is the number of people in a group who experienced a good or bad outcome. These are recorded at various follow-up times in a trial. Again, in our example the number of people who died (mortality) in the quinine group is 10 and the number who died in the artemether group is 51. An outcome *event rate* is the percentage of people in a treatment group that achieve a certain outcome. In some abstracts the event rates are explicitly reported for outcomes in addition to, or instead of, the number of people who experienced the outcomes.

While there has been previous work directed at finding some of these mention types, in order to correctly interpret the detected values used to calculate summary statistics, this system needs a more effective approach than the current state of the art,

especially for treatment groups and outcomes. Furthermore, the task of identifying the polarity of the outcomes is currently unstudied.

Unlike the task of mention finding, correctly identifying the quantities that we need to calculate summary statistics is largely unstudied. Furthermore, in some papers this information appears only in figures or tables that must be interpreted.

3.1.2 Interpreting values. In order to calculate summary measures from the detected mentions and quantities, the system must interpret the values that it has found. For a given group size, the system must identify the group it describes and follow-up time for when the size was recorded. Furthermore the system must determine if any people in the group have been lost by this follow-up time. Given an outcome number or event rate, the system needs to identify the outcome it measures, whether the outcome is good or bad, which group the number was recorded for, and the follow-up time for when the outcome was measured. The system also needs to cluster the mentions that refer to the same group or outcome into sets. This allows related information found in different sentences to be linked and it identifies the unique entities discussed in the paper. After the relationships between mentions and quantities have been established, the system must then pair the outcome results from each treatment group, measured at the same follow-up time and then calculate the summary statistics for best and worst case scenarios. All of these tasks are unstudied.

3.1.3 Compiling the final summary. After all of the necessary information has been identified and the summary measures have been computed, this information needs to compiled into a summary.

3.2 Overview of ACRES

Figure 3.4 provides a high-level overview of ACRES. The process of summarizing an abstract consists of three main phases. First the text is preprocessed to

add grammatical and semantic information to the text. Then, the system identifies sequences of words and numbers that describe key elements of the trial (e.g. group names, outcome event rates). Finally, the system interprets what it has found, computes summary measures and compiles this information into XML summaries.

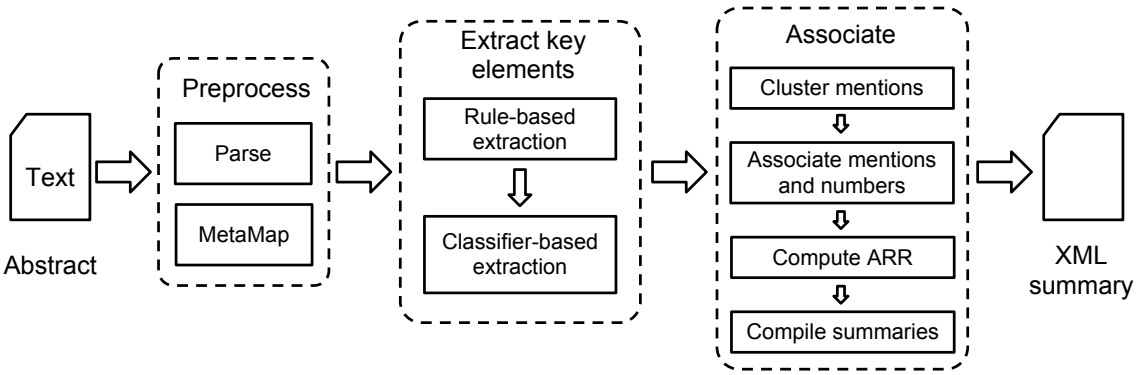

Figure 3.4. Overview of main processing stages in ACRES.

3.3 Pre-processing

Figure 3.5 provides an overview of the preprocessing stage. A detailed description of the steps in this stage is given in Section 5.1. The purpose of the pre-processing stage is to augment the text with grammatical and semantic information. Later stages use this information to extract and interpret key clinical elements from the text.

The first pre-processing step is to normalize the text before it is tokenized and parsed. Normalization consists of identifying common key phrases and grammatical structures such as "95% confidence interval" and replacing them with simpler terms and structures. This substitution is performed to prevent parsing errors and reduce variability in the resulting parse trees.

After normalization, the text is parsed to determine the grammatical structure of each sentence and identify dependency relationships between the tokens in the

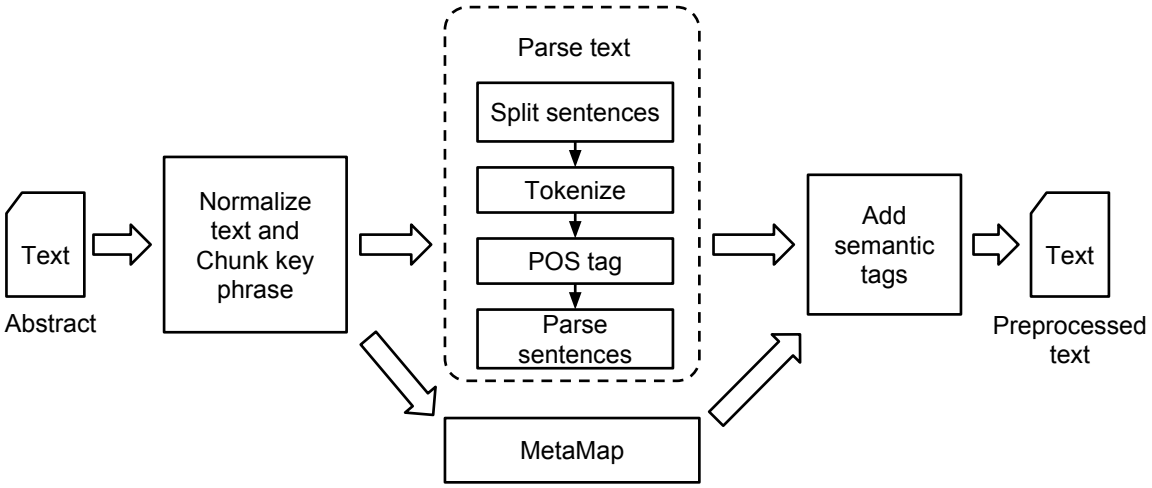

Figure 3.5. Overview of preprocessing performed before key numbers and mentions are found.

sentence. Parsing itself is a multi-stage process whereby the text is split into sentences; each sentence is tokenized; part-of-speech (POS) tags are assigned to each token; and parse trees are constructed for each sentence.

To recognize biomedical terms and phrases, MetaMap is applied to the text. MetaMap performs its own parsing of the text, so it does not use the output from the parser. However, MetaMap only performs shallow parsing. It does not produce phrase structure or dependency parse trees, both of which are used by the summarization system.

The final pre-processing step is to add semantic tags to words that appears in word lists that define certain classes of words such as *TIME*, *POPULATION* or *MEASUREMENT*.

3.4 Extract key elements

After pre-processing, the system examines the text to identify all of the information needed to produce an EBM oriented summary. A detailed explanation of the

extraction process is given in Chapter 5.

The process of extracting key elements progresses in several stages. First, as shown in Figure 3.6, the system applies a collection of high precision rules to identify certain commonly occurring, easily recognizable numeric values such as measurements and intervals. After this, rule-based extractors are applied to the text to identify *time* phrases, *primary outcome* phrases, and *age* phrases. These phrases are relatively easy to identify and while they are not used directly in the summary, they are used in later stages of the system to identify and interpret elements that will appear in the summary.

As seen in Figure 3.7, after rule-based extractors are applied, a collection of trained classifiers is applied to the text. Each classifier is trained to label words in a sentence as belonging to a particular entity type (e.g. group, outcome, event rate) or not. For condition, group and outcome entity types, consecutive entity tokens are grouped together and considered to be a *mention* of that type.

The system uses a classifier that is capable of providing alternate token labelings for a sentence. These alternate labelings are used to find a more accurate labeling than the one originally assigned by the classifier. After all of the entity token classifiers have been applied to a sentence, the alternate labelings are used to rerank the output from the outcome classifier.

3.5 Associate elements

After the system identifies the segments of the text that correspond to the key element types needed for a summary, there are various associations between the elements that must be identified. Detailed descriptions of the steps in this process is found in Chapter 6.

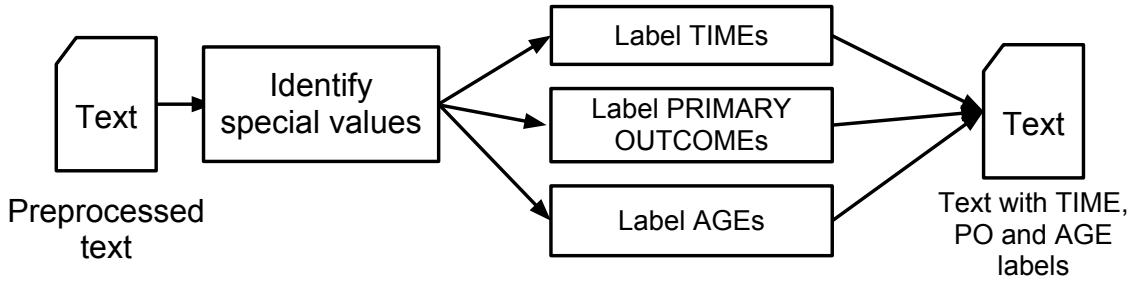

Figure 3.6. Overview of rule-based extraction step. This part of the system labels TIMEs, PRIMARY_OUTCOME phrases and AGE phrases.

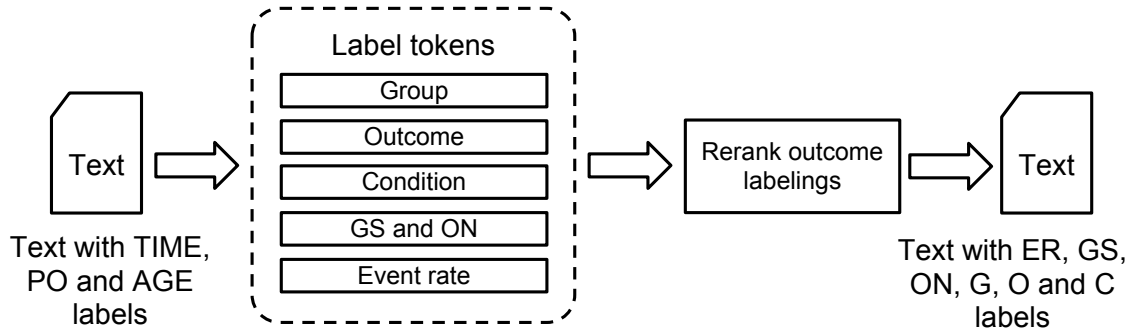

Figure 3.7. Overview of classifier-based extraction step. This part of the system labels EVENT_RATEs, GROUP_SIZEs, OUTCOME_NUMBERs, GROUPs, OUTCOMEs and CONDITIONs.

As shown in Figure 3.8, the system identifies the condition, group and outcome mentions that refer to the same unique condition, group or outcome entity. Mentions that refer to the same unique entity are grouped into clusters. Clustering mentions reduces redundancy in the summary and allows information found in an earlier sentence to be used with information found in a later sentence. For instance, a group's size is sometimes mentioned in an earlier sentence and the number of bad outcomes may appear in a later sentence. If both values are associated with the same group cluster, then they can be combined to compute an outcome event rate for that group cluster.

Once the entity clusters have been determined, the system identifies the relationships that exist between the detected numbers and the group and outcome mentions that appear in the same sentence. These associations are needed in order to compute summary statistics for each outcome. The numbers themselves are useless if we do not know the outcome and group to which they pertain. The first step is to identify the group to which each group size belongs. A classifier-based approach is used to perform this association. Next, a rule-based approach is employed to identify the outcome numbers and event rates that describe the same *outcome measurement* for a group. An outcome measurement is either the number of good or bad outcomes for a group or the outcome event rate given in the text. Either or both may be present in a text. After identifying the straightforward cases where outcome numbers and event rates describe the same outcome measurements, the next step is to associate groups and outcomes with outcome measurements. For each sentence, the system computes the probability for each potential association of group, outcome and outcome measurement. A matching algorithm is used to find the optimal assignments of outcome measurements to (group, outcome) pairs, such that the sum of the probabilities of each (group, outcome, outcome measurement) association is maximized.

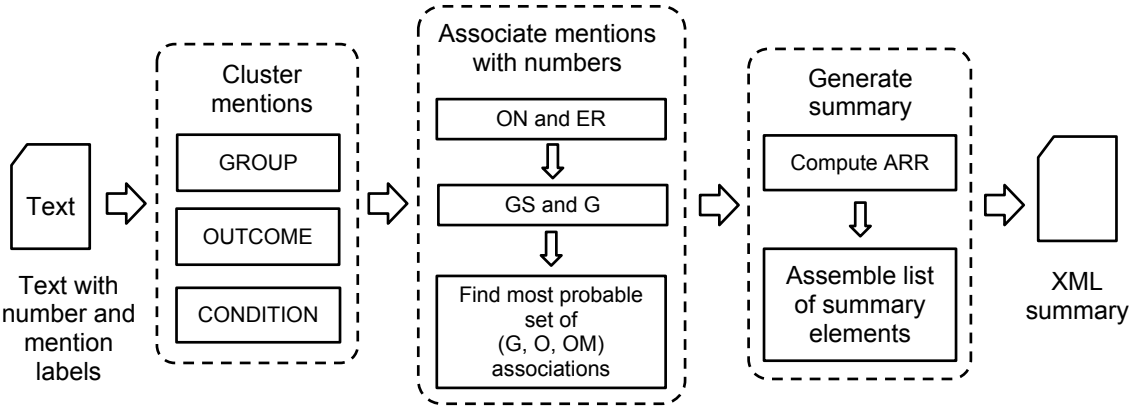

Figure 3.8. Overview of the association stage. This part of the system is responsible for distilling extracted elements down into an EBM oriented summary.

A classifier-based approach is used to estimate the probabilities for each potential association.

Finally, the system computes summary statistics (ARR) from outcome measures for the same outcome, but different groups. It generates summaries in XML format from the lists of detected age values, conditions, groups, outcomes and computed summary measures.

CHAPTER 4

DATA

In order to train and test ACRES, it is necessary to have a corpus of medical texts where all of the necessary information has been identified. Since no such corpus existed, it was necessary to create one. Over the course of this thesis work multiple corpora were created with different characteristics. This chapter describes the construction of each corpus and compares their characteristics. The creation of corpora containing all annotations needed for training a system to produce summaries with summary statistics is one of the novel contributions of this thesis work.

4.1 Corpora annotations

The following elements were annotated in the corpora used by the system.

- *Groups*: The names of the groups who are assigned a particular type of treatment. Group names usually include the name of the treatment assigned to the group (e.g. *quinine group* or *artemether group*).
- *Outcomes*: The names of outcomes that are measured in the paper. Whether the outcome is *good* (something the treatment should improve) or *bad* (something the treatment should prevent or decrease) was also annotated.
- *Conditions*: Medical conditions that are common to all participants in the trial.
- *Age values*: Values that describe the minimum, maximum, median or average age of the trial participants.
- *Group sizes*: The number of people in a treatment group.
- *Outcome numbers*: The number of good or bad outcomes measured for a particular group at a given follow-up time.

- *Event rates*: The percentage of people in a treatment group that experienced a good or bad outcome.

A decision that must be made when analyzing medical research papers is whether to annotate and process only the abstract, as is commonly done, or the full text of the paper. For the summarization system described in this document, the decision was made to focus on abstracts only. The primary reason for this was to maximize annotation effort. Abstracts have several advantages, they are shorter and take less time to annotate than full text, while still containing much of the important information of a paper. They are also publicly available in text or XML format through PubMed[11], whereas full text is not always freely accessible and is often available only in PDF or HTML which are more difficult to automatically process.

Unfortunately, the information that we need to calculate summary statistics is not always in abstracts. In a random sample of 54 BMJ (British Medical Journal) articles, it was possible to calculate summary statistics for 30 (56%) papers. Of these 30 papers, 13 contained all needed information in the abstract, 11 required the full text to be examined, and for 6 it was necessary to examine tables to find all of the necessary information. This shows that while abstracts are a good source of clinical results, there is benefit to be gained from expanding the scope of the summarization system to include full text and tables in the future.

4.2 Structured vs. Unstructured abstracts

In order to communicate the results of paper more efficiently, some journals require abstracts to conform to a *structured* format with section labels such as can be seen in Figure 4.1. Typical sections include OBJECTIVE, DESIGN, PARTICI-

[11] `http://www.ncbi.nlm.nih.gov/pubmed/`

PANTS, INTERVENTION, OUTCOME MEASURES, RESULTS, CONCLUSION. The text in each section may vary from sentence fragments to multiple complete sentences.

While structured abstracts may be required by publishers such as BMJ, other journals do not have such a requirement. Hence, some abstracts have a more traditional *unstructured* format such as the one shown in Figure 4.2.

The existence of both structured and unstructured abstracts presents a challenge. As can be seen when comparing Figures 4.1 and 4.2, there are significant grammatical and stylistic differences between the two types of abstracts. The section labels in structured abstracts provide useful cues to the summarization system, but the fragments can be problematic to process. An entire fragment (or a subset of it) may be a detailed description of a single entity or multiple entities. Furthermore the number of sections in a structured abstract may vary with some abstracts only containing a few sections such as BACKGROUND, METHODS, FINDINGS and INTERPRETATION. Unstructured abstracts primarily consist of complete sentences, but they do not provide semantic cues in the form of section labels.

4.3 Corpora construction

Due to the time consuming nature of manually annotating occurrences of all key elements in an abstract, construction of the corpora used by the system progressed in several stages.

For my initial work on finding mentions and quantities I created a corpus of 100 BMJ abstracts obtained through PubMed. BMJ abstracts were specifically targeted because the full text for all the articles are freely available online in html format (including tables) which is easier to work with than PDF. The abstracts in this corpus are from the first 100 randomized controlled trials electronically published in 2005 and

Rectal artemether versus intravenous quinine for the treatment of cerebral malaria in children in Uganda: randomised clinical trial.

Aceng JR, Byarugaba JS, Tumwine JK.

OBJECTIVE: To compare the efficacy and safety of rectal artemether with intravenous quinine in the treatment of cerebral malaria in children.
DESIGN: Randomised, single blind, clinical trial.
SETTING: Acute care unit at Mulago Hospital, Uganda's national referral and teaching hospital in Kampala.
PARTICIPANTS: 103 children aged 6 months to 5 years with cerebral malaria.
INTERVENTION: Patients were randomised to either intravenous quinine or rectal artemether for seven days.
MAIN OUTCOME MEASURES: Time to clearance of parasites and fever; time to regaining consciousness, starting oral intake, and sitting unaided; and adverse effects.
RESULTS: The difference in parasitological and clinical outcomes between rectal artemether and intravenous quinine did not reach significance (parasite clearance time 54.2 (SD 33.6) hours v 55.0 (SD 24.3) hours, P = 0.90; fever clearance time 33.2 (SD 21.9) hours v 24.1(SD 18.9 hours, P = 0.08; time to regaining consciousness 30.1 (SD 24.1) hours v 22.67 (SD 18.5) hours, P = 0.10; time to starting oral intake 37.9 (SD 27.0) hours v 30.3 (SD 21.1) hours, P = 0.14). Mortality was higher in the quinine group than in the artemether group (10/52 v 6/51; relative risk 1.29, 95% confidence interval 0.84 to 2.01). No serious immediate adverse effects occurred.
CONCLUSION: Rectal artemether is effective and well tolerated and could be used as treatment for cerebral malaria.

Figure 4.1. Structured abstract

Usefulness of intravascular low-power laser illumination in preventing restenosis after percutaneous coronary intervention.

Derkacz A, Protasiewicz M, Poreba R, Szuba A, Andrzejak R.

Despite the several years of studies, no factor that could reduce the restenosis rate without significant limitations has been introduced. The aim of the present study was to evaluate the influence of low-power 808-nm laser illumination of coronary vessels after percutaneous angioplasty in preventing restenosis. The procedure of laser intravascular illumination was performed on 52 patients (laser group), and another 49 patients formed the control group. All patients were monitored for major adverse cardiac events (MACE) at the 6- and 12-month follow-up points. The MACE rate after 6 and 12 months was 7.7% in the laser group at both points. The MACE rate was 14.3% and 18.5% at 6 and 12 months of follow-up in the control group, respectively (p = NS). Follow-up coronary angiography was performed after 6 months. The difference in the restenosis rate was insignificant (15.0% vs 32.4%); however, significant differences were observed in the minimal lumen diameter (2.18 $\pm$ 0.70 vs 1.76 $\pm$ 0.74 mm; $p < 0.05$), late lumen loss (0.53 $\pm$ 0.68 vs 0.76 $\pm$ 0.76 mm; $p < 0.01$), and the late lumen loss index (0.28 $\pm$ 0.39 vs 0.46 $\pm$ 0.43; $p < 0.005$) in favor of the laser group. In conclusion, the new therapy seemed effective and safe. Marked differences between late loss, late loss index, and minimal lumen diameter were observed. The late lumen loss in the laser group was only slightly greater than that in studies of drug-eluting stents, and MACE rate remained within very comparable ranges. This suggests that intravascular laser illumination could bring advantages comparable to those of drug-eluting stents without the risk of late thrombosis.

Figure 4.2. Unstructured abstract

2006. Articles which did not appear to be evaluating treatments were ignored. The annotated mentions were verified by an EBM researcher at the University of Illinois at Chicago medical school. The corpus was later expanded with the help of a masters student at IIT to include additional, more recent BMJ abstracts. Now the corpus consists of 188 BMJ abstracts from randomized controlled trials published from 2005 to 2009. Since it contains only abstracts from BMJ journals, this corpus is referred to in this document as the *BMJ* corpus.

The scope of the trials described by the abstracts in the BMJ corpus is rather broad, covering everything from drug trials for malaria to didgeridoo playing to alleviate sleep apnea. The *Cardio* and *Ischemia* corpora were created to examine the system's effectiveness on set abstracts for a single topic. The Cardio corpus is a set of 42 abstracts from different journals obtained using the PubMed query "cardiovascular disease." The Ischemia corpus is a collection of 117 abstracts from various journals obtained using the query "myocardial ischemia." Only articles describing primary analysis of randomized controlled trials that compare the clinical effectiveness of two or more treatments were annotated. In addition we excluded abstracts reporting results for subgroups as the system does not support them at this time. Since the extraction of key values and the computation of absolute risk reduction (ARR) statistics was the focus, abstracts were excluded if they did not contain at least one group size, outcome number or event rate.

4.4 Corpora characteristics

Due to differences in selection criteria for each corpus, the corpora exhibit some differences. Tables 4.1 - 4.5 describe the various characteristics of each corpus.

Table 4.1 provides a basic description of each corpus. BMJ is the largest corpus, containing 188 abstracts, whereas Cardio is the smallest, containing only 42.

The size differences reflect time constraints and result from the differing motivations behind their construction. BMJ was constructed first as the system was developed. Cardio and Ischemia were created later to focus on trials related to a single topic. Cardio was created before Ischemia to augment the BMJ corpus in further system development. Ischemia was created last and was held out as a final test set. Beyond size, there are other key differences between the corpora. All BMJ abstracts are structured, since this is a common feature of BMJ journals. Since Cardio and Ischemia abstracts come from various journals, they contain unstructured abstracts. This leads to differences in average sentence lengths, since structured abstracts contain significantly more sentence fragments than unstructured ones. Since it was part of the selection criteria for Cardio and Ischemia, all of their abstracts contain key values needed for computing ARR values, whereas only 76% of BMJ abstracts contain at least one group size, outcome number or event rate. Finally, due to the more technical nature of the papers in Cardio and Ischemia, they contain significantly more acronyms occurrences than BMJ. Acronyms create additional challenges. The system needs to recognize them and connect them with their expanded form in the text.

Table 4.2 gives the number of value annotations for each corpora. The corpora differ with respect to the number of annotated outcome numbers and event rates. Abstracts containing outcome numbers were specifically targeted for the Cardio corpus. Early versions of the summarization system did not extract event rates directly from the text. It computed event rates from outcome numbers and group sizes reported in the text. Event rates that appear in papers are often rounded. As a result, ARR values computed from outcome numbers and group sizes can be more precise than those computed from the event rates reported in the text. However, it is more common for event rates to appear in abstracts than outcome numbers and more ARR values can be computed if the textual event rates are used. For this reason, support for textual

event rates was added to the summarization system and when the ischemia corpus was created, it included abstracts without reported outcome numbers.

Looking at the number of condition, group and outcome mention annotations given in Table 4.3 we see there are a similar number of annotations on average in each corpus. Group mentions are the most common particularly in Cardio and Ischemia abstracts which contain more outcome measurements than BMJ. As a result additional group mentions are needed to identify the numerical results in Cardio and Ischemia and these mentions tend to be abbreviated. Table 4.4 shows the number of unique condition, group and outcome entities that are referred to in each of the corpora, as well as the average number of textual references to each of the entities. While groups are the most frequently mentioned entity, trials typically compare only two different groups (a control and an experiment). Occasionally trials will compare multiple experimental groups, but this is less common. Unlike groups, condition and outcome entities tend to only get mentioned once or twice in an abstract.

The total number of ARR values that can be calculated for each corpus is shown in Table 4.5. Since Cardio and Ischemia corpora contain more key values on average for reasons previously stated, more ARR values can be computed on average for their abstracts. Table 4.5 also compares the importance of outcome number verses textual event rates for the computation of ARR values. This shows that although computing event rates from outcome numbers and group sizes is needed in order to calculate some ARR values, these instances are in the minority. Most ARR values can be computed just from the event rates reported in the text. This observation holds for all corpora, including Cardio which was biased to include abstracts with outcome numbers. Half of the potential ARR values in BMJ and most of the ones in Ischemia can only be calculated from reported event rates.

Table 4.1. Basic characteristics of each corpus: the total number of abstracts in each corpus; the number and percentage of abstracts that have section labels; the number and percentage of abstracts containing at least one group size, outcome number or event rate; the average number of sentences in an abstract; the average number of tokens in a sentence; and the average number of acronym occurrences that appear in an abstract.

	BMJ	Cardio	Ischemia
Number of abstracts	188	42	117
Structured abstracts	188 (100%)	39 (93%)	94 (80%)
Abstracts with key values	143 (76%)	42 (100%)	117 (100%)
Avg. number of sentences	13.4	13.2	11.5
Avg. sentence length (tokens)	23.0	27.7	29.9
Avg. acronym occurrences	2.0	8.5	11.0

4.5 Contributions

The corpora described in this chapter are unique. Existing corpora of biomedical abstracts were primarily created for identifying for genes, proteins and their interactions (GENIA [17][18], PennBioIE [21], GENETAG [47], BioInfer [38]). The most relevant available corpus is that of the BioText project. They developed a corpus of abstracts with annotations for treatments, diseases and their relationships [41]. Prior to this thesis work there was no known corpora of abstracts containing annotations needed to train a system to extract and compute ARR values. This chapter describes the first corpora of RCT abstracts that contains annotations for condition, groups, outcomes, age values, group sizes, outcome numbers and event rates.

Table 4.2. The total number of annotated numeric values for each corpus as well as the average number of annotated values in an abstract.

	BMJ		Cardio		Ischemia	
	Total	Avg.	Total	Avg.	Total	Avg.
Age values	138	0.7	16	0.4	26	0.2
Group sizes	294	1.6	92	2.2	152	1.3
Outcome Numbers	266	1.4	191	4.5	122	1.0
Event rates	417	2.2	158	3.8	648	5.5

Table 4.3. The total number of annotated condition, group and outcome mentions for each corpus as well as the average number of annotated mentions in each abstract and the average length (number of tokens) of each mention type.

	BMJ			Cardio			Ischemia		
	Total	Avg.	Len.	Total	Avg.	Len.	Total	Avg.	Len.
Conditions	286	1.5	5.6	131	3.1	6.1	271	2.3	5.2
Groups	1794	9.5	4.0	499	11.9	2.7	1256	10.7	2.6
Outcomes	1632	8.7	4.3	349	8.3	3.8	915	7.8	4.4

Table 4.4. The total number of *unique* condition, group and outcome *entities* for each corpus as well as the average number of unique entities in each abstract and the average number of mentions that refer to a unique entity for each type.

	BMJ			Cardio			Ischemia		
	Total	Avg.	Men.	Total	Avg.	Men.	Total	Avg.	Men.
Conditions	185	1.0	1.5	60	1.4	2.2	153	1.3	1.8
Groups	440	2.3	4.1	88	2.1	5.7	260	2.2	4.8
Outcomes	838	4.5	1.9	173	4.1	2.0	540	4.6	1.7

Table 4.5. Statistics regarding the number absolute risk reduction (ARR) calculations that should be computed for all of the abstracts in each corpus based on information given in the abstract: the total number of ARR that could be computed for each corpus and average per abstract; the total and average number of ARRs that can only be computed from outcome numbers and group sizes that appear in abstract (*computed event rates*); and the total and average number of ARRs that can only be computed from event rates that appear in the abstract text (*textual event rates*).

	BMJ		Cardio		Ischemia	
	Total	Avg.	Total	Avg.	Total	Avg.
All possible ARR	252	1.3	100	2.4	357	3.1
ARR from computed ER only	28	0.1	14	0.3	3	0.03
ARR from text ER only	125	0.7	15	0.4	298	2.5

CHAPTER 5

EXTRACTING KEY ELEMENTS

This chapter describes how ACRES recognizes sections of text that describe key elements of interest in an EBM-oriented summary. These elements describe the population in the study, the treatments involved and the outcome results for each treatment group. This information is needed for health care practitioners to determine how the results of the study apply to their patients. The extracted elements are:

- *Age values.* Values that describe the ages of the population (e.g. minimum age, maximum age and/or median age).
- *Conditions.* Text that describes medical conditions that are common to the population.
- *Groups.* Text that defines the name of the treatment group. The group name usually contains a description of the treatment.
- *Group sizes.* The number of participants in the treatment group.
- *Outcomes.* Text that describes an outcome that was measured for each treatment group.
- *Outcome numbers.* The *number* of participants in a group that experienced a good or bad outcome.
- *Outcome event rates.* The *percentage* of participants in a group that experienced a good or bad outcome.

Some of this information is easier to extract than others. For instance, population age values can be extracted using rules. The system identifies the elements

in stages, starting with the most easily detected entities and passes the detected information along to the next stage where it is used to detect new types of entities.

Figure 5.1 illustrates the sequence of operations applied to a paper in order to find the key trial information. The final result is a version of the text with all of the key trial information labeled.

1. *Pre-processing*: Normalize the text. Identify and chunk key phrases such as "per protocol" and "intention to treat analysis". Parse the resulting sentences and identify phrases that match concepts in the UMLS Metathesaurus, and construct dependency parse trees.

2. *Rule-based extraction*: Identify age, time and primary outcome phrases using rules. Additional rules are used to extract age values from age phrases. Time and primary outcome phrases are used to find elements in the classifier-based extraction stage.

3. *Classifier-based extraction*: Use trained classifiers to identify the sizes of the treatment groups; number of good or bad outcomes; outcome event rates; conditions common to all participants; outcomes measured; and names of treatment groups.

5.1 Pre-processing stage

A series of steps are performed to prepare the text for the mention and quantity finders.

5.1.1 Normalize text. The first preprocessing step is to convert word forms of numbers such as "five" and "seventy-size" to numeric forms (i.e. "5" and "76"). The conversion is done using rules and a lookup table of word forms for numbers

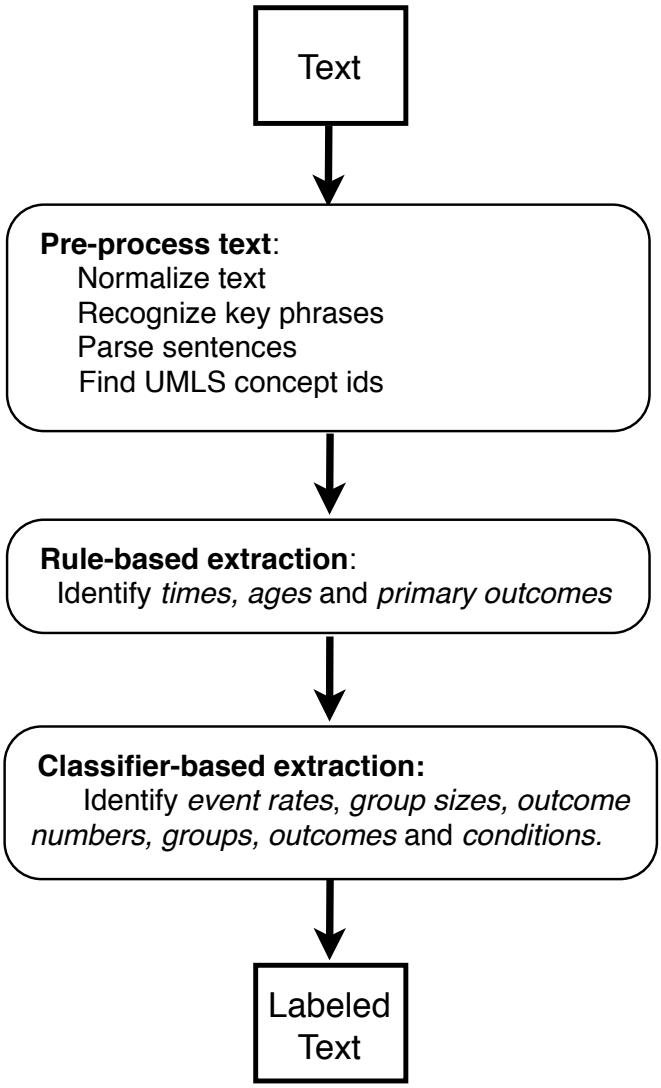

Figure 5.1. The process of identifying all key trial information. Each stage performs an operation on the text and passes the updated text to the next stage.

1. Given a string of text, split it into tokens based on whitespace.
2. Scan the list of tokens until a token appears in lookup table of word forms for $0, 1, \ldots, 19$, 20, 30, 40, 50, 60, 70, 80, 90, 100, then do the following:
   (a) Convert the token to the equivalent number form using the table and store this value in `currentNumber`.
   (b) Move to the next token.
   (c) Repeat the following as long as the current tokens appears in the lookup table.
      i. Convert the token to the equivalent number form using the table.
      ii. If the word was "hundred" or "thousand", multiply `currentNumber` by either 100 or 1000 as appropriate.
      iii. Otherwise, add this number to `currentNumber` and update `currentNumber`.
      iv. Move to the next token.
   (d) Replace the sequence of word form tokens with the value in `currentNumber`.
3. Repeat step 2. until the end of the string has been encountered.

Figure 5.2. Algorithm for converting numbers in word form to number form in a string of text.

$0, 1, \ldots, 19$, 20, 30, 40, 50, 60, 70, 80, 90, 100. A description of the algorithm used to convert numbers in word form to numeric form in a string of text can be found in Figure 5.2. Word forms of numbers are not common, but do sometimes occur, mainly with quantities less than 10. Converting the numbers makes it easier to identify and label quantities.

In addition to converting numbers from word form to numeric form, comparison symbols "<", and ">" are converted to "less than" and "greater than". Parsers tend to treat the comparison symbols and their word forms differently, so this standardizes results from the parser. Note that this is a really a parser problem. The normalization step merely alleviates the problem. In addition, "greater/less than or equal" phrases are reduced to "greater than" or "less than" with an added annotation for the phrase to encode the equality option. This is also done to standardize the

output from the parser. For instance the Stanford CoreNLP[12] parser will parse the following

scored greater than 10 on the severe impairment battery

as

```
(VP (VBD scored)
   (NP
     (QP (JJR greater) (IN than) (CD 10)))
   (PP (IN on)
     (NP (DT the) (JJ severe) (NN impairment) (NN battery)))
```

where the entire comparison phrase "greater than 10" is parsed into a single quantifier phrase. However if the comparison is "greater than *or equal to*" instead of simply "greater than", as in

scored greater than *or equal to* 10 on the severe impairment battery

the parser produces the completely different parse tree

```
(VP
  (VP (VBD scored)
    (ADVP (JJR greater) (IN than)))
  (CC or)
  (VP
    (ADJP (JJ equal)
```

[12]http://nlp.stanford.edu/software/corenlp.shtml

```
    (PP (TO to)
      (NP (CD 10))))
  (PP (IN on)
    (NP
      (NP (DT the) (JJ severe) (NN impairment) (NN battery))
```

which is not only more complicated, but does not even capture the intended meaning. Finally words and phrases such as "above", "more than", "below", and "less than" are covered to "greater than" or "less than" as long as they precede and number. Table 5.1 provides the complete list of comparison tokens and patterns that are detected and the resulting normalized word from.

For similar reasons as with the comparison symbols, "v" and "vs(.)" are converted to "versus". The Stanford parser does not recognize the various abbreviations of "versus" which commonly when reporting numeric results.

5.1.2 Chunk key analysis phrases. There are a number of statistics besides ARR and NNT that are commonly reported in papers. Since these phrases often describe results, they imply the presence of mentions and quantities such as groups, outcomes, group sizes and outcome numbers. Table 5.2 contains a list of commonly reported statistics. This table also includes phrases for the two types of analyses used when reporting outcome results ("intention to treat" and "per protocol"). Detected phrases are replaced with special unique tokens to simplify parsing and classification in later stages. For instance, "intention to treat analysis" is replaced with "ITT_analysis". Not only does this simplify the processing of the sentences by reducing the number of tokens to label, it eliminates certain parsing errors that can occur from parsing phrases such as "95% confidence interval".

Table 5.1. Comparison token patterns and their normalized version.

Detected pattern	Normalized form
<	less than
less than, fewer than	
below, under, at most	
>	greater than
more than, above, over	
<=	less than
< or =	(with “or equal to” annotated)
>=	greater than
> or =	(with “or equal to” annotated)

Table 5.2. The special phrases that are identified and chunked.

intention to treat (analysis)	per protocol (analysis)
(adjusted) hazard ratio	odds ratio
absolute risk reduction	relative risk reduction
absolute risk increase	relative risk increase
95% confidence interval	relative risk
number needed to treat	number needed to harm
risk ratio	

Table 5.3. The numeric patterns that are recognized before parsing along with the parsed form that is inserted back into the parse tree.

Detected pattern	Parse tree
NUMBER / *NUMBER*	(NP (CD *NUMBER*) (IN of) (CD *NUMBER*))
NUMBER of *NUMBER*	
NUMBER ± *NUMBER*	(NP (CD *NUMBER*) (IN ±) (CD *NUMBER*))

5.1.3 Numeric patterns. There are common numeric patterns that often cause problems for the parser. A list of these patterns is found in Table 5.3. To improve parsing, the system identifies these patterns and temporarily replace them with the token "2". After the sentence is parsed, the parsed form of the pattern is inserted back into the parse tree for the sentence, replacing the token "2".

In addition to handling of these numeric patterns, the system identifies explicit percentages of the form "*NUMBER* %". When these are encountered, the number is given a *percentage* annotation and the percent symbol is deleted.

5.1.4 Parse sentences. After the system simplifies the sentences by identifying key phrases and common numeric patterns, the next step in the system is to identify the basic syntactic elements in the sentence and their dependencies. For this, the system uses the Stanford parser. The parser generates phrase structure and dependency parse trees for each sentence.

A phrase structure parse of a sentence recursively breaks the sentence down into its constituent parts such as noun phrases, verb phrases, prepositional phrases and so forth. A dependency parse of a sentence identifies the grammatical relationships that exist between words in the sentence. As an example, consider the following sentence.

> Participants were randomised on a 2:1 basis, 104 to intervention and 49 to remaining on the wait listing (control).

For this sentence, the Stanford parser produces the phrase structure parse tree in Figure 5.3 and the dependency graph in Figure 5.4.

The dependency graph in Figure 5.4 is created from the list of dependencies shown in Figure 5.5. The dependencies form a directed graph where the words in the sentence are vertices and the dependencies define the directed edges. Each dependency has the format *dep_type(head, dependent)*, where *dep_type* is the type of the dependency, the first argument is the *head* or *governor*, and the second argument is the *dependent*. The dependency defines a directed edge from the *head* to the *dependent*. For instance, the dependency `nsubjpass(randomised-3, participants-1)` results in a directed edge from "randomised" to the "participants" which is identified

Participants were randomised on a 2:1 basis, 104 to intervention and 49 to remaining on the wait listing (control).

```
(ROOT
  (S
    (NP (NNS Participants))
    (VP (VBD were)
      (VP (VBN randomised)
        (PP (IN on)
          (NP
            (NP (DT a) (CD 2:1) (NN basis))
            (, ,)
            (NP
              (NP (CD 104))
              (PP (TO to)
                (NP (NN intervention))))
            (CC and)
            (NP (CD 49))))
        (PP (TO to)
          (NP
            (NP (VBG remaining))
            (PP (IN on)
              (NP
                (NP (DT the) (NN wait) (NN listing))
                (PRN (-LRB- -LRB-)
                  (NP (NN control))
                  (-RRB- -RRB-))))))))
    (. .)))
```

Figure 5.3. Phrase-structure parse tree produced by the Stanford Parser for a sample sentence.

as the passive nominal subject of the passive clause containing the verb.

The dependency relationships are used by the mention and number finders. Features based on the neighboring tokens in the graph are used when labeling a given tokens.

5.1.5 MetaMap. In addition to parsing the sentence, the system applies MetaMap [1], described in Section 2.4, to the original sentence to identify words and phrases that correspond to concepts in the UMLS Metathesaurus. MetaMap performs its own shallow parsing on the sentence, chunking it into phrases. Each word in a MetaMap identified phrase is assigned the UMLS concept codes and semantic types of the highest scoring concepts found by MetaMap for that phrase.

Participants were randomised on a 2:1 basis, 104 to intervention and 49 to remaining on the wait listing (control).

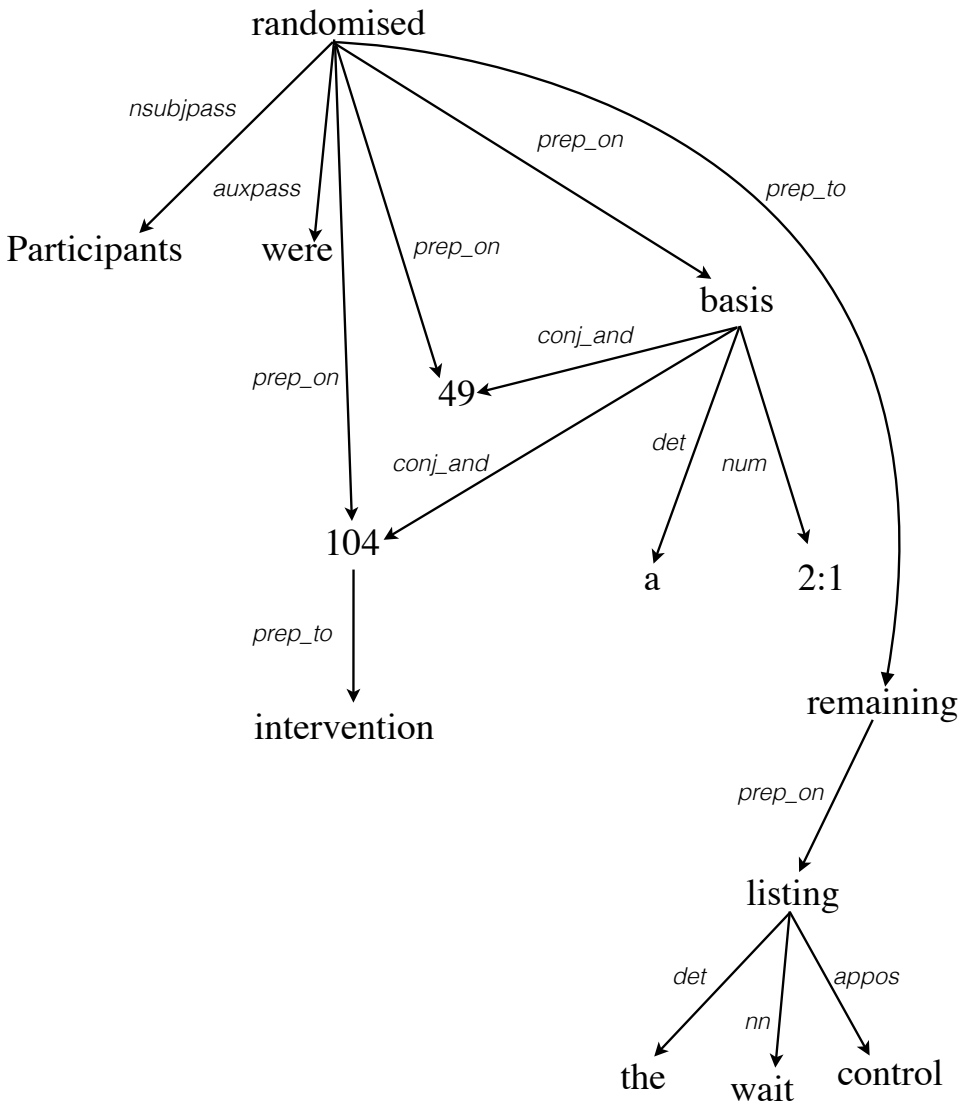

Figure 5.4. Dependency graph based on the collapsed typed dependencies produced by the Stanford Parser.

Participants were randomised on a 2:1 basis, 104 to intervention and 49 to remaining on the wait listing (control).

```
nsubjpass(randomised-3, participants-1)
auxpass(randomised-3, were-2)
det(basis-7, a-5)
num(basis-7, 2:1-6)
prep_on(randomised-3, basis-7)
prep_on(randomised-3, 104-9)
conj_and(basis-7, 104-9)
prep_to(104-9, intervention-11)
prep_on(randomised-3, 49-13)
conj_and(basis-7, 49-13)
prep_to(randomised-3, remaining-15)
det(listing-19, the-17)
nn(listing-19, wait-18)
prep_on(remaining-15, listing-19)
appos(listing-19, control-21)
```

Figure 5.5. Collapsed typed dependencies produced by the Stanford Parser.

5.1.6 Semantic tags. In addition to MetaMap, a collection of word lists for different semantic types are used to add semantic labels to words in the abstracts. There are words lists for the following semantic types.

- *Statistic*. Words used to refer to statistics commonly found in clinical research papers. The word list is given in Table 5.4.
- *People*. Words that are used to refer to populations of people. The word list is given in Table 5.5. This list was developed by Xu et al.[52] to identify population description phrases.
- *Time*. Word is a unit of time. The word list is given in Table 5.6.
- *Measurement*. Word is a unit of measure commonly used in clinical research. The word list is given in Table 5.7.
- *Anatomy*. Words used to refer to anatomical topics. The list was obtained from

Wikipedia[13].

- *Drug*. Word is a drug name. The list was obtained from RxList[14].
- *Procedure*. Word is a medical procedure related term. The list was obtained from MedicineNet[15].
- *Symptom*. Word is a term that refers to a symptom or condition. The list was obtained from MedicineNet[16].
- *Group*. Common words used to refer to treatment groups. The set of words consists of "intervention", "control", "group" and "placebo".
- *Outcome*. Common outcome words. The set of words is "die", "death", "mortality", "injury" and "cure".

If a word appears in one of these lists, it is assigned the name of the corresponding semantic type. This semantic information is used in later stages to aid in recognizing important entities such as age values, conditions, group and outcome mentions.

Table 5.4. Words for common statistics used in clinical research.

HR	ARR	ARI	NNT	NNH	RRR
RRI	RR	RI	ratio	interval	

[13] `http://en.wikipedia.org/wiki/List_of_anatomical_topics`

[14] `http://www.rxlist.com`

[15] `http://www.medicinenet.com/`

[16] `http://www.medicinenet.com/`

Table 5.5. Common terms identified by Xu et al.[52] that are often used to describe trial participants.

patients	men	women	subjects	volunteers	persons
people	participants	children	infants	newborns	teens
students	adults	residents	smokers	neonates	veterans
individuals	donors	males	boys	girls	seniors
adolescents	workers	athletes	users	babies	recipients
addicts	diabetics	outpatients	inpatients	overweight	clients
physicians					

Table 5.6. Time unit strings used to identify time values.

sec	second(s)	min	minute(s)	hr(s)	hour(s)
day(s)	wk(s)	week(s)	month(s)	yr(s)	year(s)

Table 5.7. Units of measure used to identify measurement values.

mm	millimeter(s)	cm	centimeter(s)	cc	mg
milligram(s)	kg	kilogram(s)	oz	ounce(s)	lb(s)
pound(s)	ml	milliliter(s)	millilitre(s)	l	liter(s)
litre(s)					

5.2 Rule-based extraction

In this stage, the system uses rules to identify entities that are relatively easy to detect. This resulting information is then used as features in the later, classifier-based, detection stages. Rules are used to identify the following entities.

- *Special numeric values*: Values that are easy to recognize and frequently occur in abstracts such as confidence intervals or measurements.
- *Times*: Phrases that describe a period of time such as "3 weeks" or "10 days."
- *Age phrases*: Phrases that describe age characteristics of the trial population.
- *Primary outcomes*: Phrases that list the primary outcomes in the study.

5.2.1 Special values. There are many types of numeric values that appear in abstracts, e.g. populations sizes, confidence intervals, hazard ratios, and measurements. While group sizes, the number of bad outcomes and outcome event rates are challenging to identify, there are others that are often easier to identify, such as measurements and statistics such as hazard ratios or odds ratios. Even though many of these values are not currently included in the EBM summaries, it is useful to identify them whenever possible since this information can be used for recognizing other types of entities. Table 5.8 lists the values that we identify. The values are recognized by scanning each sentence checking for matches with the set of patterns found in Table 5.9.

1. *Clinical trial statistics*. These are statistics calculated from the results of the study. They include risk calculations, ratios, and the number needed to treat or harm. Typically there is a term that describes the statistic, e.g. "hazard ratio" that immediately precedes the value. Sometimes there is an additional token

between the term and the value, such as a parenthesis, comma, equals sign, or a form of the verb "to be." At this stage, the statistic terms have already been identified and replaced with a single token that represents the entire term as described in Section 5.1.2. When a token sequence matches the trial statistic pattern, the value in the pattern is assigned the label for the statistic term.

2. *Confidence intervals.* 95% confidence intervals are often reported for trial statistics. As with statistic terms, confidence interval term phrases like "95% confidence interval" have already been recognized during the preprocessing stage and are replaced with a single confidence interval token. The first number in the token sequence gets the label for the *minimum* value in the interval (*CI_MIN*) and the second number gets the label for the *maximum* value in the interval (*CI_MAX*).

3. *Time values.* If a number is followed by a time unit word (see Table 5.6) then the number is considered to be a time value.

4. *Measurement values.* If a number is followed by a measurement unit word (see Table 5.7) or a token that is a combination of letters and forward slashes ("/"), then the number is considered to be a measurement value.

5. *P-values.* In addition to confidence intervals, p-values are often given for reporting the statistical significance of trial results. If a number is preceded by "p" and an equal sign or "less/greater than", then the number is considered to be a p-value.

6. *Other numeric intervals.* Confidence intervals are not the only patterns that appear in clinical research papers. Intervals are also reported for measurements, populations ages, and durations of time. If the pattern for confidence intervals does not match a sequence of tokens, we check to see if the pattern for numeric

Table 5.8. Special values that can be identified using rule-based approach.

absolute risk reduction/increase	hazard ratio	confidence interval
relative risk reduction/increase	odds ratio	p-value
number needed to treat/harm	risk ratio	measurement value
relative risk	numeric interval	time value

intervals matches. If so, then similar to confidence intervals, the first number is considered to be the start of the interval (*INTERVAL_BEGIN*) and the second is considered to be the end (*INTERVAL_END*). If the second value is a measurement, then the numbers are given labels indicating that they define a measurement interval, i.e. *MI_BEGIN* and *MI_END*.

5.2.2 Times. These are phrases that describe a length of time. This can be a follow-up time when an outcome was measured (e.g. "mortality at 12 months"), a duration of time during which a treatment was administered (e.g. "participants were given treatment X for one week"), or a value describing the ages of the trial participants (e.g. "participants were 65 years or older"). All phrases matching either the patterns

- *NUMBER UNITS_OF_TIME*
- baseline [follow-up]

are labeled as *time*, where the set of time unit strings is given in Table 5.6.

5.2.3 Population age phrases. Rules are used to identify phrases describing the age range of the participants in the study. After a phrase is identified, rules are used to parse the phrase and identify the *age values* corresponding to the *min age*,

Table 5.9. Patterns used for identifying special values values.

Detected pattern	Value label
STAT_TERM (“(”|“=”|“,”| *TO BE*)? *NUM*	*STAT_TERM*→ *NUM*
CONFIDENCE_TERM NUM_1 to NUM_2	$CI\_MIN \rightarrow NUM_1$
	and $CI\_MAX \rightarrow NUM_2$
NUM TIME_UNITS	$TIME\_VALUE \rightarrow NUM$
NUM MEASUREMENT_UNITS	$M\_VALUE \rightarrow NUM$
p (“less than” | “=” | “greater than”) *NUM*	$P\_VALUE \rightarrow NUM$
NUM to *M_NUM*	$MI\_BEGIN \rightarrow NUM$
	and $MI\_END \rightarrow M\_NUM$
NUM_1 to NUM_2	$INTERVAL\_BEGIN \rightarrow NUM_1$
	and $INTERVAL\_END \rightarrow NUM_2$

max age, *mean age*, and *median age*. The algorithm for identifying age phrases is as follows.

1. Scan each sentence looking for a word whose lemma is "age" or "old."
2. If found, start at this word's node in the phrase structure parse tree for the sentence, travel up toward the root of the parse tree until we find the smallest phrase that includes this word and at least one candidate age value.
3. Label all words in the resulting phrase as an age phrase.

A number is considered to be a candidate age value if it satisfies the following conditions.

- The number is between 1 and 365 (inclusive).
- The number is not a percentage.
- The number is not a special value (defined in Table 5.8), or if it is a special value, it is either the beginning or end of an interval or it is a time value.

The motivation for identifying age phrases is to find phrases that contain values that describe the age characteristics of the subjects in the clinical trial. We are really interested in the age values themselves, not the phrase as a whole. This means that we need to parse the age phrases and interpret the values that we find to see which ones describe the ages of the population. We do this using a rule-based approach. There are several common patterns that appear in age phrases. These are listed in Table 5.10. The patterns are used to identify candidate age values. These values are then examined and invalid or incompatible age values are discarded. For a detected age value to be considered acceptable, it must satisfy the previously stated candidate value criteria used when identifying the age phrase as well as the following criteria.

Table 5.10. Patterns used for parsing age phrases and recognizing age values.

Detected pattern	Interpretation
(med|median) ... *VAL*	$MEDIAN = VAL$
(mean|average) ... *VAL*	$MEAN = VAL$
between ... VAL_1 ... VAL_2	$MIN = VAL_1$
VAL_1... to ... VAL_2	and $MAX = VAL_2$
greater than ... *VAL*	$MIN = VAL$
over *VAL*	
VAL ... (or|and) (older|more|greater|over)	
less than ... *VAL*	$MAX = VAL$
under *VAL*	
VAL ... (or|and) (younger|less|under)	

- There should be at most only one value of each age value type: *min*, *max*, *mean*, *median*. If there are multiple candidate age values of the same type, discard all values of this type.
- The minimum age should be less than the maximum age.
- The median and mean ages should be between the minimum and maximum ages. The default minimum and maximum ages are 0 and ∞.

5.2.4 Primary outcomes. Primary outcomes are the outcomes that are the main focus of the study. All other outcomes are secondary outcomes. In some abstracts, the primary outcomes are clearly identified as in the sentence

> The primary outcomes were the Oswestry disability index and the shuttle walking test measured at baseline and 2 years after randomisation.

Sentences like this one occur frequently enough so that the system can use them to identify outcomes. The system parses these sentences and labels the phrases containing the description of the primary outcome. These labels are used later as features in the outcome classifier. The primary outcome phrases are identified by scanning the abstract for occurrences of the following pattern.

primary (composite)? (outcome | outcomes | endpoint | endpoints)

If this pattern is followed by a form of "to be" acting as a linking verb, then all words in the complement are given the label "primary_outcome." For instance, if the previous example is parsed as:

```
(NP (DT the) (JJ primary) (NNS outcomes))
(VP (VBD were)
    (NP (NP (DT the) (NNP Oswestry) (NN disability) (NN index))
          (CC and)
          (NP (NP (DT the) (NN shuttle) (VBG walking) (NN test))
                (VP (VBN measured) (PP (IN at) ...)))))
```

then each word in "the Oswestry disability index and the shuttle walking test measured at baseline and 2 years after randomisation" receives the primary outcome label. This label will be used as a feature for the classifier that is trained to identify outcome words. Because of this, it is acceptable that some words recieve primary outcome labels even though they do not belong to outcome mentions. For instance, "2 years after randomisation" describes the follow-up time when the outcomes were measured and it is not part of the outcome "the shuttle walking test."

Finally, the primary outcome label also serves to determine which detected outcomes are primary outcomes. For example, suppose the outcome finder described in Section 5.3.1 labels "shuttle walking test," as an outcome. Since this outcome mention contains at least one word with the primary outcome label, the mention is considered to describe one of the primary outcomes in the study.

5.3 Classifier-based extraction

While information such as times and age phrases can be identified with a rule-based approaches, most of the information that needs to be extracted does not exhibit the same level of regularity that can be easily exploited. Therefore we use a supervised machine learning based approach for finding the remaining information. More specifically, we treat the task of finding conditions, treatment group names, outcomes, and even the treatment group sizes and outcome numbers as an application of *named entity recognition*. The goal of named entity recognition is to automatically identify the sections of a text that name entities such as people, organizations, locations, etc. Entities may also be a specific type of information such as email addresses, dates/times, or monetary values. Here, the entities are conditions, treatment groups and outcomes.

While there has been considerable research focused on finding named entities in biomedical research papers, this work has mainly been concerned with finding the names of genes, proteins and drugs. Relatively little research has been focused on finding treatment groups, outcomes, or the quantities that we need. Related prior work includes the use of probabilistic graphical models for identifying treatments and diseases [41]; shallow semantic parsing to identify treatments and outcomes [35]; conditional random fields (CRF) to extract diseases [24][6] and trial characteristics including the age groups, trial locations and the number of observations [25]; rules

and hand-crafted grammar to extract number of trial participants and population demographics [52]; support vector machines to identify the total number of participants in the trial [15]; and a linear combination of classifiers to identify sentences that contain PICO elements [3].

There are various challenges to recognizing groups, outcomes and conditions. They lack common orthographic features such as numbers, special characters (e.g. ':', '-', '@'), or uppercase letters that aid in recognizing entities such as dates, email addresses, or genes/proteins. Mentions may be long and have poorly defined boundaries such as the following example.

> conventional coronary artery bypass grafting surgery using cardiopulmonary bypass

The entities described in the text may not contain any terms found in medical lexicons, such as "playing the digeridoo" and "swimming with dolphins" which are potential treatments for sleep apnea and depression, respectively. Also, some entities may only be referred to indirectly as in following.

> half had additional advice on anxiety management and half *did not*

In this example, the second treatment, "no additional advice on anxiety management," is not explicitly mentioned, but is merely implied. Handling this type of case is currently an unstudied problem.

Extracting group sizes, outcome numbers and event rates is also challenging. There are many difference types of numbers that appear in a text besides these. The total number of participants in the trial, demographic percentages, hazard ratios, odds ratios, number of outcome events and percent changes in outcome measurements are all reported in a similar manner to the values ACRES needs to extract.

ACRES uses a collection of classifiers to label each token in the sentence as a mention token (condition, group or outcome) or not (*other*). Consecutive tokens with the same label are grouped together and constitute a *mention*. There are three separate trained classifiers for conditions, groups and outcomes (one for each type). During development it was discovered that separate binary classifiers outperformed a single joint classifier for mention extraction. A similar approach is used for extracting group sizes, outcome numbers and event rates. A trained classifier labels percentages and floating point values in a sentence as event rates or not. Since group sizes (number of people in a group) and outcome numbers (number of group participants that experience an outcome) are always integers and are often reported together, a trained classifier is used to label integers as group size, outcome number or *other*. These classifiers are all applied to the sentences in *parallel*.

5.3.1 Classifier. The system uses conditional random field (CRF) classifiers [22] to perform the token labeling. Conditional Random Fields were designed for segmenting and labeling sequential data (e.g. labeling tokens in a sentence) have been successfully applied to many different natural language processing tasks such as, part of speech tagging [22], named-entity recognition [27][13][24], shallow parsing [43], information extraction [36], and table extraction [37]. We use a first order linear chain CRF where the label of the current token is considered to be partially dependent on the labels of the tokens immediately before and after it. The classifier is *supervised*, that is it is trained on a collection of labeled tokens. For a CRF classifier implementation, the system uses the MALLET v2.0.7 SimpleTagger [28].

Given the sequence of words in a sentence $\mathbf{w} = < w_1, w_2, \ldots, w_n >$ a CRF classifier finds a sequence of labels or tags $\mathbf{t} = < t_1, t_2, \ldots, t_n >$ that maximizes the

conditional probability

$$p(\mathbf{t}|\mathbf{w}) = \frac{1}{Z(\mathbf{w})} \exp\left(\sum_{j=1}^{n}\sum_{k}\lambda_k f_k(t_{j-1}, t_j, \mathbf{w}, j)\right), \tag{5.1}$$

where $Z(\mathbf{w})$ is a normalization factor and f_k is a binary feature function. Each feature function f_k has its own weight λ_k.

5.3.2 Features. With a few exceptions, all of the trained classifiers use the same set of features to classify tokens. The following describes the features used by both classifiers to label tokens.

1. *Token.* These features are capture characteristics of the token in question. They include

   - *Lexical.* Features that capture the characteristics of the token. These features do not apply if the token is a number.
     - Token lemma and POS tag.
     - The special annotation for the token, if any.
     - Is the token an acronym?
   - *Numeric.* Features that capture the characteristics of the token if it is a number.
     - Is the number a percentage, integer, or floating point value?
     - Is the number negative?
     - Is it a *small* integer (< 10)?

2. *Semantic.* These features capture information about the meaning of the token in the sentence.

   - *UMLS.* The MetaMap-based semantic features for a token are the UMLS Metathesaurus semantic types (if any) for the phrase containing the token as well as the UMLS concept ID for the phrase.

- *Semantic tag.* These features are the semantic tags described in Section 5.1.6 that are assigned to the token during the preprocessing stage.
- *Previously assigned labels.* These features are the labels assigned to the token by previous extraction stages. For all three mention classifiers they include: *time*, *age*, *primary_outcome*.
- *Pattern*: Is the integer part of a pattern that often indicates group sizes or outcome numbers? Patterns used are:
  - "n = *INTEGER*". This pattern often indicates that the integer is a group size.
  - "*INTEGER* / *INTEGER*" or "*INTEGER* of *INTEGER*". These patterns are often used to report the proportion of participants in a group that achieve a given outcome, i.e. *outcome number* / *group size*. The pattern does not apply if the first integer is greater than the second, or if one of the integers has been identified as an special value.

3. *Acronym.* If the token is an acronym get the lexical and semantic features for all expansions of this acronym in the abstract.

4. *Syntactic context.* These features capture the syntactic context of the token in question.
   - Is the token inside parentheses?
   - The closest parent verb in the parse tree. Starting at the token in question travel up the parse tree until a verb phrase is reached. Then return the main verb (the first word) in the verb phrase.
   - Dependency features. These features are based on a collapsed typed dependency parse of the sentence. They consist of the *token* and *semantic*

features for each governor and dependent token and the type of relationship. For the number classifiers, only features for the governor tokens are used; any dependent tokens of number are ignored.

5. *Token context*. Features based on the *three* or *four* tokens on either side of the token in question. These features include the *token* and *semantic* features computed for the context tokens. The number classifiers use a window of three tokens, the others use four.

6. *Sentence context*: Label for the section that the sentence appears in, e.g. "Objective" or "Results", as well as the NLM category assigned to the section.

5.3.3 Extracting key numbers. Two trained linear chain CRF classifiers are used to label the numbers using the features described in Section 5.3.2. Unlike the classifiers used for conditions, groups and outcomes, these classifiers only label tokens that are numeric values in sentences. They do not assign labels to every token in a sentence. One classifier is trained to label *percentages* and *floating point* values as *event rates* or *other*. The other classifier labels *integers* as *group size*, *outcome number*, or *other*. The classifiers do not label numbers if they were identified as special values as described in Section 5.2.1 or if they are followed by the token "times".

5.3.4 Finding Conditions, Groups and Outcomes. Three trained linear chain CRF classifiers are used to identify the names of *conditions*, *treatment groups* and *outcomes* mentioned in an abstract. There is one trained classifier for each mention type. A classifier for a given type (e.g. group) label each token in the sentence as belonging to a mention of that type (i.e. assigned label GROUP) or not (assigned label *other*). Consecutive tokens with the same label (excluding those labeled as *other*) are grouped together and considered to be a *detected mention* for that type.

While both the condition and outcome classifiers are applied to all sentences

in an abstract, the group classifier is only applied to sentences that contain non-negative numbers that are not special values as described in Section 5.2.1. Since numbers often report some characteristic of a treatment group such as treatment duration, size or number of outcomes, they are good indicators that a sentence will contain a group mention. Compared with all abstract sentences, there is less variation in the structure of sentences that report group size and outcome results for groups. Focusing on sentences that are more likely to contain group and outcome results, allows the classifier to better learn the structure of these sentences and identify group mentions. For similar reasons, the group classifier also ignores sentences in sections labeled "INTERVENTION(S)" structured abstracts. Sentences in these sections are problematic. They vary structurally from sentence fragments to complete sentences that give detailed descriptions of one intervention or list multiple interventions. They differ greatly from sentences that report numerical characteristics of groups. Future work includes developing separate methods for extracting group information from these sentences.

5.4 Re-ranking classifier output

Experience gained from early development of the system revealed that increasing mention recall, particularly for *outcome* mentions, is more critical for computing ARR values than increasing precision. Classifier ensembles are a commonly used approach for increasing classifier performance. With this approach, multiple different classifier models are applied to the same data and their results are combined using majority vote or more sophisticated schemes. There are many different methods for creating classifier ensembles. The key to successful ensemble approaches is high *complementarity* between the different models [4], the percentage of times where one classifier is wrong and another is right. There should be as little agreement on errors as possible. Generating different classifier models with this kind of constructive

1. Compute the top-n labelings of the sentence.
2. For each token in the sentence, calculate the most popular label for that token.
3. Find the labeling in the top-k labelings of the sentence that contains the most tokens which have been assigned the most popular label for that token.
4. Use this labeling to re-label each token in the sentence.

Figure 5.6. Algorithm for re-ranking the top-k labelings of a sentence

disagreement in practice can be challenging, particularly with conditional random field models. Previous approaches have trained classifiers on different (potentially overlapping) subsets of the training or trained classifiers with different subsets of features.

This document presents a novel alternative to previous ensemble approaches. Instead of creating an ensemble of CRF models with differing parameters, it uses the top-k CRF labelings of a sentence to identify the best labeling for that sentence. A benefit of CRF inference is the ability to obtain the top-k most likely labelings for a given sentence. Each labeling is guaranteed to differ by at least one token label. The algorithm for finding the best labeling of a sentence is described in Figure 5.6. The algorithm uses the top-n labelings to find the sequence of most popular labels for the sentence. It uses this sequence to select the best matching labeling from among the top $k \leq n$ labelings. Increasing k improves recall, but precision suffers. For re-ranking outcome labelings, the system uses $k = 3$ and $n = 15$. These values were determined empirically using cross-validation on the combined BMJ-Cardio corpus. For breaking ties when selecting the labeling that best matches the sequence of popular labels, preference is first given to the labeling containing the fewest number of *other* labels (more likely to increase recall), then preference is given to the higher ranked labeling (the one the classifier thought was more likely).

Table 5.11. Negation words used by the system.

not	no	without	never	neither	none	non

5.5 Post-processing classifier output

Finally, after the classifiers have been applied to the abstract and re-ranking has been applied to the outcome labelings, the following collection of rules is applied to clean-up the classification results for *groups* and *outcomes*.

- Look for noun phrases that end with the token "group", as this is usually an indicator of a group mention, and label all tokens in the noun phrase as *group*.
- Discard group or outcome mentions that only consist of stop words, symbols, times or measurements.
- Scan sentences looking for the longest token sequences that match detected mentions (disregarding token order) and assign the same label to these token sequences.
- Any parentheses or commas that begin or end a mention are removed.
- If a negation word precedes the first word in a mention, it is added to the mention. Table 5.11 contains the list of the negation words used.
- Resolve obvious group/outcome conflicts. If a mention of one type is a proper subset of a mention of another type, delete the shorter mention.

The purpose of these rules is to improve recall and precision specifically for groups and outcomes which is critical for successful computation of ARR values.

5.6 Contributions and related work

The novel contributions of the key element extraction phases of the system are:

- *Methods used to normalize text prior to parsing.* First work to normalize comparison operators and phrases; chunk key phrases; and employ special handling of numeric patterns prior to parsing medical text.
- *Novel rule-based methods for identifying special numeric values in medical text.*
- *Novel rule-based method for extracting age values.*
- *A method for finding and extracting outcome numbers and event rates.* Although there has been a little prior work that involved extracting the total number of trial participants[12][52][15] and the sizes of treatment groups[12], to my knowledge this is the first work to propose extracting outcome numbers or the event rates.
- *Novel feature set for recognizing condition, group and outcome mentions using CRF classifier.* In addition to lexical, semantic and syntactic features, the classifier uses features from tokens that share dependency relationships and tokens from acronym expansions.
- *Novel method method for improving recognition of outcomes using re-ranking of alternate CRF labelings.*
- *Novel rule-based methods for post-processing output of group and outcome classifiers.*

5.6.1 Prior work related to finding clinical entity mentions. There has been much work on entity recognition over the years a survey of this work can be found in

[31]. The following is the related work most relevant to finding the types of entities that I am looking for.

Rosario and Hearst [41] developed a hidden Markov-like graphical model for identifying treatments and diseases in sentences from medical texts and classifying their relationships. The features used by their system, for a given word, were: the word itself, its part of speech, the phrase constituent it belongs to, its Medical Subject Headings (MeSH) id, various orthographic features and whether the MeSH subheirarchy of the word is usually corresponds to treatments, diseases or neither. They found that the most important features for deciding if a word was part of a treatment or disease were: the word itself, its MeSH id and part of speech.

Paek et al. [35] used shallow semantic parsing to identify agent, patient and effect (i.e. treatment, group, and outcome) entities in sentences containing one of five predicates ("reduce", "improve", "suggest", "increase", and "use"). These sentences were extracted from the conclusion sections of abstracts of randomized controlled trials. Sentences were parsed into their constituents and a classifier was used to identify the constituents that were arguments for the predicate in the sentence.

Dawes et al. [10] investigated the feasibility of identifying population/problem, exposure/intervention, comparison, outcome, duration and results entities in a set of 20 abstracts from clinical studies. They compiled a list of terms that often indicate their key entities. For instance, they found that words such as "mortality" and "incidence" that are commonly part of outcome entities. However, while they were able to identify terms for their comparison, outcome, duration and results entities, they where less successful in identifying common terms for the patient/population/problem and exposure/intervention entities which correspond to the group and treatment entities that we wish to extract.

Leaman and Gonzalez [24] developed BANNER, a biomedical CRF-based NER system. They applied their system to various publicly available biomedical data sets including [41] and achieved good results compared with existing, well known NER systems.

Summerscales et al. [46] used a CRF-based NER system to find treatments, groups, and outcomes. They found that most useful features for determining if a word was part of a treatment or outcome were: the word itself, its part of speech, its context features (features from neighboring words), and the label from the section of the abstract that the word appears in (assuming the abstract is formatted with section labels). For identifying group mentions they found that the word itself and its context features were most useful. Word shape features (character n-grams and various binary word shape features), while commonly used for named entity recognition, where not found to be helpful for finding treatments, groups, and outcomes.

Chung [7] developed a method for labeling the sentences in the abstracts of medical papers. Sentences were labeled based on their rhetorical role in the abstract. The labels used were *Aim*, *Method*, *Results*, and *Conclusion*. Some sentences were also labeled as *Intervention*, *Outcome*, or *Participant*.

In [8] Chung created a method for extracting the phrases that mention the names of all of the treatments groups joined by coordinating conjunctions. For instance it would extract *allopurinol or placebo* from the sentence

Patients were randomly allocated to allopurinol or placebo.

Xu et al[51] present an unsupervised, iterative method for learning patterns for extracting treatment terms from randomized controlled trials for the purpose of creating a medical treatment lexicon. Sentences are parsed and noun phrases are

labeled as treatments if they are part of a phrase that matches one of the learned patterns (e.g. "treated with *NP*").

Chowdhury and Lavelli[6] develop a CRF-based NER system for recognizing disease mentions. Their system uses general linguistic features (e.g. POS tags), orthagraphic features (e.g. is initial letter capitalized, are all letters capitalized), context features (e.g. token bi-grams and tri-grams near token to be classified), syntactic dependency features (e.g. target token to which the current token is a direct or indirect object), and dictionary lookup features from the UMLS Metathesaurus. They show that their system is able to outperform BANNER[24] for recognizing diseases.

Boudin et al. [3] use an ensemble of classifiers to identify sentences that contain PICO elements. Their ensemble consisted of a J48 decision tree, a random forest of decision trees, a support vector machine, a multi-layer perceptron and a Naive Bayes classifier. The classifiers use features based on the presence of cue-words, overlap with the title, MeSH semantic types, number of words, punctuation marks and numbers in the sentence.

Lin et al. [25] use CRF-based approach to extract trial characteristics including the age group phrases, interventions, trial locations and the number of subjects in the trial. Their approach uses orthographic features, stemmed word tokens, key word lists and numeric features. Although they extract age phrases, the do not parse the phrases to recognize the individual age values.

5.6.2 Prior work related to finding quantities. Unlike the task of finding clinical entity mentions, there has been little prior work that has focused on finding the quantities that we seek. Demner-Fushman and Lin[12] use a pattern-based approach to find and extract population sizes. Xu et al.[52] developed a method to extract

subject demographic information from medical abstracts. This information includes the trial sizes as well as the disease/symptoms studied and subject demographic information such as age, gender, and ethnicity. They first use text classification augmented with a Hidden Markov model to identify sentences containing demographic information and then parse the sentences to extract the the desired information.

The most recent attempt at finding and extracting population sizes is by Hansen et al.[15]. They focus solely on finding the original number of participants in the trial, before subjects drop out or are allocated to different treatment groups. They use a variety of features to classify integers found in an abstract. The largest number is then select from the set of candidate trial size numbers for a given abstract. This number is considered to be the total number of participants in the trial.

CHAPTER 6

SUMMARY CONSTRUCTION

After the key elements have been identified in the text, the summarization system identifies the elements that should be associated. This process enables the system to determine the unique conditions, groups and outcomes discussed in the paper. These association stages are also needed in order to compute summary measures from the values extracted from the text. This chapter describes the stages illustrated in Figure 6.1. It details the methods used for identifying the various relationships that exist between the detected elements, computing summary measures and constructing EBM-oriented summaries.

6.1 Element associations

The element extraction phase of the system identifies sections of text that correspond to different types of entities that are needed in EBM-oriented summaries. However, a summary cannot be constructed directly from this raw data.

Entities such as conditions, groups and outcomes may be mentioned multiple times in the text. To have a concise summary, the summarization system needs to recognize when multiple mentions refer to the same entity. The redundant mentions should be clustered and a canonical name should be found to represent that cluster in the summary.

Key values extracted by the system must also be processed further before they can be included in a summary. Group sizes, outcome numbers and event rates are not informative until we know which outcome they measure and/or to which group they belong.

Table 6.1 lists the associations that need to be made among the extracted

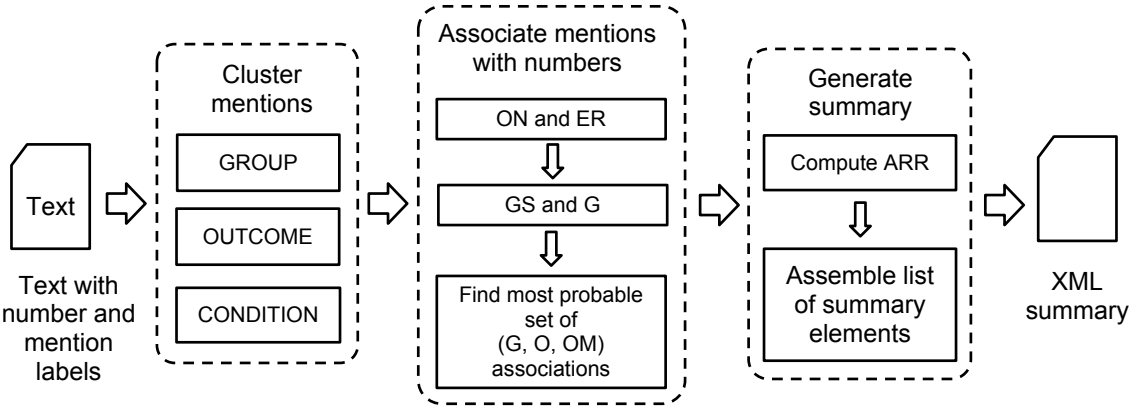

Figure 6.1. Overview of stages that take extracted elements, identify relationships between them and compile the resulting data into EBM oriented summaries.

elements in order to create concise and informative summaries. The system employs a combination of rule-based and classifier-based methods to perform the necessary associations.

Table 6.1 also introduces a new element type, *outcome measurements*. Here, an outcome measurement refers to all of the values from the text that describe the percentage of people from a single group who achieve a certain outcome. This can be either an outcome number with a group size, an event rate reported in the text or a combination of the two. The system needs to identify the cases where both outcome numbers and event rates are reported and determine the values that refer to the same outcome measurement.

6.2 Clustering mentions

The same entity, such as a condition, group or outcome, may be mentioned multiple times in a paper. The system needs to identify mentions that refer to the same entities and group them in to clusters. The purpose of this procedure is to:

- identify the *unique entities* that are discussed in the paper which is necessary

Table 6.1. A description of the associations that need to be made.

Association	Description
(condition, condition)	Match mentions referring to same condition.
(group, group)	Match mentions referring to the same group.
(group, group size)	Match group size with its group.
(group, outcome number)	Match group with value describing number of group participants with a given outcome.
(group, event rate)	Match group with value describing percentage of group participants with a given outcome.
(outcome, outcome)	Match mentions describing the same outcome.
(outcome, outcome number)	Match outcome with value describing number of group participants with this outcome.
(outcome, event rate)	Match outcome with value describing percentage of group participants with this outcome.
(outcome, outcome measurement)	Match outcome with value(s) describing the percentage and potentially the number of participants with this outcome.
(outcome number, group size)	Match the number of group members achieving an outcome with the size of the group.
(outcome number, event rate)	Match the number of group members achieving an outcome with the reported percentage achieving this same outcome.

for the final summary

- allow summary statistics to be calculated when the group size is mentioned in a separate sentence as the outcome number

The challenge with this task is that the same entity may be referred to in different ways throughout the paper. For instance, "intensive rehabilitation programme", "rehabilitation", and "control group" may all refer to the same treatment group.

A concern with clustering mentions is the possibility of merging two mentions that refer to different entities. For this application, erroneously clustering two mentions that refer to different entities is significantly more serious than not clustering two mentions that refer to the same entity. Redundant elements in a summary are undesirable, but they are not misleading and they not reduce the information content of the summary. However, since the summary includes only one entry for each mention cluster, erroneous clusters that contain mentions for different entities will only be represented by a mention that describes one of the entities. The resulting summary may be missing references to key entities.

In order to avoid clustering mentions that refer to different entities, while accepting a small risk of redundant clusters, the system employs a conservative rule-based approach for clustering mentions. It repeats the following steps for each sentence S in an abstract.

1. *Cluster identical mentions within sentence S.* This step creates a collection of sets $C_s^1, C_s^2, \ldots, C_s^n$, where all of the mentions for a set C_s^i are identical or share a grammatical relationship.

2. *Merge $C_s^1, C_s^2, \ldots, C_s^n$ with the global set of clusters for the abstract $C_a^1, C_a^2, \ldots, C_a^n$.* If a mention in C_s^i is an exact match for a mention in global cluster C_a^j,

merge C_s^i with C_a^j. Otherwise, if no exact match for C_s^i exists, add C_s^i to the set of global mention clusters.

Here, two mentions are considered identical if their sets of lemmatized tokens are the same (not counting common words listed in Table 6.2). When clustering group mentions, if the two mentions have an appositive relationship, recognized by the parser, they are considered to be identical and they are included in the same cluster. When clustering outcome mentions, if one mention is a the subjective complement of a generic reference such as "the primary outcome" (e.g. "*the primary outcome* was *X*"), then the two mentions are also considered to be identical.

Each cluster has one mention that *represents* the cluster in the summary. Initially when clusters are created, each contains only a single mention and that mention is the representative by default. When two clusters are merged, the *longer* representative mention (i.e., the one with the most tokens) is chosen as the representative for the new cluster. The longer one is chosen since it is likely to be more informative. To prevent overly long and potentially erroneous mentions from representing the cluster, a threshold is placed at seven tokens. That is, the longer mention is always chosen, unless it is longer than seven tokens.

When checking for matches between a sentence cluster C_s^i and a global cluster C_a^j, the two clusters are considered to match if the mention representing C_s^j matches one of the mentions in C_a^j using the matching criteria for merging sentence clusters. If no match is found among the set of global mention clusters, the sentence cluster C_s^i is added to the set of global clusters. With *condition* and *group* clusters, before an unmatched cluster C_s^i is added to the global cluster list, it is compared with the global set one more time with a *relaxed* match criteria. With the relaxed criteria, the representative from sentence cluster is considered to match that of a global cluster

Table 6.2. Common words that are ignored when comparing mention to see if they match.

a	an	the	of	group(s)	arm	had

if they have any non-trivial words in common or (for group mentions) they have the same role in the trial (either *experiment* or *control*). The match must be unambiguous. If a sentence cluster matches more than one global cluster, it is considered still unmatched and it is not merge with any of them. Non-trivial words are any word that is not a symbol or a common word as given in Table 6.2. Possible roles for group mentions are *experiment*, *control* or *unknown*. If a group mention contains terms that are common indicators of control groups, its role is considered *control*. Otherwise, if it contain terms commonly associated with references to experimental groups, its role is *experiment*. If neither option is possible, its role is *unknown*. For the purposes of matching based on role, the role must be either experiment or control. Table 6.3 gives the lists of terms used to identify the role of a group.

The relaxed matching criteria is not used for merging outcome clusters since there tends to be more overlap between the mentions and small differences in mention wording can imply different outcomes. Studies will often report results for both individual and *composite* outcomes, that is, outcomes where one of a few different sub-outcomes are possible. For instance a study may report the number of subjects who suffered a stroke and the number who suffered either a stroke *or* a myocardial infarction.

Table 6.3. Common terms that often indicate the role of a treatment group in a study. An experimental group mention cannot contain any control terms.

Control		Experiment	
standard care	usual treatment	experiment(al)	(new) treatment
usual care	standard treatment	(new) therapy	(new) intervention
control	placebo		

6.3 Associating mentions and values

In the element extraction stage, the system identifies numbers in the text that are group sizes, outcome numbers and event rates. However, the values themselves are not useful unless the system can determine which group and outcome in the study the numbers were recorded for. The system associates these numbers with groups and outcomes in the following series of steps.

1. *Associate outcome numbers and group sizes.* Use rule-based approach to identify instances where outcome numbers and group sizes are reported together in the text and link the two numbers.

2. *Associate outcome numbers and event rates.* Use rule-based approach to identify outcome numbers and event rates that report the same outcome measurement for the same group.

3. *Associate group sizes with groups.* Use classifier based approach to identify the group that the group size is characterizing. This step is only applied to group sizes that are not already associated with outcome numbers. Those group sizes will be associated with groups along with their linked outcome numbers in a later step.

4. *Associate groups and outcomes with outcome measurements.* Use classifier-based approach to identify the group and outcome that the outcome measurement is for.

After these associations have been made, the system can then calculate summary measures for the trial outcomes.

6.3.1 Rule-based association. The initial mention-value association stages use rule-based approaches to link values of different types that are reported together. These value-value associations are then used in later classifier-based mention-value association stages.

6.3.1.1 Associating outcome numbers with group sizes. When outcome numbers appear in the text, they are often reported along with the number of participants in the group as in the following example.

> 7/57 (12%) of the probiotic group developed diarrhoea associated with antibiotic use compared with 19/56 (34%) in the placebo group (P=0.007).

If an outcome number is reported together with a group size in the text, matching either the pattern "ON / GS" or "ON of GS", then the outcome number and group size are linked and the pair are used to calculate an event rate value.

6.3.1.2 Associating outcome numbers with event rates. Authors may report outcome results for treatment groups in a few different ways.

- *Outcome number only.* Authors report *only* the *number* of group participants who achieve a given outcome.
- *Event rate only.* Authors report *only* the *percentage* of group participants who achieve a given outcome.

- *Both outcome number and event rate.* Authors report *both* the *number* of participants and the *percentage* of participants who achieve a given outcome.

When both outcome numbers and event rates are present in a text, the system needs to determine if the they refer to the same outcome measurements or to different ones. Fortunately, when both are reported, they often appear in close proximity and most cases can be characterized by one of a few different patterns list in Table 6.4. The system uses these patterns to identify outcome numbers and event rates that refer to the same outcome measurement for a group. When found, the number pairs are linked and together they get associated with group and outcome mentions.

The algorithm for linking outcome numbers and event rates that refer to the same outcome measurement for a group is given in Figure 6.2. It links outcome numbers and event rates that match the patterns listed in Table 6.4 as long as their event rates are not *incompatible*. That is, either the event rate calculated from the outcome number and its associated group size is *equivalent* to the event rate extracted from the text, or the outcome number does not have an associated group size. The event rates for an outcome number and an extracted event rate are considered to be *equivalent* if they have the same value when rounded to the nearest percent; the floor of both values is the same; or the ceiling of both values is the same.

6.3.2 Classifier-based association. Associating group size values with group mentions and associating outcome measurement values with group and outcome mentions is more challenging than the previous tasks of linking values that are reported together. There is more variety in how group size and outcome measurement information is reported in a sentence for groups and outcomes. Mentions are not always adjacent to their values. For these reasons, the system employs a classifier-based approach.

1. For each possible (ON, ER) pair in the sentence do:

   (a) If the pair (ON, ER) share a relationship that matches one of the patterns found in Table 6.4 and they do not have incompatible event rate values, then:

      i. Link the pair (ON, ER) and do not consider either value as part of another pair.

Figure 6.2. Algorithm for linking outcome numbers (ON) and event rates (ER) that report the same outcome measurement for the same group.

Table 6.4. Common patterns used when reporting both the number of outcomes and the event rate for an outcome.

Pattern	Example
ER (ON	"...incidence of low cardiac output syndrome was 46% (17 of 37) in the single-dose group..."
ON (ER	"... 90 (36%) control patients had an adverse composite primary outcome..."
ON POPULATION (ER	"... 9 patients (9%) in the placebo group..."
ON of GS (ER	"... 3 of 37 (8.1%) patients in the continued-transfusion group developed new brain MRI lesions..."
ON of GS POPULATION (ER	"...ischemic stroke occurred in 279 of 3705 patients (7.5%) assigned to apixaban..."

To associate values with mentions, the system first estimates the probability of each possible value-mention pairing in the sentence using the MegaM v0.92 [9] maximum entropy (MaxEnt) classifier.

- (group size, group)
- (outcome number, outcome)
- (outcome number, group)
- (event rate, group)
- (event rate, outcome)

A MaxEnt classifier estimates the probability that a given instance x has a label c,

$$p(c|x) = \frac{1}{Z(x)} \exp\left(\sum_i \lambda_i f_i(c, x)\right), \tag{6.1}$$

where f_i is a binary feature function and $Z(x)$ is a normalization factor defined by

$$Z(x) = \sum_c \exp\left(\sum_i \lambda_{c,i} f_i(c, x)\right). \tag{6.2}$$

For the purpose of association, an instance, $x = \langle v, m \rangle$, is a possible association between a value v and a mention m in a sentence and c = TRUE, if v and m should be associated, and c = FALSE, if the pair should not be associated. The MaxEnt classifier will estimate the probability that a given value-mention pair should be associated, $p(\text{TRUE}|\langle v, m \rangle)$, based on features that characterize the relationship between v and m. For now, the system only looks for associations that exist within a sentence.

After the potential pair probabilities have been estimated, the task of associating values with mentions can be treated as an *assignment problem* at the sentence

level. The objective is to find the optimal set of associations between values and mentions in a sentence that maximizes the sum of the pair probabilities. The optimal set of association is determined using the Hungarian method.

6.3.2.1 Association features. A trained MaxEnt classifier is used to estimate the probability that a given value and mention in a sentence should be associated. A separate classifier is trained for each type of value-mention pairing. The following is a description of the features used by the classifier to estimate the probability that a given (*value*, *mention*) pair should be associated.

- *Proximity features.* Features that capture the proximity relationship between the value and mention.
  - Is the mention the closest one to the value (i.e. the mention with the fewest number of tokens in between the mention and the value)?
  - Does the mention occur after the value?
  - Are the mention and value adjacent in the sentence or, at most, separated by a parenthesis?
- *Dependency features.* Features that capture the dependency relationships (if any) between the value and the tokens in the mention.
  - Is the value a *governor* of one of the tokens in the mentions? If so, what is the relationship type?
  - Is the value a *dependent* of one of the tokens in the mention? If so, what is the relationship type?
- *Intermediate features.* Features that based on the tokens and elements in between the value and mention in the sentence.

  - Are there any *group*, *outcome*, *group size*, *outcome number* or *event rates* between the value and mention in the sentence?
  - Do any of the tokens in the set {'versus', ',', ';'} appear between the value and mention? If so, which ones?
  - Do "and" or "or" appear in between the value and mention?

- *Order features.* Do both elements in the pair appear in similar positions in the sentence? For instance, for a given (group size, group mention) pair are they both the first group size and group mention in the sentence? The pair are considered to have the same order if both of the following conditions are met:

  1. $N_v \equiv N_m \pmod{N_{min}}$
  2. $i_v \equiv i_m \pmod{N_{min}}$

  where N_v is the number of values of this type in the sentence; N_m is the number of mentions of this type in the sentence; $N_{min} = \min(N_v, N_m)$; i_v is the index of the value in the list of values in the sentence (of the same type); and i_m is the index of the mention in the list of mentions (of the same type) in the sentence.

  As an example, consider a sentence with 6 event rates and 3 group mentions ($6 \equiv 3 \pmod 3$). If the value-mention pair in question is the 5^{th} event rate in the sentence and the 2^{nd} group mention in the sentence, then the pair has the same order since $5 \equiv 2 \pmod 3$. However, if the pair consists of the 5^{th} event rate and the 3^{rd} group, then the pair do not have the same order since $5 \not\equiv 3 \pmod 3$.

  If the number of values and mentions are not congruent modulo the length of the smaller list, e.g. $N_v = 7$ and $N_m = 2$, then the pair are not considered to have the same order.

6.3.2.2 Finding the optimal set of associations. The system finds the optimal set of value-mention assignments for a sentence using the *Hungarian method*. The Hungarian method was developed by Kuhn [20] and Munkres [30] based on earlier work by Hungarian mathematicians König and Egerváry. The system uses a Python implementation of the Munkres algorithm[17].

The Hungarian method solves the *assignment problem* where there are n workers and n jobs. For each possible assignment of worker i to job j there is an associated nonnegative cost $c(i, j)$. The algorithm finds the optimal assignment of workers to jobs such that the sum of the assignment costs is minimized and each worker is assigned exactly one task.

Here, the assignment problem is assigning values to mentions. The cost of assigning value i to mention j is the probability that the pair should be associated. Since the Hungarian method finds the set of assignments with minimum total cost and we want the set of associations with maximum total cost (probability), the cost function we actually use is $c(i, j) = 1.0 - P(\langle i, j \rangle)$, where $P(\langle i, j \rangle)$ is the probability that value i should be associated with mention j as determined by the MaxEnt classifier. For sentences where the number of values and mentions are not equal, additional dummy values or mentions are created and their pair probabilities are set to zero.

6.3.2.3 Associating group sizes with groups. For group sizes that are not linked with outcome numbers, they must be associated with groups individually. The association step is performed now so that the sizes of group entities are known when associating outcome numbers with groups and outcomes.

As previously mentioned, the task of associating group size values with group

[17]http://software.clapper.org/munkres/

mentions in a sentences is treated as an instance of the assignment problem. Association probabilities are computed for each possible ⟨group size, group⟩ pair in the sentence using methods described in Section 6.3.2.1. The set associations with maximum sum of pair probabilities is found using the Hungarian method as described in Section 6.3.2.2. All associations with probability less than 0.5 are discarded. All other associations are kept. This pruning of ⟨group size, group⟩ associations takes care of low confidence associations, as well as any dummy group size values or group mentions added to create a $n \times n$ square cost matrix for the assignment algorithm.

6.3.2.4 Associating outcome measurements with groups and outcomes. In order to compute absolute risk reduction (ARR) values for groups and outcomes, the system needs to identify the group and outcome that each outcome measurement is recorded for. As with associating group sizes with groups, this task is viewed as an assignment problem. However, now the goal is to associate outcome measurements with two types of mentions (groups and outcomes). We wish to find the set of ⟨outcome measurement, group, outcome⟩ associations such that the sum of their association probabilities is maximized. More formally, if M, G, O are the sets of outcome measurements, group mentions and outcome mentions in a sentence respectively, then $A = M \times G \times O$ is the set of all possible ⟨outcome measurement, group, outcome⟩ associations. We wish to find the set of associations $B \subset A$ that maximizes

$$\sum_{\langle m,g,o\rangle \in B} P(\langle m, g, o\rangle), \tag{6.3}$$

where $P(\langle m, g, o\rangle)$ is the probability that outcome measurement m should be associated with group mention g and outcome mention o. It is estimated from the individual group and outcome probabilities.

$$P(\langle m, g, o\rangle) \approx P(\langle m, g\rangle)P(\langle m, o\rangle) \tag{6.4}$$

If the outcome measurement m consists of both an outcome number n and event rate

e, then we approximate the individual outcome measurement association probabilities using the geometric means of the outcome number and event rate association probabilities.

$$P(\langle m, g\rangle) \approx \sqrt{P(\langle n, g\rangle)P(\langle e, g\rangle)} \tag{6.5}$$

$$P(\langle m, o\rangle) \approx \sqrt{P(\langle n, o\rangle)P(\langle e, o\rangle)} \tag{6.6}$$

The use of the geometric mean instead of the arithmetic mean gives more weight to lower confidence associations. If the outcome measurement only consists of an event rate or an outcome number, then the outcome measurement association probabilities $P(\langle m, g\rangle)$ and $P(\langle m, o\rangle)$ are simply the individual association probabilities of the event rate or outcome number with the group and outcome mentions.

The individual association probabilities $P(\langle n, g\rangle)$, $P(\langle e, g\rangle)$, $P(\langle n, o\rangle)$ and $P(\langle e, o\rangle)$ are estimated using the MaxEnt classifier described in Section 6.3.2.1. Separate models are learned for each of the four pair association types.

After the association probabilities are computed, the Hungarian method introduced in Section 6.3.2.2 is used to find the optimal set of of ⟨outcome measurement, group, outcome⟩ associations. The cost for a potential association $\langle m, g, o\rangle$ is complement of the association probability.

$$C(m, \langle g, o\rangle) = 1.0 - P(\langle m, g, o\rangle) \tag{6.7}$$

The procedure for computing the cost matrix C and determining the final associations is outlined in as follows.

1. *Create ⟨group, outcome⟩ pairs.* Create a ⟨group, outcome⟩ pair for each group and outcome entity mentioned in the sentence. Discard any pairs where the tokens of one entity are identical to the other as this is most likely an error made

by either the group or outcome token classifier during the mention extraction stage.

2. *Pair unmatched outcome numbers and event rates.* Look for outcome numbers and event rates that could refer to the same outcome measurement that were not already paired during the ⟨outcome number, event rate⟩ association stage described in Section 6.3.1.2.

    - Keep a list of potential matches for each outcome number and event rate.
    - If the event rate calculated from an outcome number is equivalent to an event rate reported in the text (as defined in Section 6.3.1.2), add the outcome number to the event rate's match list and the event rate to the outcome number's match list.
        - If the outcome number has an associated group size, use that to calculate the event rate.
        - Otherwise, check the list of group entities mentioned in the sentence. If one of them has a group size that results in an equivalent event rate, add the outcome number and event rate to each other's match lists.
    - Discard problematic, potentially redundant or useless outcome numbers.
        - Outcome numbers that match multiple event rates.
        - Outcome numbers whose matching event rate is also a potential match for other outcome numbers.
        - Outcome numbers that cannot be used to calculate an event rate, i.e. they do not have an associated group size and there are no group sizes associated with any group entities that are mentioned in the sentence.
    - Link ⟨outcome number, event rate⟩ pairs that only have each other as matches. Unite them into one outcome measurement.

3. *Compute* $P(\langle m, g, o\rangle)$. For each possible outcome measurement association estimate the probability $P(\langle m, g, o\rangle)$ that outcome measurement m should be associated with group g and outcome o.

   (a) If m contains both an outcome number n and event rate e, then

      i. If n does not have an associated group size and g has an associated group size and the resulting event rate is not equivalent to e

         A. Then $P(\langle m, g, o\rangle) = 0$ (no event rate can be calculated)

         B. Else, $P(\langle m, g, o\rangle) = P(\langle n, g\rangle)P(\langle e, g\rangle)P(\langle n, o\rangle)P(\langle e, o\rangle)$

   (b) Else if m contains only an outcome number n

      i. If n has an associated group size or g has an associated group size,

         A. Then $P(\langle m, g, o\rangle) = P(\langle n, g\rangle)P(\langle n, o\rangle)$

         B. Else, $P(\langle m, g, o\rangle) = 0$ (no event rate can be calculated)

   (c) Else m must consist soley of event rate e

      i. $P(\langle m, g, o\rangle) = P(\langle e, g\rangle)P(\langle e, o\rangle)$

4. *Compute* $C(m, \langle g, o\rangle)$. For each possible outcome measurement association compute the assignment cost $C(m, \langle g, o\rangle) = 1.0 - P(\langle m, g, o\rangle)$.

5. *Associate*. Use the Hungarian method as described in Section 6.3.2.2 to find the optimal set of ⟨outcome measurement, group, outcome⟩ associations. Discard associations that have zero probability.

6.4 Calculating summary statistics

After identifying the relationships that exist between each of the extracted elements, the system needs to determine if it has enough information to calculate absolute risk reduction (ARR) and number needed to treat (NNT) statistics.

For each sentence, the system pairs outcome measurements that are linked to the same outcome cluster. For each pairing, ARR, NNT and their confidence intervals are computed using Equations 1.1 - 1.3. If there are more than two groups, the system computes summary values for each possible pairing.

At this point the system does not attempt to identify the experiment and control groups. In many cases the role of the treatment group in the study is not specified in the abstract. Therefore, for now, when calculating summary measures, the system identifies the *more effective* and *less effective* treatment groups for an outcome and calculates values using this distinction. Instead of the difference between the event rates of the control and experimental groups, ARR now becomes the difference between the event rates of the *less effective* (LER) and *more effective* (MER) treatment groups for the outcome.

$$\text{ARR} = \text{LER} - \text{MER} = \frac{N_{less}^{bad}}{N_{less}} - \frac{N_{more}^{bad}}{N_{more}} \tag{6.8}$$

When comparing the effectiveness of two treatment groups for a given outcome, the *more effective* group is the group with the *lower* bad outcome event rate. If the abstract phrases the outcome as *good*, some result that the treatment should increase, then the more effective treatment group is the one with the *higher* good outcome event.

6.4.1 Classifying outcomes. Most outcomes mentioned in abstracts are phrased as *bad* outcomes, i.e. some event that the treatment should reduce or prevent, such as stroke, heart attack or death. In some cases, the outcome is phrased as a *good* outcome that the treatment should increase in the population. Good outcomes could be negated bad outcomes such as "not die" or "did not develop malaria." They could also be non-negated phrases that describe a positive outcome such as "cured" or "lost weight."

In order to identify the more and less effective treatment groups for an out-

Table 6.5. Common lemmas that indicate a problem or recovery.

Problem				Recovery	
adverse	sick	injury	recurrence	recovered	cure
die	death	mortality	incidence	quit	stop
problem	condition	disease			

come, the system needs to identify the *polarity* of the outcome phrase, i.e. whether the phrase describes a good or bad outcome event. To determine the polarity of an outcome mention, the system uses the following set of rules.

1. Does the outcome contain a word whose lemma appears in a list of common *bad* outcome lemmas? If so, does the outcome also contain a *negation* word?

2. Does the outcome contain a word whose lemma appears in a list of common *good* outcome lemmas? If so, is the outcome free of negation words?

Table 6.5 provides a list of common lemmas that indicate a *problem* (bad outcome) or imply *recovery* (good outcome). The list of negation words is given in Table 5.11.

6.4.2 Locating group size information. Group size information is often found in the same sentence as the outcome number. However, in some cases, the group size is mentioned an earlier sentence. When the size of a group is needed in order to calculate an outcome event rate, the system first checks if there is a group size associated with the outcome number. If so, then the system uses this group size value. If not the system looks for the most *salient* group size, that is the most recent preceding group size associated with the outcome number's group. It begins its search at the outcome number and works backwards toward the beginning of the sentence. If

unsuccessful, the search moves up to the end of the previous sentence and continues. The search process ends once a size for the group is found or the start of the abstract has been reached.

6.5 Compiling summaries

After all of the key trial information has been identified, clustered and associated, the final summaries must be generated. Summary creation is treated as a task of filling slots in a summary template. The summaries have a form similar to Figure 1.2, so the have slots for the following information.

- *Population age information.* Minimum, maximum, mean and/or median age of trial population.
- *Conditions.* List of medical conditions common to the trial population.
- *Groups.* List of treatment group names.
- *Outcomes.* List of outcomes evaluated in the trial and *summary statistics* for each outcome.

The system creates lists of each element type as the elements are identified, clustered and associated. Filling slots in a summary template is a matter of processing the lists and identifying a representative name for the condition, group, and outcome clusters that have already been computed. If a cluster consists of a single mention, the mention is placed in the appropriate slot in the summary. For clusters that consist of multiple mentions, the representative mention, as described in Section 6.2, is chosen to represent the cluster. The system outputs the final summaries in XML format so they are easily machine readable.

6.6 Contributions and prior work

This chapter presents the first known approach for automatically detecting the relationships between detected mentions and quantities so that summary statistics may be calculated. It also presents novel approaches for identifying the polarity of outcomes.

There has been little prior work focused on the tasks addressed in this chapter. Niu and Hirst[32] present a method for identifying the polarity of sentences that summarize the main clinical outcomes of a trial. They try to determine if a sentence is stating whether the result of the study is good (experimental treatment is effective) or bad (experimental treatment was not effective).

CHAPTER 7

EVALUATION

This chapter provides evaluations of each of the key parts of the summarization system. In addition to evaluations based on automated comparisons with ground truth, this chapter also contains human evaluations provided by EBM experts. The expert evaluations provide an indication of the clinical usefulness of the summaries.

7.1 Methodology

The summarization system was developed, trained and tested on the corpora of medical abstracts described in Chapter 4. The system was developed and optimized using the combined *BMJ* and *Cardio* corpora (*BMJCardio*). For the evaluations described in this chapter, the summarization system was then trained using the entire *BMJCardio* corpus and it was tested on the *Ischemia* corpus.

Performance measures for most of the system components are *recall*, *precision* and *F-score*.

- *Recall*. The percentage of target items that the system successfully found. It is defined as

$$R = \frac{N_{tp}}{N_{correct}} = \frac{N_{tp}}{N_{tp} + N_{fn}} \tag{7.1}$$

  where N_{tp} is the number of *true positives*, the number of items correctly identified; $N_{correct}$ is the total number of *correct* items that exist; and N_{fn} is the number of *false negatives*, the number of correct items that were *not found* by the system.

- *Precision*. The percentage of items found by the system that are actually correct. It is defined as

$$P = \frac{N_{tp}}{N_{found}} = \frac{N_{tp}}{N_{tp} + N_{fp}} \tag{7.2}$$

where N_{tp} is the number of *true positives*; N_{found} is the total number of items *found* by the system; and N_{fp} is the number of *false positives*, the number of *incorrect items* that were returned by the system.

- *F-score* (or *f-measure*). This is the harmonic mean of recall and precision. It is defined as

$$F = \frac{2 \cdot R \cdot P}{R + P}. \tag{7.3}$$

These measures are commonly used in information retrieval and information extraction work. They are useful when the number of target items that need to be identified or found is very small compared with size of the data set, as is the case for all of the elements that the system needs to identify and return.

7.2 Element extraction

This section evaluates the performance of the element extraction phase of the system. It looks at how well the system is able to identify sections of text that correspond to condition, group, outcome, group size, outcome number event rate and age values.

7.2.1 Baseline comparison. For a baseline comparison, conditions, groups and outcome mentions were extracted using the biomedical named entity recognizer BANNER that has been shown to be effective at identifying treatment and disease mentions [24]. The same parameter settings are used as those described in [24].

The baseline comparison for extracting group size, outcome number and event rates uses a rule-based approach consisting of the following rules:

1. If a sequence of sentence tokens matches "n = $INTEGER$", then label the integer as a group size.

2. If a sequence of sentence tokens matches "$INTEGER_1$ / $INTEGER_2$" or "$INTEGER_1$ of $INTEGER_2$", then if $INTEGER_1 \leq INTEGER_2$ label the $INTEGER_1$ as a outcome number and $INTEGER_2$ as a group size.

3. Label *all* percentages as event rates.

7.2.2 Evaluation criteria. Evaluating the accuracy of the number extraction part of the system is relatively straightforward, since numbers consist of only a single token. If a detected number has the same annotation as the label assigned by the classifier, it is a true positive, otherwise it is a false positive.

Evaluating detected condition, group and outcome mentions, on the other hand, is problematic. The difficulty is that mention boundaries are often ambiguous. Consider the following clause.

62 children developed kwashiorkor (defined by the presence of oedema)

Here the outcome is kwashiorkor (a form of protein-energy malnutrition). However, "developed kwashiorkor", "kwashiorkor (defined by the presence of oedema)" or even possibly "the presence of oedema" could be all be considered acceptable. One method for handling this situation is to annotate all acceptable versions of an entity (e.g. annotate all three possibilities in this case). However, annotating every possibility can greatly add to the complexity of the annotation process and the annotator may miss some acceptable versions of the entity. An alternative approach is to relax the criteria for determining when an entity recognized by the system matches an annotated entity in the corpus. This is the approach used for evaluating detected mentions. A detected mention is considered a match for an annotated mention if they consist of the same set of words (ignoring "a", "an", "the", "of", "had", "group(s)", and "arm") or if the detected mention overlaps the annotated one and the overlap is not a symbol or stop

word. If a detected mention overlaps multiple annotated mentions, it is considered to be a false positive. If multiple detected mentions overlap the same annotated mention the detected mention with the most overlapping tokens (not counting symbols and stop words) is considered to be a true positive and the others are counted as false positives. Annotated mentions that do not match detected mentions are considered to be false negatives.

7.2.3 Results. Recall, precision and F-score for extracted mentions are given in Table 7.1. From these results it appears that the system outperforms the baseline (BANNER) overall. While BANNER has higher precision for group and outcome mentions, its recall is significantly lower.

Table 7.1. Recall, precision and F-score for the summarization system and baseline system for extracted condition, group and outcome mentions.

	Conditions			Groups			Outcomes		
	R	P	F	R	P	F	R	P	F
ACRES	**0.41**	**0.60**	**0.49**	**0.80**	0.80	**0.80**	**0.60**	0.51	0.55
No post-process	**0.41**	**0.60**	**0.49**	0.60	0.81	0.69	0.56	0.60	**0.58**
No boosting	**0.41**	**0.60**	**0.49**	**0.80**	0.80	**0.80**	0.54	0.54	0.54
No boost & no post	**0.41**	**0.60**	**0.49**	0.60	0.81	0.69	0.51	**0.64**	0.57
BANNER	0.37	0.56	0.44	0.51	**0.85**	0.64	0.43	**0.58**	0.50
BANNER w/post	0.37	0.56	0.44	0.71	0.83	0.77	0.49	0.51	0.50

Table 7.1 also contains results from variants of the system without key components intended to increase recall of group and outcome mentions. The post-processing step described in Section 5.5 significantly increases recall for group mentions with only

a slight decrease in precision. A similar result occurs when the same post-processing rules are applied to output from BANNER. Post-processing shows a more modest increase in recall for outcome mentions along with a greater decrease in precision. Again this result is also seen with output from BANNER. As the purpose of the post-processing stage is to increase recall for group and outcome mentions in order to ultimately calculate more ARR values, the system does not contain any post-processing rules for condition mentions. Hence, results for condition mentions are unaffected by the presence or absence of the post-processing stage.

Although, the post-processing stage performs various tasks to cleanup detected mentions such as discarding mentions that only consist of stop words and resolving group/outcome conflicts. However, the key component of this step is the algorithm that searches for unlabeled matches for detected mentions and labels them. Without this step, group results are nearly identical to those achieved without any post-processing (recall 59%, precision 81%). The step that looks for and labels "*X* group" phrases does not increase recall as the classifier is already capable of identifying this type of mention.

In addition to post-processing, the complete system also features a novel technique for boosting recognition of outcome mentions that uses alternate CRF labelings as described in Section 5.4. When comparing the effects of post-processing with those of outcome boosting, it appears that outcome boosting leads to a slightly greater increase in outcome recall and a smaller decrease in precision. When combined, both methods result in an substantial increase in outcome recall at the cost of lower precision. While this leads to a small decrease in F-score, the higher recall allows more ARR values to be calculated (32% versus 18%) with only a small decrease in precision (64% versus 67%).

Even with the same post-processing, results from BANNER are lower than those achieved with the system with or without outcome boosting. Both methods use Mallet toolkit implemented conditional random field (CRF) models to label tokens in a sentence[18]. The key difference between the two approaches is the features used by the models. BANNER uses a combination of orthographic (e.g. capitalization, letter digit combinations), morphological (e.g. 2-4 character prefixes and suffixes) and shallow syntax features (e.g. lemma, part of speech tags). It does not use semantic features, features that use deeper syntactic information from full parses of the sentence or features from neighboring tokens in the sentence. Since the system uses these additional features, we look at the impact of each of these types of features. Table 7.2 shows the performance of the mention extractor with different feature sets, without boosting or post-processing. It gives results for all features, variants omitting each category of features described in Section 5.3.2 and the system with only features based on the word (lexical) and its neighboring tokens (token context).

Overall, features that describe the token being classified (token features) and the token and semantic features from its neighboring tokens (token context), appear to provide the most benefit. Surprisingly semantic and syntactic features do not appear to add much benefit, particularly for condition and outcome mentions. However, semantic features do appear to be useful for identifying groups. While there is a semantic tag for common group words, omission of this feature only reduces recall to 58% versus a recall of 54% that occurs without any semantic features. Hence, the semantic features overall do provide useful information for groups. Syntactic features primarily consist of token and semantic features for tokens that share dependency relationships for the token in question. A pitfall with these features is that they are sensitive to parse errors. The creators of BANNER specifically avoided syntactic

[18]BANNER uses Mallet version 0.4. The system uses version 2.0.7

features that rely on parse information [24] for this reason.

Table 7.2. Recall, precision and F-score for condition, group and outcome mention extractors with different feature sets.

	Conditions			Groups			Outcomes		
	R	P	F	R	P	F	R	P	F
All features	0.41	0.60	**0.49**	**0.60**	0.81	**0.69**	0.51	0.64	0.57
No token features	0.39	0.56	0.46	0.53	0.76	0.62	0.44	0.58	0.50
No semantic features	0.41	0.62	**0.49**	0.54	0.84	0.66	0.51	0.63	0.56
No syntactic features	0.41	0.60	**0.49**	0.59	0.82	**0.69**	0.51	0.63	0.56
No acronym features	**0.42**	0.58	**0.49**	0.59	0.82	**0.69**	**0.52**	**0.65**	**0.58**
No sentence features	0.39	0.60	0.48	0.58	0.82	0.68	0.51	**0.65**	0.57
No token context	0.40	0.53	0.46	0.57	0.78	0.66	0.51	0.57	0.53
Token+context	0.35	0.60	0.44	0.51	**0.87**	0.64	0.44	0.64	0.52
Token+con+sem	0.38	0.60	0.47	0.58	0.82	0.68	0.47	0.63	0.54
Token+con+syn	0.39	**0.63**	0.48	0.52	0.86	0.65	0.48	0.64	0.55
Token+con+sem+syn	0.39	0.59	0.47	0.59	0.82	0.68	0.50	**0.65**	0.56
Token+con+sentence	0.41	0.63	**0.49**	0.52	**0.87**	0.65	0.50	0.62	0.55

If we look at classifier performance with a reduced feature set that only contains token and token context features, then observe recall and precision as additional types of features are added, we see that semantic, syntax and sentence features do improve performance. Looking at Table 7.2, it appears that syntax and sentence fea-

tures are particularly useful for condition and outcome mentions. Semantic features provide more benefit to group mentions than condition and outcomes.

A consideration when examining the effect of semantic and syntactic features is that both categories add a significant number of features. When condition and outcome classifiers are trained on the BMJCardio corpus, there are 6833 semantic features and 27,576 syntactic features (which include 12,247 semantic features for tokens sharing dependency relationships). For the group classifier which is trained on fewer sentences the numbers are lower (4847 semantic, 18,073 syntactic which includes 8504 additional semantic features). By comparing the effect of omitting semantic or syntactic features with their contributions obtained when added to a system with a reduced feature set (token+context), it appears that more training data is needed to take full advantage of semantic and syntactic features.

Detection results for numbers extracted by the system are given in Table 7.3. As with mentions, the summarization system appears to outperform the baseline approaches overall. While the baseline is able to identify nearly all event rates (it only missed two event rates), its precision is much lower. The system significantly outperforms the baseline in terms of precision and especially recall for identifying group sizes and outcome numbers. The baseline system only identifies 15% of outcome numbers, while the system is able to identify 83%.

To examine the impact of each type of feature, Table 7.3 also contains results for variants of the system with the omission of each category of feature described in Section 5.3.2. As with mention recognition, token and token context features appear to be most useful for identifying group sizes, outcomes and event rates. These features are more useful than the common pattern features which are used in the baseline for group sizes and outcome numbers. In fact, the group size/outcome number classifier

achieves similar performance without semantic features which include the common numeric pattern features.

Table 7.3. Recall, precision and F-score for the summarization system, baseline system and system variants with different feature sets for extracted group size, outcome numbers and event rates.

	Group sizes			Outcome numbers			Event rates		
	R	P	F	R	P	F	R	P	F
ACRES	0.70	0.74	0.72	0.83	0.74	**0.78**	0.95	0.88	0.91
Baseline	0.59	0.70	0.64	0.15	0.50	0.23	**0.99**	0.79	0.88
No token features	0.67	0.63	0.65	**0.90**	0.58	0.70	0.92	0.86	0.89
No semantic features	0.70	0.78	0.74	0.82	**0.75**	**0.78**	0.95	0.88	0.91
No syntactic features	0.72	0.74	0.73	0.82	**0.75**	**0.78**	0.93	0.87	0.90
No sentence features	0.66	0.76	0.70	0.85	0.72	**0.78**	0.96	**0.89**	**0.92**
No token context	0.67	0.65	0.66	0.86	0.57	0.68	0.93	0.87	0.90
Token+pattern	0.56	**0.79**	0.66	0.33	0.37	0.35	**0.99**	0.84	0.91
Token+context	0.66	0.78	0.72	0.82	0.72	0.76	0.96	0.86	0.91
Token+con+sem	0.67	**0.79**	0.73	0.85	0.69	0.76	0.94	0.87	0.90
Token+con+syn	0.64	0.78	0.71	0.83	0.72	0.77	0.97	0.87	**0.92**
Token+con+sentence	**0.75**	0.77	**0.76**	0.79	0.74	0.76	0.95	0.87	0.91

Token features, which contain a feature for whether the number is a percentage or not, are the most important features for recognizing event rates. These features

alone are sufficient for the classifier to achieve over 99% recall for event rates (only 3 false negatives). Additional features improve precision at the cost of recall. All event rates in the BMJCardio corpus and all but two in the Ischemia corpus are reported as explicit percentages in the text (e.g. "35%"). However, this rule alone is not enough as only 79% of percentages in the Ischemia corpus are event rates. In BMJ, 71% are event rates. In Cardio, 78% are event rates. Sentence features (label of section containing the sentence) appear to be the only features that do not provide increased recall or precision. The sections containing event rates also contain the majority of non-event rate percentages. Hence, the section label does not provide discriminative information for event rates. However, the section label is useful for group sizes which are primarily found in sections describing the study (e.g. "Methods") and results (e.g. "Results", "Findings") and integers appear in a wider variety of sections than percentages.

7.3 Mention clustering

The clusters of similar condition, group and outcome mentions produced by the system are compared with a those produced using a baseline approach.

7.3.1 Baseline comparison. The baseline approach clusters mentions that are exact matches of each other (contain the same set of tokens), ignoring word order.

7.3.2 Evaluation criteria. Since mention clustering is a form of coreference resolution, recall and precision for mention clusters are computing using the B-cubed algorithm [2] that is commonly used for evaluating coreference resolution results. With the B-cubed algorithm, detected clusters are compared against true clusters of annotated mentions. Recall is the weighted sum of recall scores for each mention m:

$$R_m = \frac{\text{number of correct mentions in detected cluster containing } m}{\text{number of mentions in the true cluster containing } m}. \tag{7.4}$$

Precision is the weighted sum of the precision scores for each mention m:

$$P_m = \frac{\text{number of correct mentions in detected cluster containing } m}{\text{number of mentions in the hypothesis cluster containing } m}. \quad (7.5)$$

The weights for each R_m and P_m are $1/N$, where N is the total number of *detected* mentions in the *test set*. With these weights the overall recall and precision is the average of R_m and P_m for all m.

$$R = \sum_m \frac{1}{N} R_m = \frac{1}{N} \sum_m R_m \quad (7.6)$$

$$P = \sum_m \frac{1}{N} P_m = \frac{1}{N} \sum_m P_m \quad (7.7)$$

7.3.3 Results. Results from a baseline comparison of the mention clustering stage are given in Table 7.4. The system has better recall whereas baseline shows higher precision for both condition and group mentions. Precision is similar for both approaches for outcome mentions since the system uses a more conservative merge criteria. Higher precision implies that the mention clusters contain a higher percentage of correct mentions that refer to the same entity. The baseline uses a more conservative merge criteria that only merges mentions that are identical. While this policy results in clusters that are more "pure", it leads to numerous small clusters that should be merged. If a mention differs from existing clusters by a single token, the baseline creates a new cluster for it. By using a more relaxed merge criteria, the system is able to achieve higher recall for condition and group clusters. Higher recall implies that the average mention cluster contains more of the mentions that refer to the same entity. For groups high recall is important since group sizes are often mentioned in early sentences to those containing outcome numbers. Effectively clustering group mentions allows the system to use group sizes reported in previous sentences to be used with outcome numbers to compute ARR values.

The majority of clustering errors made by the system result from including erroneous mentions in clusters. These false positive mentions are usually mentions

that either should have been annotated or erroneous mentions that happen to contain some of the same words as other mentions in the cluster. There are also cases where mentions for similar true entities are incorrectly merged. The system incorrectly includes "twice-daily *2.5*-mg dose" and "twice-daily *5*-mg dose" in the same cluster since there is non-trivial overlap between the mentions. A related issue occurs when larger mentions overlap with multiple different smaller mentions. This can be the result of errors at the mention extraction stage or it can inherent in the abstract. An abstract may report results for individual and composite outcomes (e.g. "ischemia", "stroke" and "ischemia or stroke"). Some studies compare treatments consisting of multiple interventions (e.g. "usual care" vs. "acupuncture" vs. "usual care plus acupuncture"). Another problem for both approaches is matching generic group references such as "the experimental group" and "the control group" with their more explicit mentions such as "aspirin" and "placebo". Neither approach is capable of handling these cases.

Table 7.4. Recall, precision and F-score for the summarization system and baseline system for clustering detected condition, group and outcome mentions.

	Condition			Group			Outcome		
	R	P	F	R	P	F	R	P	F
ACRES	**0.84**	0.76	0.80	**0.89**	0.83	**0.86**	**0.85**	0.86	**0.85**
Baseline	0.77	**0.94**	**0.84**	0.69	**0.90**	0.78	0.84	**0.87**	**0.85**

7.4 Value association

This section evaluates the system's ability to associate group sizes with group mentions and outcome measurements with group and outcome mentions. An outcome measurement can consist of any of the following:

- outcome number and group size,
- event rate extracted from the text,
- outcome number, group size and event rate extracted from the text.

7.4.1 Baseline comparison. Results from the system are compared with baseline approaches. For a baseline comparison, extracted group sizes are associated with the nearest extracted group mentions in the same sentence. Here, the nearest mention is the mention that is separated from the value by the fewest number tokens. Only mentions that appear in the same sentence as the value are considered. If two mentions are the same number of tokens from that same value, the one that appears before the value in the sentence is selected.

For associating outcome measurements, the baseline performs the following sequence of steps for each sentence in a given abstract.

1. For each event rate and outcome number in the sentence, find the nearest group and outcome mentions in the sentence.
2. Build a list of event rates (if any) that are associated with both a group and an outcome. If there are multiple event rates associated with the same ⟨group, outcome⟩ pair, only keep the event rate that is closest to one of its mentions. Ties are resolved by taking the event rate that has the fewest total tokens between it and its mentions. If ties still remain, then it is unclear which should be kept, so they are all discarded.
3. Build a list of outcome numbers (if any) that are associated with both a group and an outcome *and* can be used to calculate an event rate, i.e., either they have an associated group size or their groups have one.

4. For any outcome numbers associated with the same ⟨group, outcome⟩ pair as a detected event rate, check if they have equivalent event rates as defined in Section 6.3.1.2. If so, link the outcome number and event rate. Otherwise, if they are not compatible, determine which one is closer to its mentions using the procedure previously described for multiple event rates associated with the same ⟨group, outcome⟩ pair. The same procedure is used to handle cases where multiple outcome numbers are associated with the same ⟨group, outcome⟩ pair.

5. Create list of outcome measurements from final lists of event rates and outcome numbers. There should be at most one outcome measurement for each ⟨group, outcome⟩ pair.

7.4.2 Evaluation criteria. Here we are concerned with the system's ability to find the correct associations (or no-association) between *detected* values and mentions that exist *within* the sentence. Not only can the element extraction stage miss values or mentions, making correct association impossible, but some sentences report outcome results without explicitly mentioning one or any of the groups. The value group mention relationship is implied from the order of a previous sentence.

> Rates of PCI or coronary artery bypass surgery were 12.7% and 10.6% , respectively (p = 0.30) .

The system does not currently support sentences that omit references to groups or outcomes. The omission of outcome mentions is rare compared with the omission of groups.

For association, *recall* is the percentage of detected values that are associated with the correct detected mention (when a correct association exists). *Precision* is the percentage of detected value-mention associations that are correct. Recall and

precision are computed using the following definitions for true positive, false positive, true negative (correct no-association), and false negative.

- *True positive.* The detected value is associated with the correct detected mention. Both the value and mention are considered to be true positives using the criteria defined in Section 7.2.
- *True negative.* No correct association to be made and none found.
  - Either the value is a false positive
  - Or the sentence does not contain a detected mention that should be associated with the value.
- *False positive* - An association where one of the two cases is true.
  - Either the value or the mention are false positives.
  - Both are true positives, but the pair should not be associated.
- *False negative.* There was an association that should have been made, but it was missed. Either an incorrect one was found, or no association was made.

These definitions are used for both ⟨group size, group⟩ associations as well as ⟨outcome measurement, group, outcome⟩ associations. The only differences for outcome measurements appear when the text contains both an outcome number and an event rate for the same outcome measurement. If only one form of the measurement is detected (either outcome number or the event rate) and the other is missed, then the outcome measurement association is still considered to be correct as long as the group and outcome associations are correct.

7.4.3 Results. Table 7.5 shows results of a baseline comparison for the association stages of the system. Although one of the highest weighted features for associating

mentions with values is whether a mention is the closest one to the value, results for the baseline show that this information alone is not sufficient to associate mentions with values. There are sentences such as the following example where the size of the treatment group "SES" (335) is closer (in terms of tokens) to the group mention "Dual-DES". Similar situations arise for outcome numbers and event rates.

> A total of 1007 patients undergoing coronary stenting of de novo lesions in native vessels were randomized to treatment with SES (n = 335) , Dual-DES (n = 333) , or ZES (n = 339) .

Errors in mention and value extraction stages are common source of association errors. A source of error that persists even with perfect detection occurs when values are recorded for the same group-outcome pair at different follow-up times in the same sentence. At this point the system does not support follow-up times, so their presence can lead to association errors since there are multiple measurement values for the same group-outcome pair.

The way sentences report results can be complicated and cause problems for association. Some or all group mentions may precede the values and association is implied by the order the groups are listed.

> The incidence of stent thrombosis was significantly lower in the SES group (ZES versus SES versus PES , 0.7% versus 0% versus 0.8% , respectively , p = 0.02) .

While the system has a feature to capture the order that mentions and values are listed, it is confounded in this example by the additional mention "the SES group" which makes it difficult to align group mentions with event rates. In some cases a deeper understanding of the sentence is necessary to associate values and mentions. In the following example, the relationship between the groups and event rates cannot be determined just by proximity or the order groups are mentioned.

> At 12 months , the ZES group showed noninferior rates of MACE compared with the SES group (10.2% versus 8.3% , p for noninferiority = 0.01 , p for superiority = 0.17) and significantly fewer MACE than the PES group (10.2% versus 14.1% , p for superiority = 0.01) .

There are three groups, but four event rates since the event rate for "the ZES group" (10.2%) is repeated when it is compared with two different groups, "SES" and "PES".

Table 7.5. Recall, precision and F-score for the summarization system and baseline system for associating detected group sizes with detected group mentions and detected outcome measurements with detected group and outcome mentions.

	(Group size, Group)			(Outcome measurement, Group, Outome)		
	R	P	F	R	P	F
ACRES	**0.68**	**0.71**	**0.69**	**0.77**	**0.65**	**0.71**
Baseline	0.66	0.44	0.53	0.37	0.47	0.42

7.5 Summary evaluation

The summary elements produced by the summarization system are compared here with those obtained using a baseline system with a similar architecture. Summary elements are all of the pieces of information that appear in the final summary. These include population age values, conditions, group names, outcomes and ARR values. *Detected summary elements* are those produced by the system and *annotated summary elements* are the ground truth elements produced from the annotated text.

7.5.1 Baseline comparison. Since this is the first known system designed to generate EBM-oriented summaries containing summary statistics, the baseline system uses the same architecture as the proposed system except it uses the previously

described baseline methods for extracting, clustering and associating key elements.

7.5.2 Evaluation criteria. Evaluating age values is similar to evaluating extracted numbers. If a detected age value, such as a mean or maximum age value, matches an annotated age value found in the text, then the value is a true positive, otherwise it is considered to be a false positive.

Condition, group and outcome summary elements are the representative mentions (see Section 6.2) for each mention cluster. Similar to evaluating detected mentions in Section 7.2, the cluster is considered to be a true positive if its *name* (the representative mention) matches one of the mentions in a true (annotated) mention cluster. If a detected cluster matches multiple annotated cluster it is considered a false positive. If multiple detected clusters match the same annotated cluster, the detected cluster that best matches the annotated cluster is considered a true positive and the others are considered false positives. Annotated clusters that do not match detected clusters are considered to be false negatives.

An ARR value computed from detected values (a *detected* ARR) is considered to be a true positive if the associated outcome and group names are considered correct for the values (using previously mentioned criteria for group and outcome elements) and the event rates and ARR value are "close enough" to the true value. The sum of the absolute difference between the detected and true event rates must be less than 0.1% and the computed ARR value must have the same sign as the true ARR value. Otherwise, the ARR value is counted as a false positive.

The problem with computing recall and precision for the total number of ARR values computed for a collection of abstracts is that some abstracts contain many potential ARR calculations and others contain few. A few problematic or easy abstracts have the potential to skew the numbers. Furthermore, discussions with EBM experts

revealed that correct ARR values are the most important component of a summary. Summaries without correct ARR values are less useful. In order to capture the percentage of useful summaries produced by the system, we calculate recall and precision for abstract summaries that the systems gets right (in terms of ARR calculations). Recall is the percentage of abstracts for which the system is able to generate summaries with correct ARR values. Precision is the percentage of summaries produced by the system that have correct ARR values. Since a summary that misses some ARR values, but successfully calculates others may still be potentially useful, we calculate recall and precision for different levels of correctness.

- *Exact.* All ARR calculations are correct. No erroneous ARR. No missing ARR.
- *Correct only.* No erroneous ARR. All ARR in summary are correct. It is okay if some are missing, as long as there is at least *one* correct ARR.
- *Any correct.* Summary has at least *one* correct ARR. Erroneous and missing ARR are acceptable.

7.5.3 Results. Table 7.6 contains results for age values that appear in summaries. In addition to the final age values in the summaries, the table also shows the system's performance for finding the original phrases from which the values are extracted. Unfortunately, the number of age values in the Ischemia corpus is limited. There are 17 annotated age phrases in the Ischemia corpus and they contain a total of 26 age values. In spite of the small sample size, there are still insights that can be gleaned from these results.

While the system shows low precision for identifying age phrases, it has high precision for identifying age values. Detected age phrases are considered to correct if they contain all of the age values in the annotated age phrase. Age values are

considered to be correct if they have the same value, units and error bounds as the annotated age values. The only two false positive age values result from the interpreting "2 to 6 days" as an age range in the following false positive age phrase.

> infarct size , expressed as % age of LV mass , assessed by cardiac magnetic resonance (CMR) imaging performed 2 to 6 days after study medication administration (first CMR) and again 12 ± 2 weeks later (second CMR).

The majority of false negatives result from detected age phrases that are missing age values or are missing key tokens needed to interpret the values. When the system is given true age phrases, it achieves perfect precision. The three remaining false negatives result from cases that do not match existing patterns such as the following sentence.

> A total of 744 patients, 64 years old (55 to 73 years old), 179 (24.1%) women, were enrolled.

In this case the system correctly found the age range "55 to 73 years old", but missed the average age ("64 years old") which was not explicitly described as the mean or median. In some abstracts, age values are reported for the individual groups such as the the following example.

> the median ages were 54 years (saxagliptin) and 55 years (comparator)

At this point, when the system encounters multiple age values of the same type and the values are identical, it discards all values of this type. Finally, the system only looks for phrases that contain potential age values. It will not detect phrases such as "middled aged" or "elderly participants" that describe the age of the participants without numeric values.

For the remaining summary elements, a performance comparison with the baseline system is given in Tables 7.7 - 7.9. Here recall is the percentage of true summary elements that the system was able to correctly identify; precision is the percentage of the summary elements identified by the system that are correct. The proposed system outperforms the baseline overall especially with calculating ARR values. However, our while our system has similar recall for groups its precision is much higher and recall for outcomes is significantly higher, allowing it to correctly identify the outcomes and groups that the detected values are associated with. This allows our system to calculate more ARR values than the baseline system.

Table 7.8 looks at the system's ability to correctly compute ARR values. It reports the total number of ARR values whose outcome and group mentions match mentions for true ARR values. This total includes erroneous ARR values as well correct values. Sign errors are the number matched ARR values that report the wrong group as the more effective one. A sign error can occur when an incorrect event rate is associated with a particular group and outcome this error results in the less effective group appearing as the more effective one. Another cause is incorrectly identifying the polarity of the outcome, i.e. interpreting the outcome as *bad* when it should be considered *good* or visa versa. An ARR value is considered to be *qualitatively correct* (QC) if at least one of the event rates is incorrect, but this does not result in a sign error. While qualitatively correct ARR values are incorrect, they do correctly communicate the more effective treatment. As Table 7.8 shows, the majority of ARR values, for which the true ARR value can be identified, are correct. A detected ARR value may not be matched with a true ARR value if one of the group or outcome mentions does not match the group or outcome mention of a true ARR value.

Table 7.8 provides results for summaries that contain at least one correct ARR values (*Any correct*); only correct ARR values (*Correct only*); and all ARR values

are correctly detected without any errors (*Exact*). *Exact* and *Any correct* measures provide an upper and lower bound on system performance. *Correct only* provides a more realistic idea of the percentage of abstracts for which the system generates useful ARR values. For abstracts that report outcome measurements, the system is able to calculate correct ARR values (no false positives) for a third of these abstracts. For summaries that contain ARR values, just over half do not contain any false positive ARR values.

Table 7.6. Recall, precision and F-score for finding age phrases and the resulting age values that appear in the summary.

	R	P	F
Age phrases	0.82	0.58	0.68
Age values	0.73	0.90	0.81
Age values (true phrases)	0.88	1.00	0.94

Table 7.7. Recall, precision and F-score for the summarization system and baseline system for summary elements.

	Conditions			Groups			Outcomes		
	R	P	F	R	P	F	R	P	F
ACRES	0.46	**0.59**	**0.52**	**0.78**	**0.81**	**0.80**	**0.66**	0.47	**0.55**
Baseline	**0.48**	0.46	0.47	0.76	0.60	0.67	0.54	**0.48**	0.51

Table 7.8. Correctly computing ARR values. Results reported for qualitatively correct ARR values interpreted as false positives and true positives.

	ARR				QC=FP			QC=TP		
	Matches	Correct	QC	Sign err.	R	P	F	R	P	F
ACRES	135	110	4	21	**0.30**	0.60	**0.40**	**0.31**	0.62	**0.41**
Baseline	36	26	2	8	0.07	**0.62**	0.13	0.08	**0.67**	0.14

Table 7.9. Finding summaries that contain at least one correct ARR value (Any correct); at least one correct and no incorrect ARR values (Correct only); and all correct ARR values and no errors (Exact). Results reported for qualitatively correct ARR values interpreted as false positives.

	Any correct			Correct only			Exact		
	R	P	F	R	P	F	R	P	F
ACRES	**0.54**	**0.84**	**0.66**	**0.32**	0.51	**0.40**	**0.18**	**0.28**	**0.21**
Baseline	0.17	0.60	0.26	0.16	**0.57**	0.25	0.07	0.27	0.12

7.6 Exact match criteria

The evaluations in this chapter use the relaxed *partial match* criteria described in Section 7.2.2 for determining if a detected condition, group or outcome mention matches an annotated one. Here we look at results from using a strict *exact match* criteria for determining if a detected mention matches an annotated one. With exact match, a detected mention is considered a match for an annotated mention if they consist of exactly the same set of words (ignoring "a", "an", "the", "of", "had", "group(s)", and "arm"). Exact match provides a conservative lower bound for system performance.

Tables 7.10-7.13 compare results from the complete system using partial and exact match criteria. Table 7.10 shows results for condition, group and outcome mention extraction. Results for condition, group and outcome summary elements are given in Table 7.11. Since the correctness of ARR values partly depends on the accuracy of the group and outcome mentions associated with the values, Tables 7.12 and 7.13 compare results for ARR values overall and summaries with correct ARR values. As expected, recall and precision are lower when the conservative exact match criteria is used to evaluate extracted mentions. Outcome mention boundaries are especially ambiguous. For instance, "myocardial infarction" and "the rate of myocardial infarction" may both be acceptable, but only one version is annotated. As a result, outcomes are affected more by the different matching criteria than conditions or groups.

Table 7.10. Comparison of mention extraction performance using partial match and exact match criteria.

	Conditions			Groups			Outcomes		
	R	P	F	R	P	F	R	P	F
Partial match	0.41	0.60	0.49	0.80	0.80	0.80	0.60	0.51	0.55
Exact match	0.33	0.48	0.39	0.67	0.67	0.67	0.36	0.31	0.34

Table 7.11. Comparison of summary element performance using partial match and exact match criteria.

	Conditions			Groups			Outcomes		
	R	P	F	R	P	F	R	P	F
Partial match	0.46	0.59	0.52	0.78	0.81	0.80	0.66	0.47	0.55
Exact match	0.36	0.46	0.40	0.69	0.72	0.71	0.46	0.33	0.39

Table 7.12. Comparison of ARR value performance using partial match and exact match criteria for group and outcome mentions associated with the values.

				ARR		
	Matches	Correct	Sign error	R	P	F
Partial match	135	110	21	0.30	0.60	0.40
Exact match	86	76	10	0.19	0.39	0.26

Table 7.13. Finding summaries that contain at least one correct ARR value (Any correct); at least one correct and no incorrect ARR values (Correct only); and all correct ARR values and no errors (Exact). A comparison of performance using partial match and exact match criteria for mentions associated with ARR values.

	Any correct			Correct only			Exact		
	R	P	F	R	P	F	R	P	F
Partial match	0.54	0.84	0.66	0.32	0.51	0.40	0.18	0.28	0.21
Exact match	0.37	0.58	0.45	0.17	0.26	0.20	0.09	0.14	0.11

7.7 Ceiling analysis

Tables 7.14 - 7.18 also contains results from a ceiling analysis of our system, which shows how much system performance improves when earlier stages in a pipeline-based system have no error. Here we compare the original system with systems where: there is perfect extraction of all mentions and numbers; there is perfect extraction of all information and perfect clustering of mentions; there is perfect extraction, clustering and association of mentions and numbers. This analysis shows that the most important area for improvement is extracting the mentions and numbers. This is not too surprising since extracting key information is the first stage in the pipeline and errors here adversely affect later stages. Improving the association between mentions and number can greatly increase the number of ARR values calculated by the system. While improving the clustering of mentions can help with identifying the condition, group and outcome summary elements, it does not have much of an effect on computing additional ARR values.

Table 7.14 compares performance of the system's clustering stage with the baseline clustering algorithm when clustering true mentions. Both methods achieve high precision with true mentions. Since the baseline only clusters mentions that are identical, it is able to achieve perfect precision with true mentions. With detected mentions, two mentions can be identical due to a boundary detection error that omits a distinguishing feature from one or both mentions. For instance, if the system misses the dosages from group mentions "the twice-daily 2.5-mg dose of rivaroxaban" and "5 mg of rivaroxaban" and only identifies "rivaroxaban" in both cases, then the two mentions will be incorrectly clustered by the baseline. Another source of precision error with detected mentions is false positive mentions. There is no cluster that should contain a false positive mention. However, while both mentions achieve high precision with true mentions, the system has significantly higher recall for conditions

and groups since it has a more relaxed clustering criteria for these mention types. The system's stricter clustering criteria for outcomes leads to only slightly higher recall than the baseline. With true mentions, there are more mentions to cluster for each entity. Small differences in wording between mentions that refer to the same entity result in the creation of redundant clusters when a stricter clustering criteria is used as is the case with the baseline and the system's approach for outcome mentions. This reduces recall overall since the recall of each cluster is lower.

Table 7.15 compares the effectiveness of the system's value-mention association stage with the baseline when they are given true values and perfectly clustered true mentions. While performance improves for both approaches, the system significantly outperforms the baseline. The system has comparable performance for both association tasks when working with detected values and mentions. However, with true values and mentions, the system's performance for associating group sizes and groups improves much more than for associating outcome measurements. Associating outcome measurements with both outcomes and group sizes is a more challenging task than associating group sizes with groups. Better detection of group sizes and groups is the key to improving group size association.

Table 7.14. Recall, precision and F-score for ACRES and baseline system for clustering detected condition, group and outcome mentions.

	Condition			Group			Outcome		
	R	P	F	R	P	F	R	P	F
ACRES	0.84	0.76	0.80	0.89	0.83	0.86	0.85	0.86	0.85
Baseline	0.77	0.94	0.84	0.69	0.90	0.78	0.84	0.87	0.85
ACRES w/perfect extract	0.93	0.97	0.95	0.87	0.98	0.92	0.72	1.00	0.84
Baseline w/perfect extract	0.67	1.00	0.80	0.60	1.00	0.75	0.71	1.00	0.83

Table 7.15. Recall, precision and F-score for ACRES and baseline system for associating true group sizes with true group mentions and true outcome measurements with true group and outcome clusters.

	(Group size, Group)			(Outcome measurement, Group, Outome)		
	R	P	F	R	P	F
ACRES	0.68	0.71	0.69	0.77	0.65	0.71
Baseline	0.66	0.44	0.53	0.37	0.47	0.42
ACRES w/perfect extract+clust	0.92	0.99	0.95	0.85	0.88	0.86
Baseline w/perfect extract+clust	0.69	0.69	0.69	0.36	0.60	0.45

Looking at Tables 7.16-7.18, we see results for a system with perfect extraction, clustering and association allows us to evaluate the system's ability to compute ARR values from outcome measurements associated with groups and outcomes. For the

majority of false positive ARR values, the system identified the wrong polarity for the outcome mention, resulting in the wrong group being considered more effective. These polarity errors result from the outcome mentions containing negated concepts assigned by MetaMap. For instance, MetaMap assigned the negated concept "Adverse Event" to the outcome "adverse cardiovascular events" in the following sentence excerpt.

> did not reduce infarct size and was associated with higher rates of adverse cardiovascular events

The presence of negated concepts in a mention negates the polarity of the mention. In this case the mention was considered to be *bad* since it contains the term "adverse". 98% of outcomes with ARR values are considered to be *bad*. Unfortunately, the system's current approach for identifying outcome polarity does not correctly identify any good outcomes and misclassifies three bad outcomes as good outcomes. A more sophisticated method for identifying the polarity of outcomes should be an area of future investigation.

Another source of false positive ARR values is abstracts that report collective results for multiple treatment groups. The combined group results appear to the system as outcome measurements from an additional group. These results then get compared with outcome results from the individual treatment groups.

Most of the false negative ARR values occur when outcome results are reported for multiple follow-up times in the same sentence. This situation accounted for 57% of false negatives. The system does not currently handle follow-up times. When it encounters multiple measurements for the same group-outcome pair, it discards all of them. A similar situation occurs in two abstracts where results from a single group are repeated when compared to results from other groups.

The system with perfect association includes outcome measurement associa-

tions that exist across sentences. That is, perfect association includes associations where the associated group or outcome mention does not appear in the same sentence as the outcome measurement. For 20% of outcome measurements, the group or outcome is not mentioned in the same sentence as the values, but in a previous sentence in the abstract. The increase in ARR performance that occurs when the system has perfect association illustrates the potential gains that could be made by extending association to include previous sentences.

Table 7.16. Ceiling analysis results for condition, group and outcome summary elements when there is perfect mention and number extraction and perfect extraction followed by perfect clustering.

	Conditions			Groups			Outcomes		
	R	P	F	R	P	F	R	P	F
ACRES	0.46	0.59	0.52	0.78	0.81	0.80	0.66	0.47	0.55
Perfect extraction	0.96	0.91	0.93	0.97	0.87	0.92	1.00	0.70	0.82
Perfect extract+clustering	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00

Table 7.17. The effect of perfect performance at each stage in the system on computing correct ARR values.

	ARR		
	R	P	F
ACRES	0.30	0.60	0.40
Perfect extraction	0.59	0.81	0.68
Perfect extract+clustering	0.60	0.81	0.69
Perfect extract+clust+assoc	0.87	0.96	0.91

Table 7.18. The effect of perfect performance at each stage in the system on generating summaries with correct ARR values.

	Any correct			Correct only			Exact		
	R	P	F	R	P	F	R	P	F
ACRES	0.53	0.83	0.64	0.32	0.51	0.40	0.18	0.28	0.21
Perfect extraction	0.79	0.91	0.85	0.63	0.73	0.68	0.43	0.49	0.46
Perfect extract+clustering	0.81	0.94	0.87	0.64	0.73	0.68	0.44	0.50	0.47
Perfect extract+clust+assoc	0.93	0.96	0.94	0.87	0.90	0.89	0.81	0.84	0.82

7.8 Boosting outcomes

In order to calculate ARR values for outcome, the system needs to identify outcome numbers, event rates, group sizes, group mentions and outcome mentions. While the system effectively identifies most of these key elements, the identification of outcomes remain a challenge. This section compares two different approaches for boosting the performance of outcome mention extraction.

Section 5.4 presents a novel approach that considers alternate CRF labelings from the outcome token classifier. This method is compared with an ensemble approach. The ensemble uses a committee of five CRF token classifiers trained on overlapping subsets of training data. Each classifier in the ensemble is trained on a random selection of abstracts from the training set. Abstracts are selected with replacement so the training set for a classifier may contain duplicates. The number of abstracts in each sets is 70% of the size of the entire training set. The size was selected empirically using the development corpus in order that each classifier in the ensemble be sufficiently trained to be effective, and yet ensure that there is sufficient

constructive disagreement between the classifier models. During development, it was found that increasing outcome recall is more critical for the calculation of ARR values than increasing precision. Although false positive outcomes are undesirable, they do not harm performance as much as false negatives. The association stage can potentially handle false positive outcomes by associating outcome measurements with true outcomes (with which they should ideally have higher pair association probabilities) and not associating values with any erroneous outcomes. However, if the outcome for an outcome measurement is missed in the sentence, then that outcome measurement is now useless and an ARR calculation is missed. To increase outcome recall at the risk of some loss to precision, when combining results from the separate classifiers, the ensemble approach assigns the outcome label to a token as long as one of the classifiers assigns the outcome label to this token.

Table 7.19 compares the *complementarity* between the classifier models in the ensemble and the alternate label sequences in the alternate label approach. Complementarity is the percentage of instances where one classifier assigns the wrong label to a token, but another classifier in the ensemble assigns the correct label to the token. Low complementarity implies that the classifiers mostly assign the same labels, defeating the purpose of multiple classifiers. Ideally, whenever one classifier is wrong, the others in the ensemble will be correct, leading to high complementarity. Complementarity is computed for each classifier in the ensemble and averaged to obtain the overall complementarity for the ensemble. Since the alternate label approach only has one classifier model, complementarity is computed between the top three label sequences returned by classifier for a sentence. Each alternate label sequence is viewed as output from a different classifier model in an ensemble, even though they are really generated by the same model. Table 7.19 contains complementarity values for both approaches when trained on BMJ, Cardio and BMJCardio corpora individ-

ually and applied to the Ischemia corpus. Since the Cardio corpus (42 abstracts) is much smaller than the BMJ corpus (188 abstracts), for BMJ both approaches are trained on five random 42 abstract subsets of the BMJ corpus. Results for BMJ are averaged over the five sets. Results from the ensemble approach were averaged over five different runs with different random seeds used for randomly selecting the training sets for the individual classifiers in the ensemble.

Table 7.19. Outcome complementarity for the system using alternate CRF labels and the system using an ensemble approach when trained on BMJ, Cardio and BMJCardio corpora.

	BMJ	Cardio	BMJCardio
Alternate labels	**0.38**	**0.37**	**0.41**
Ensemble	0.18	0.18	0.28

Comparing complementarity values between the two approaches we see that the alternate label approach has much higher complementarity than the ensemble approach. With the alternate label approach, the alternate sequences are guaranteed to differ by at least one label in the sequence. The ensemble approach relies on different training sets to provide different model parameters for each classifier in the ensemble. Similar training sets can result in models with similar parameter values leading to high agreement and low complementarity. This result is seen when comparing complementarity values for the ensemble approach when trained on different corpora. The BMJ and Cardio sets are much smaller than the BMJCardio corpus (42 abstracts vs. 230 abstracts). The smaller corpora lead to greater overlap between the training sets for classifiers in the ensemble. As a result, the ensemble models are similar for BMJ and Cardio and complementarity is lower than when trained on BMJCardio. Training on smaller corpora does not affect the complementarity of the

alternate label approach as much as it does with the ensemble approach.

Tables 7.20 and 7.21 compare system performance without any boosting of outcome mention finding, using the proposed alternate label approach and using the described ensemble approach. Results are given for classifiers trained on BMJCardio as well as the previously described BMJ and Cardio subsets. Results for BMJ are averaged over the five random BMJ subsets. Again, results for the ensemble approach are averaged over five different runs with different randomly selected training subsets for each classifier in the ensemble. Post-processing for outcome mentions, as described in Section 5.5, is applied to the results from all three approaches. Table 7.20 gives results for extracting outcome mentions, outcome summary elements and compute ARR values. Table 7.21 shows results for summaries with correct ARR values. Both methods substantially improve recall for outcomes which leads to an increase in recall for ARR values. While some precision is lost when recognizing outcomes, the percentage of summaries with all correct ARR values increases.

Comparing approaches trained with the different corpora we see that the larger BMJCardio corpus results in better performance for outcome mentions, elements and ARR values. Of the two smaller corpora, the Cardio corpus leads to superior ARR results for all three approaches. The average abstract in the Cardio corpus contains more outcome measurements than those in the BMJ corpus. This allows the system to more effectively learn to associate outcome measurements with groups and outcomes.

When trained on the smaller corpora, the alternate label approach achieves slightly higher outcome recall than the ensemble leading a small edge in ARR recall, particularly when trained on Cardio. When trained on the larger BMJCardio corpus, where its complementarity greatly increases, the ensemble approach appears to be more effective overall. This result implies that the alternate label approach is a

better choice for corpora containing abstracts restricted to a small number of topics. It is still capable of maintaining substantial complementarity whereas the ensemble suffers. The ensemble approach is a better choice for larger corpora covering a wider variety of topics. Since the alternate label method consistently maintains higher complementarity for all corpora, a promising direction for future work is to investigate more sophisticated methods for selecting labels from the alternate label sequences.

Table 7.20. A comparison of summary element results achieved with different training sets: BMJCardio, Cardio and random subsets of 42 BMJ abstracts. This table shows recall, precision and F-score for outcome mentions, outcome summary elements and ARR values for the system without any boosting; the system using alternate CRF labels; and the system using an ensemble approach.

	BMJ			Cardio			BMJCardio		
	R	P	F	R	P	F	R	P	F
Outcome mentions:									
No boosting	0.38	**0.53**	0.44	0.42	**0.56**	0.48	0.54	**0.54**	0.54
Alternate labels	**0.46**	0.51	**0.48**	**0.46**	0.54	**0.50**	0.60	0.51	0.55
Ensemble	0.43	0.52	0.47	0.44	0.54	0.48	**0.63**	0.51	**0.56**
Outcome elements:									
No boosting	0.45	**0.55**	0.50	0.46	**0.55**	0.50	0.60	**0.49**	0.54
Alternate labels	**0.53**	0.51	**0.52**	**0.51**	0.53	**0.52**	0.66	0.47	**0.55**
Ensemble	0.51	0.50	0.50	0.50	0.51	0.50	**0.70**	0.44	0.54
ARR:									
No boosting	0.10	**0.62**	0.17	0.18	0.64	0.28	0.28	0.61	0.39
Alternate labels	**0.12**	0.57	**0.20**	**0.22**	**0.65**	**0.33**	0.30	0.60	0.40
Ensemble	**0.12**	**0.62**	**0.20**	0.20	0.63	0.30	**0.31**	**0.62**	**0.41**

Table 7.21. A comparison of summary results achieved with different training sets: BMJCardio, Cardio and random subsets of 42 BMJ abstracts. Recall, precision and F-score for summaries with correct ARR values for the system without any boosting; the system using alternate CRF labels; and the system using an ensemble approach.

	BMJ			Cardio			BMJCardio		
	R	P	F	R	P	F	R	P	F
Any correct:									
No boosting	0.25	0.76	0.38	0.39	0.78	0.52	0.51	0.81	0.62
Alternate labels	**0.28**	0.72	0.40	**0.44**	**0.81**	**0.57**	**0.53**	**0.83**	**0.64**
Ensemble	**0.28**	**0.77**	**0.41**	0.39	0.78	0.52	0.52	0.80	0.63
Correct only:									
No boosting	0.19	**0.58**	0.29	0.28	**0.56**	**0.37**	0.32	0.51	0.40
Alternate labels	0.20	0.52	0.29	**0.29**	0.53	**0.37**	0.32	0.51	0.40
Ensemble	**0.21**	**0.58**	**0.31**	0.28	0.55	**0.37**	**0.35**	**0.54**	**0.42**
Exact:									
No boosting	0.08	**0.23**	0.12	0.15	**0.30**	**0.20**	0.15	0.24	0.18
Alternate labels	**0.09**	0.22	**0.13**	**0.16**	0.29	**0.20**	0.18	0.28	0.21
Ensemble	0.08	**0.23**	0.12	0.15	0.29	**0.20**	**0.19**	**0.29**	**0.23**

7.9 Expert evaluations

Once summaries have been generated, it is important to ask if they are clinically useful. To determine this, a random selection of summaries were evaluated by

EBM experts.

7.9.1 Early evaluations. Expert evaluations were obtained for ARR values computed by the pilot system[45]. The 30 ARR values calculated by the pilot system were independently evaluated by two EBM researchers. Accuracy results determined by the researchers is given in Table 7.22.

The evaluators are asked independently to classify each of these aspects as *correct* (no errors), *qualitatively correct* (contains a minor error, but still useful), or *wrong* (not useful at all). Disagreement arose regarding summary statistics for outcomes that were not the main outcome of interest (e.g. number of people who found their treatment acceptable), and the correctness of detected *per-protocol* results (ignores those who drop out of the trial) when *intention to treat* results (analysis includes those who dropped out) were missed. While there was little agreement on what both considered questionable (only agreement on one that both considered to be wrong), they did agree on 19 (63%) summary stats that they both considered to be correct. This indicates that even the questionable summary stats found by the system may still be useful in some respect.

Table 7.22. Summary statistic accuracy as determined by EBM researchers.

	Correct	Qual. Correct	Wrong
R1	24 (80%)	3 (10%)	3 (10%)
R2	24 (80%)	1 (3%)	5 (17%)

7.9.2 Complete summary evaluations. Once the system was able to produce complete summaries, the EBM experts were asked to evaluate each element in the summary as well as the summaries as a whole. For each element in the summary the

experts were asked to rate it in one of four ways.

- *Correct.* The item is correct to your satisfaction.
- *Qualitatively correct.* The item may contain a small error that would prevent you from labeling it as correct, but you would still consider it useful in understanding the trial. The item does not mislead, confuse, or impair your understanding of the trial or its results. For example, a group name that contains some extra words that are not part or its name, but do not affect its usefulness.
- *Incorrect.* In your opinion the item is wrong, nonsense, and/or misleading.
- *Duplicate.* This item is redundant (but *not* incorrect). It contains essentially the same information that appears in another correct or QC element. In the case of duplicates, the *best* one (in your opinion) should be rated as C or QC, and the others should be rated as duplicates (again, assuming that they are not incorrect).

Experts were also asked to specify the true number of elements that should appear in the summary, *based on the contents of the abstract.* For instance if a trial compares results for three groups, the true number of groups is 3. In some cases, this may be zero. Finally, experts are asked to rate the overall usefulness of the summary.

- *Very helpful.* Summary contained information that helped you grasp the results of the paper.
- *Somewhat helpful.* Summary contained some useful information, but not as much as you would like. However, you would rather have the summary, than not.

- *Not helpful.* Did not mislead, but did not help you in your understanding of the paper.
- *Somewhat misleading.* Not only was the summary unhelpful, it contents actually gave you the wrong idea about some aspect of the trial.
- *Very misleading.* Significantly hinders your understanding of the paper.

We learned from discussions with the EBM experts that summaries with ARR values are more clinically useful than summaries without these values. Since manual evaluation is time-consuming, we only asked the experts to evaluate a set of randomly selected Ischemia summaries that contain *at least one* ARR value (regardless of its correctness).

Table 7.23 summarizes element ratings for both human (R1) and automatic summary element evaluations performed by the system (Automatic). Expert evaluations are currently available for 12 summaries. Recall and precision are computed from the ratings where *correct* ratings are considered true positives; *incorrect* and *duplicate* are considered false positives. Separate recall and precision values are calculated with *qualitatively correct* treated as false positives and as true positives.

Precision is similar between the expert and system for all types of elements. This result implies that the evaluation criteria used by the system for summary elements provides a reasonable estimate of summary element accuracy. Except for ARR values, the automatic evaluation does not have criteria for rating an element as qualitatively correct. To the system a condition, group or outcome element is correct, incorrect or a duplicate match for a correct element.

Table 7.24 contains confusion matrices for conditions, groups, outcomes and ARR values based on the number of element ratings assigned by the expert (R1) and

the system (Automatic). These matrices provide a look at how the automatic evaluation agrees with the human expert evaluation. The expert and system agreed that all 5 age values found by the system were correct. Disagreement concerning acceptable mention boundaries is one cause for different evaluation results between the expert and the automatic evaluation. This effect is apparent with outcome and condition elements where mention boundaries are especially ambiguous. Some outcomes and condition mentions that contain lists or conjunctions of individual outcomes or conditions could be considered one single mention or multiple individual mentions. For instance, the following phrase could be considered one condition or two individual conditions. It was annotated as two condition mentions, but the system extracted the entire phrase as one mention. The expert considered this to be acceptable, but the system did not since the mention matched multiply annotated mentions.

> patients with ST-segment elevation myocardial infarctions less than 12 h and planned primary PCI

For the purpose of communicating trial characteristics to a reader, either option would be effective. Unfortunately, it is not realistic to annotate for all acceptable mention boundaries. This ambiguity affects the estimation of the number of true condition and outcome mentions. Group mentions and outcome mentions that are associated with values do not have this type of ambiguity as they are more distinct entities. Another source of disagreement is missing outcome and condition annotations. While outcomes associated with values were meticulously annotated, outcome mentions that did not have measurements reported in the text were sometimes missed by annotators. These missing annotations caused the majority of cases where the system rated an outcome mention as incorrect but the expert rated it as correct or qualitatively correct. Overall, 24 outcome mentions that were considered incorrect by the system were considered to be correct or qualitatively correct by the expert. Only 4 outcome

mentions that the system identified as correct were considered incorrect by the expert.

Table 7.25 reports the number of summaries that the expert found to be overall helpful or misleading. The expert found half of the summaries to be at least somewhat helpful. Only two summaries were found to be misleading in someway. Accurate ARR values with confidence intervals were the key to helpful or misleading summary.

Table 7.23. The number of correct, qualitatively correct, incorrect, duplicate ratings for each type of summary element. Recall, precision and F-score are calculated from the ratings with qualitatively correct treated as false positives and as true positives.

						QC=FP			QC=TP		
	C	QC	I	Dup.	True No.	R	P	F	R	P	F
Age value:											
R1	5	0	0	0	5	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
Automatic	5	0	0	0	6	0.83	**1.00**	0.91	0.83	**1.00**	0.91
Condition:											
R1	4	2	1	0	14	**0.29**	**0.57**	**0.38**	**0.43**	**0.86**	**0.57**
Automatic	4	0	3	0	21	0.19	**0.57**	0.29	0.19	0.57	0.29
Group:											
R1	23	0	3	1	27	0.85	**0.85**	0.85	0.85	**0.85**	0.85
Automatic	23	0	4	0	25	**0.92**	**0.85**	**0.88**	**0.92**	**0.85**	**0.88**
Outcome:											
R1	37	25	15	13	83	0.45	0.41	0.43	**0.75**	**0.69**	**0.72**
Automatic	40	0	38	12	57	**0.70**	**0.44**	**0.54**	0.70	0.44	0.54
ARR:											
R1	14	6	11	0	43	0.33	0.45	0.38	**0.47**	**0.65**	**0.54**
Automatic	16	1	14	0	41	**0.39**	**0.52**	**0.44**	0.41	0.55	0.47

Table 7.24. Comparision of correct, qualitatively correct, incorrect and duplicate element ratings for the expert (R1) and the automatic evaluations performed by the system.

		Automatic															
		Condition				Group				Outcome				ARR			
		C	QC	I	D	C	QC	I	D	C	QC	I	D	C	QC	I	D
R1	C	2	0	2	0	22	0	1	0	22	0	13	2	12	0	2	0
	QC	1	0	1	0	0	0	0	0	8	0	11	6	3	0	3	0
	I	1	0	0	0	1	0	2	0	4	0	11	0	1	1	9	0
	D	0	0	0	0	0	0	1	0	6	0	3	4	0	0	0	0

Table 7.25. The number of summaries that each expert determined to be very helpful, somewhat helpful, not helpful, somewhat misleading or very misleading.

	V. Helpful	S. Helpful	N. Helpful	S. Misleading	V. Misleading
R1	0	6	4	1	1

7.10 Contributions

This chapter provides evaluations of each main component of the summary system along with expert evaluations of the summaries themselves. The main contributions of this chapter are as follows.

- Performance evaluation for first known system to generate EBM-oriented summaries containing summary statistics. This evaluation includes a ceiling analysis showing and error analysis which provide insight into areas to investigate for

future improvement.

- Ensemble versus alternate CRF labeling selection. A comparison of the effectiveness of two different approaches for boosting outcome mention extraction.
- Expert evaluations of the usefulness of automatically generated EBM-oriented summaries containing summary measures.

CHAPTER 8

SUMMARY AND CONCLUSION

In this document I describe ACRES, a machine learning based system for the novel task of automatically generating EBM-oriented summaries for medical research papers. While there has been some prior work on aspects of this problem, there has been no known attempt to find and correctly interpret all of the information needed for computing summary measures and assembling EBM-oriented summaries.

8.1 Contributions

The following is a summary of the primary contributions of the work described in this document.

- *Unique corpora.* This work provides the first collection of RCT abstracts that have been annotated for use in developing a system to produce EBM-oriented summaries.

- *System that produces EBM-oriented summaries.* This work describes the first system to automatically generate EBM-oriented summaries containing summary statistics. It provides an analysis of each aspect of the system and identifies problematic situations for this application.

- *Use of alternate CRF labelings for improved outcome extraction.* This work proposes the use of alternate CRF labelings for boosting outcome extraction and improving the calculation of summary statistics. It examines the circumstances where this method is preferable to an ensemble approach.

8.2 Summarization in other domains

The focus of this work was the summarization of research papers describing

the results of clinical trials. However, clinical research is only one form of quantitative scientific research. Other types of quantitative research include Physics, Chemistry, Biology, Psychology, Engineering and Computer Science. The architecture and methods of ACRES can be generalized to summarize research in these areas as well.

Quantitative research is concerned with accurately testing and measuring the predictions of a given hypothesis. In the case of clinical research, the hypothesis is that a certain experimental treatment is more effective than a comparison treatment at achieving a specific outcome for patients with specific characteristics. Outcome event rates, absolute risk reduction and confidence intervals provide quantitative measures for evaluating this hypothesis.

The ACRES framework consists of a series of methods to extract the key elements that describe a clinical experiment (a comparison of treatments for a given set of outcomes), its results and produce a summary containing only information describing the experiment and its results. To adapt this approach to another other domain, it will be necessary to identify the key element types that are used to describe the hypothesis and the experimental results in the target domain. A corpus of abstracts from the target domain with annotations for the key elements and their relationships will be needed to train the system for the new domain. The feature sets for the classifiers may need to be augmented with semantic features unique to the target domain (e.g. domain-specific word lists). The method for associating numeric values with mentions using the Hungarian method can be used to associate experimental results with hypothesis elements. A summary with slots relevant to the target domain may be constructed from the extracted elements in a similar manner to ACRES.

To illustrate, consider applying the ACRES framework to the domain of experimental physics. Figure 8.1 contains the abstract from an experimental physics

paper by Thrane and Coughlin [48]. Although this paper does not compare groups of people with different interventions, it does compare different methods (seedless vs. seeded clustering algorithms) for addressing a particular problem (detected gravity-wave events). It also reports results that quantify the effectiveness of the tested methods. Figure 8.2 shows an ideal summary for the example abstract that is similar to the type of summary produced by ACRES. Although the element types are somewhat different, the ACRES framework could be used to produce the summary in Figure 8.2. Instead of conditions, groups and outcomes, the new system would extract text describing the *problem* or *question*; *methods* proposed for addressing the problem or question; and *results* measured in the experiments. Entity resolution would identify mentions that refer to the same entity such as the multiple references to the seedless clustering algorithm. The value association approach used in ACRES could be used to associate experimental results with descriptions of what they measure and which methods produced them. Final summaries could be constructed by filling slots in a summary template from the collection of extracted and associated elements produced by the system.

8.3 Future work

Although the system is able to generate EBM-oriented summaries, there are several avenues to pursue in order to extend this work.

While the extraction of key numbers and mentions is effective compared with current baseline approaches, as the ceiling analysis demonstrates, additional improvements can greatly improve the overall performance of the system. Conditions and outcomes could benefit the most from efforts to improve mention extraction. An analysis of the different types of features used in the token classifiers for numbers and mentions seems to indicate that there is little to be gained from developing new com-

Searching for gravitational-wave transients with a qualitative signal model: Seedless clustering strategies

Eric Thrane and Michael Coughlin

Gravitational-wave bursts are observable as bright clusters of pixels in spectrograms of strain power. Clustering algorithms can be used to identify candidate gravitational-wave events. Clusters are often identified by grouping together seed pixels in which the power exceeds some threshold. If the gravitational-wave signal is long-lived, however, the excess power may be spread out over many pixels, none of which are bright enough to become seeds. Without seeds, the problem of detection through clustering becomes more complicated. In this paper, we investigate seedless clustering algorithms in searches for long-lived narrow-band gravitational-wave bursts. Using four astrophysically motivated test waveforms, we compare a seedless clustering algorithm to two algorithms using seeds. We find that the seedless algorithm can detect gravitational-wave signals (at a fixed false-alarm and false-dismissal rate) at distances between 1.5-2x those achieved with the seed-based clustering algorithms, corresponding to significantly increased detection volumes: 4.2-7.4x. This improvement in sensitivity may extend the reach of second-generation detectors such as Advanced LIGO and Advanced Virgo deeper into astrophysically interesting distances.

Figure 8.1. Example experimental physics abstract.

Searching for gravitational-wave transients with a qualitative signal model: Seedless clustering strategies

Problem:

- Gravitational-wave bursts are observable as bright clusters of pixels in spectrograms of strain power
- problem of detection through clustering

Methods:

- seedless clustering algorithms in searches for long-lived narrow-band gravitational-wave bursts
- two algorithms using seeds

Results:

- detect gravitational-wave signals (at a fixed false-alarm and false-dismissal rate) at distances between
  - the seedless algorithm: 1.5-2x
- significantly increased detection volumes:
  - the seedless algorithm: 4.2-7.4x

Figure 8.2. Desired summary for example physics abstract.

plex grammatical and semantic features. Based on the success of the post-processing rules for group mentions, it may be more promising to develop a hierarchical approach to element extraction. A token classifier, as used in this work, identifies mention candidates. A higher-level logic-based approach such as Markov logic networks [39] could be used to post-process the results from the token classifier.

For 20% of outcome measurements, the group or outcome mention related to the measurement is not explicitly mentioned in the sentence. The association stage needs to be able to identify when a group or outcome is not present in the sentence and must be inferred from the context of the sentence. In these situations, it must be able to identify the most likely candidate from the mentions found in previous sentences.

The development of ACRES has been guided by feedback from EBM experts who have evaluated the summaries produced by the system. We plan to conduct an extensive user study that examines the benefit of ACRES summaries in a clinical setting. In addition, we would like to look how EBM summaries could be used for purposes besides clinical decision making. For instance, EBM summaries, particularly if augmented to include cost effectiveness results, could aid experts who compile systematic review for the purpose of health care economics.

Finally, other types of quantitative research can benefit from summaries that includes experimental results. We would like to apply the ACRES framework to new domains.

8.4 Conclusion

Quantitative science research increases our understanding of the world around us. However, as the body of research literature increases, it becomes more challenging to keep up with the results of this research. This dissertation describes a method for

summarizing clinical research abstracts in order to more efficiently identify the results of the studies. The framework used by this approach can be generalized and applied to texts describing quantitative research results in other domains.

APPENDIX A

ARTICLE ANNOTATION SCHEME

A.1 Overview

This chapter describes the scheme for annotating the following information in medical research papers.

- Treatment groups: The names of the groups of people who are assigned a particular type of treatment. Group names usually include the name of the treatment assigned to the group (e.g. *quinine group* or *artemether group*). Groups will also have to be marked as *control* or *experiment*.

- Outcomes: The names of outcomes that are measured in the paper. Whether the outcome is *good* (something the treatment should improve such as recovering from an injury or disease) or *bad* (something the treatment should prevent or decrease such as developing an injury, disease or dying) also needs to be annotated.

- Times: These are the follow-up times when outcomes are measured for each treatment group.

- Group sizes: The number of people in a treatment group. Annotations for group sizes also include references to the treatment groups that they describe.

- Outcome numbers: The number of good or bad outcomes measured for a particular group at a given follow-up time. Annotations for outcome numbers include references to the treatment group they are recorded for, time when they are measured, and the name of the outcome.

- Lost to follow-up: The number of people who where originally assigned to a treatment group, but were not available at a particular follow-up time when outcomes were measured. Annotations for the number lost to followup include

references to the name of the treatment group they were originally assigned to and the follow-up time when they were lost.

- Demographic information: This includes text describing the age, gender, and medical conditions of the trial participants.

All of this information is needed for training and testing a system to automatically generate clinically useful summaries of medical research papers. The demographic and disease/condition information are needed by physicians to determine the study's relevance to their particular patient. The remaining information is needed to calculate the summary statistics *absolute risk reduction* (ARR), which is the percentage of control patients (those with the standard treatment) who would benefit from from taking the new treatment (the experimental treatment), and the *number needed to treat* (NNT) with the new treatment to prevent one bad outcome that would happen with the control. While these statistics sometimes appear papers, often they do not and physicians must calculate them.

A.2 Annotating Abstracts

For now we are only concerned with annotating abstracts, not full papers. This decreases the amount of annotation effort involved and often papers that contain the numeric information that we want, report this in the abstracts[19].

The abstracts have been obtained from Pubmed[20] and are in their original XML format. While there are many XML elements in these files, only the text in the

[19]In a random sample of 54 BMJ (British Medical Journal) articles, I found that it was possible to calculate summary statistics for 30 (56%) papers. Of these 30 papers, 13 contained all needed information in the abstract, 11 required the full text to be examined, and for 6 it was necessary to examine tables to find all of the necessary information.

[20]http://www.ncbi.nlm.nih.gov/pubmed/

AbstractText elements needs to be annotated.

Annotations are XML tags that placed around the segments of a sentence corresponding to a piece of information that we are interested in. These annotations may be added in any text editor, XML editor, or in more sophisticated software packages such as GATE[21].

If you encounter an abstract that does not contain all of the types of information that we are annotating, it is okay. Simply annotate what is there. Times and the number lost to follow-up are not always explicitly mentioned. Some abstracts may not contain any of the information that we want.

A.2.1 Treatment groups. Groups are noun phrases that denote specific treatment groups in the study, including the control group. They are tagged with the <GROUP> tag, which has the attributes:

- *id* - which is unique to the particular group in the study.
- *role* - which is "control" or "experiment" if it is clear from the paper which treatment group is the control group and which has the experimental treatment. This attribute is omitted if the roles are not clear.

In some cases the name of a particular treatment group may seem rather long or the boundaries of the name may seem unclear. In this case try to identify both the minimal and maximal versions of the group name.

- The maximal treatment group is the full noun phrase denoting the treatment group - consider replacing it with the NP "Treatment X group" and seeing if

[21]http:gate.ac.uk

1. the sentence is still grammatical and meaningful
2. no other bits of the noun phrase could be deleted and maintain this quality (i.e., it is the maximal such NP).

Treatment abbreviations inside parentheses are not tagged separately, but usually included in the full group name.

- The minimal version is the minimal noun phrase that denotes the treatment, uniquely distinguishing it from any other treatment condition in the abstract. Preferably, it should be a base NP (noun preceded by possible determiner and adjectives/adverbs), though this may not always be the case. The words "group", "arm", and the like, are considered part of the short group.

The full group name should be annotated with the <GROUP> tag. Inside the full group name, annotate the shortest possible version of the group name with the <SHORT> tag, which has no attributes. A group can have more than one short version.

If a treatment group mention is small enough (as is often the case) that it does not make sense to distinguish between "long" and "short", (e.g. "phenobarbital" or "didgeridoo playing"). In this case the a short version does not need to be annotated.

The following are some examples:

```
<group id="0" role="control">placebo group</group>

<group id="0" role="control"><short>placebo</short> group
   (<short>control</short>)
</group>
```

```
<group id="1" role="experiment">
       home based <short>medication review</short> by pharmacists
 </group>
```

In most sentences, the group names will be relatively short and therefore it often not necessary to identify the "short" version of the group name.

A.2.2 Outcomes. Outcomes are phrases that denote measured outcomes of the study. The outcome subjects (e.g. "in population X", "in patients with condition Y") are not normally included. However, post-modifying prepositional phrases may be included if they further define the outcome (e.g. "injuries to the knee or ankle", "injuries of the knee"). Adjectives describing the degree of the outcome (e.g. mild, moderate, severe, etc.) *should be* included in the outcome. An outcome is tagged with the <OUTCOME> tag, which has these attributes:

- *id* - which is unique to the particular outcome in the study, as for Group above.
- *type* - which is "good" if the outcome is something that we want to increase or "bad" if it is something that we want to decrease. This attribute only applies to the outcome that is annotated. For instance in the clause "33 children in the treatment group did not develop malaria", the outcome is "develop malaria" and should be considered "bad" even though the number reported is the number of "good" outcomes (i.e. not developing malaria).

As with the group names, longer outcome names may have a short version that can be annotated with the <SHORT> tag.

Examples:

```
<outcome id="0" type="bad">
  <short>kwashiorkor<short> ( defined by the
  <short>presence of oedema</short> )
</outcome>

<outcome id="1" type="bad">
   admitted for <short>worsening heart failure or to die</short>
</outcome>

<outcome id="2" type="good">stopped smoking</outcome>

<outcome id="3" type="bad">occurrence of
   <short>symptomatic venous thromboembolism</short>
</outcome>
```

A.2.3 Outcome thresholds. Some outcomes are situations where a trial participant has some sort of measurable value above or below a specific threshold (e.g. "systolic blood pressure above 140 mm Hg", "epds scores > or = 12"). If an outcome mention contains such a threshold, the text describing the threshold should be annotated with the <THRESHOLD> tag. The annotated text should begin with the token that describes the comparison and end with the threshold value. If the threshold value is followed by units, they should be included as well.

Examples:

```
<outcome id="0" type="bad">systolic blood pressure
```

```
  <threshold>above 140 mm Hg</threshold>
</outcome>

<outcome id="1" type="bad">epds scores
  <threshold>&gt; or = 12</threshold>
</outcome>
```

A.2.4 Follow-up times. Times associated when an outcome is measured (e.g. "six weeks", "5 months", "six week follow up") are tagged with <TIME> tag, which has the attributes:

- *id* - which is unique to the particular time in the study, as for groups and outcomes above.
- *units* - which specifies the units (e.g. "days", "weeks", "months") for the follow-up time if they are not part of the annotated time string.
  Note: This attribute may be omitted if the units are already part of the annotated text.

A.2.5 Group sizes. The number of people in a treatment group is tagged with the <GS> tag which has the attributes:

- *group* - which is the id of the group associated with this value
- *time* - which is the follow-up time for when the group has this particular size. This attribute is only needed if an outcome is measured at multiple times and the size of the group changes, due to people dropping out. It should not be needed most of the time.

A.2.6 Outcome numbers. The number of good or bad outcomes for a given treatment group is tagged with the <ON> tag which has the attributes:

- *group* - which is the id of the group associated with this value.
- *outcome* - which is the id of the outcome associated with this value.
- *time* - which is the follow-up time for when this outcome was measured. As with group sizes, this attribute may not always be necessary.

A.2.7 Lost to follow-up. The number of participants lost to follow-up is not usually explicitly mentioned in an abstract. However, if it is mentioned, the number of participants who were lost to follow-up at a given follow-up time is tagged with the <LOST> tag which has these attributes:

- *group* - which is the id of the group associated with this value.
- *time* - which is the follow-up time for when the participants dropped out.

A.2.8 Demographic information. In most cases abstracts briefly describe the attributes that all of the trial participants have in common (e.g. age, gender, medical condition). Typically all of this information is found in a single sentence in the abstract that describes the participants in the trial.

- Population description: The word or phrase that briefly describes the population of people involved in the study should be tagged with the <POPULATION> tag. For instance, possible population descriptors could be “children”, “adolescents”, “participants”, “hospital patients”, “postmenopausal women”, “adults”, “newborn infants”, or “medical and surgical patients”.

- Age: The phrase that describes the age range for the participants in the study should be tagged with the <AGE> tag. Example age phrases include "aged 6-12 years", "aged 65 years or over", "aged 40 or more", "aged 18 to 40 years", "aged 13-17", "aged > or =8 weeks", "mean age 63 (SD 10.7) years".

- Disease/Condition: The phrase that describes a disease or medical condition that all of the participants have in common should be tagged with the <CONDITION> tag. As with treatment groups and outcomes, <SHORT> tags may be used with longer disease/condition mentions when exact mention boundaries are unclear. It is possible that there may be multiple conditions mentioned in the same sentence, each one should be individually tagged.

  The following are examples of sentences with tagged demographic information.

```
30 <population>people</population> <age>aged &gt; or =50</age>
<condition>with knee pain<condition>.
```

```
222 <population>patients</population>; 165 (74%)
<population>women<population>, <age>mean age 83 years</age>.
```

```
316 <population>patients</population> who <condition>
<short>needed urgent intramuscular sedation</short> because
of agitation, dangerous behaviour, or both</condition>.
```

```
<population>Patients</population> <condition><short>attending
the emergency department with acute chest pain</short> during
the year before and the year after the intervention started
</condition>.
```

<population>Children</population> <age>aged 3-36 months</age> <condition>visiting a family paediatrician for <short>acute diarrhoea</short></condition>.

742 <population>pregnant women</population> <condition>with one previous lower segment caesarean section</condition> and <condition>delivery expected at > or=37 weeks</condition>.

CHAPTER B

EBM SUMMARY STRUCTURE

This chapter describes the XML structure of files containing evidence-based medicine (EBM) oriented summaries of medical research papers.

B.1 Study element

The entire summary of an abstract is contained in the root element `Study`. This element contains the following elements.

- `Created` - Contains `Month`, `Day` and `Year` elements for the date that the summary was generated.
- `Name` - The PubMed ID for the abstract.
- `Title` - The title of the article.
- `AbstractLink` - HTML link to the abstract on PubMed.
- `Subjects` - Contains information about the participants in the trial including age, inclusion/exclusion criteria, and the names and sizes of the treatment groups in study. This information is drawn from both the trial registry and the abstract.
- `Outcomes` - Contains the list of outcomes measured in the trial along with outcome results extracted from the abstract.

B.1.1 Subjects element. This element contains an `Eligibility` element describing the trial population and a `Group` element for each treatment group in the study.

The `Eligibility` element contains the following elements which describe the populations in the study.

- `Age` - Contains list of `AgeValue` elements describing the age range of the population. The text value for `AgeValue` element is a number that describes the minimum, maximum, mean, or median age of the trial participants. `AgeValue` has the following attributes.
  - `Type` - The type of age value. These are `min`, `max`, `mean`, and `median`.
  - `Units` - The units of time for the specified age values (`days`, `weeks`, `months`, `years`).

  An `Age` element only contains at most one `AgeValue` element of each type. For instance, an `Age` element will never contain two minimum `AgeValue` elements.
- `Criteria` - Describes an inclusion/exclusion criteria used to determine if a person was eligible for the trial. It has the attribute `type` with the following values.
  - `inclusion` - The criteria describes a characteristic of participants *included* in the study.
  - `exclusion` - The criteria describes a characteristic of participants *excluded* in the study.
  - `unknown` - The nature of the criteria could not be determined.

  The criteria element contains a `Name` element with the condition text extracted from the abstract.

The `Group` element contains information describing a treatment group in the study. It contains two elements.

- `Name` - Group mention text extracted from the abstract.
- `Size` - The number of participants in the treatment group. If multiple sizes were reported for the group, this is the largest size.

B.1.2 Outcomes element. This element contains `Outcome` elements for each outcome in the summary. Each `Outcome` element contains the following elements.

- `Name` - Outcome mention text extracted from the abstract.
- `Type` - The importance of the outcome in the trial, i.e. whether the outcome was a *primary*, a *secondary* outcome or if its importance is *unknown*.
- `Endpoint` - Contains elements describing the outcome values and summary statistics for a pair of groups.

The `Endpoint` element contains the following elements.

- `Group` - Has outcome measurement information for a treatment group. Has attribute with ID of group. Also has the following elements containing outcome measurement information.
  - `Bad` - The number of bad outcomes.
  - `GroupSize` - The number of participants in the treatment group.
  - `EventRate` - The outcome event rate for this group.
- `SummaryStatistics` - Element with ARR, NNT and confidence interval values computed for a pair of groups. It contains a `Statistic` element with following elements.
  - `AbsoluteRisk` - Contains ARR value. Has attribute `Type` which is either `ARR` for absolute risk reduction or `ARI` for absolute risk increase.

- `NumberNeeded` - Contains NNT value. Has attribute `Type` which is either `NNT` for number needed to treat or `NNH` for number needed to harm.

The actual values for both `AbsoluteRisk` and `NumberNeeded` elements appear in `Value` elements. If it is possible to compute confidence intervals, these elements also contain at `Interval` element with `Lower` and `Upper` attributes defining the bounds of the confidence interval. The `Statistic` element also contains attributes `Better` and `Worse` with IDs of the more and less effective treatment groups for this outcome measurement.

B.1.3 Elements with IDs. A number of the elements have a `Id` attribute. The value of this attribute is an identifier that is unique to the element over all summaries for a particular version of the system. The ID is used within a summary to refer to a particular element from within another element. It is also used when evaluating summary elements. The following elements have an attribute with a unique ID: `AgeValue`, `Criteria`, `Group`, `Size` (group), `Outcome`, `Type` (outcome) and `Endpoint`.

B.2 Sample summary

The following is an example of a summary generated by the summarization system in the XML format described in this chapter.

```
<?xml version="1.0" encoding="utf-8"?>
<Study Version="032">
  <Created>
    <Month>8</Month>
    <Day>2</Day>
    <Year>2013</Year>
  </Created>
  <Name>21129714</Name>
  <Title>comparison of effectiveness of carvedilol versus
    bisoprolol for prevention of postdischarge atrial fibrillation
     after coronary artery bypass grafting in patients with heart
     failure.
```

```
</Title>
<AbstractLink>http://www.ncbi.nlm.nih.gov/pubmed/21129714
</AbstractLink>
<Subjects>
  <Eligibility>
    <Age>
      <AgeValue bounds="10" id="21129714v032av0" type="mean"
        units="years">66</AgeValue>
    </Age>
    <Criteria Id="21129714v032c0"  type="unknown">
      <Name>underwent CABG</Name>
    </Criteria>
    <Criteria Id="21129714v032c1"  type="inclusion">
      <Name>with decreased left ventricular function</Name>
    </Criteria>
  </Eligibility>
  <Group Id="21129714v032g0">
    <Size Id="21129714v032g0size">160</Size>
    <Name>1 receptor antagonist bisoprolol</Name>
  </Group>
  <Group Id="21129714v032g1">
    <Size Id="21129714v032g1size">320</Size>
    <Name>an in-hospital cardiac rehabilitation program
    </Name>
  </Group>
  <Group Id="21129714v032g2">
    <Size Id="21129714v032g2size">160</Size>
    <Name>the carvedilol group</Name>
  </Group>
</Subjects>
<Outcomes>
  <Outcome Id="21129714v032o0">
    <Type Id="21129714v032o0oType">unknown</Type>
    <Name>atrial fibrillation ( AF</Name>
  </Outcome>
  <Outcome Id="21129714v032o1">
    <Type Id="21129714v032o1oType">unknown</Type>
    <Name>mortality and morbidity</Name>
  </Outcome>
  <Outcome Id="21129714v032o2">
    <Type Id="21129714v032o2oType">unknown</Type>
    <Name>new-onset AF</Name>
  </Outcome>
  <Outcome Id="21129714v032o3">
```

```
  <Type Id="21129714v032o3oType">unknown</Type>
  <Name>postdischarge AF</Name>
</Outcome>
<Outcome Id="21129714v032o4">
  <Type Id="21129714v032o4oType">unknown</Type>
  <Name>developed AF</Name>
  <Endpoint id="21129714v032o4ep0">
    <Group Id="21129714v032g2">
      <Bad>37</Bad>
      <GroupSize>160</GroupSize>
      <EventRate>23.1%</EventRate>
    </Group>
    <Group Id="21129714v032g0">
      <Bad>23</Bad>
      <GroupSize>160</GroupSize>
      <EventRate>14.4%</EventRate>
    </Group>
    <SummaryStatistics>
      <Statistic Better="21129714v032g0"
         Worse="21129714v032g2">
        <AbsoluteRisk Type="ARR">
          <Value>8.7%</Value>
          <Interval lower="0.2%" upper="17.2%"/>
        </AbsoluteRisk>
        <NumberNeeded Type="NNT">
          <Value>11.5</Value>
          <Interval lower="5.8, " upper="499.5"/>
        </NumberNeeded>
      </Statistic>
    </SummaryStatistics>
  </Endpoint>
</Outcome>
<Outcome Id="21129714v032o5">
  <Type Id="21129714v032o5oType">unknown</Type>
  <Name>all AF episodes were asymptomatic</Name>
</Outcome>
<Outcome Id="21129714v032o6">
  <Type Id="21129714v032o6oType">unknown</Type>
  <Name>outpatient visit</Name>
</Outcome>
<Outcome Id="21129714v032o7">
  <Type Id="21129714v032o7oType">unknown</Type>
  <Name>heart rate</Name>
</Outcome>
```

```
    <Outcome Id="21129714v032o8">
      <Type Id="21129714v032o8oType">unknown</Type>
      <Name>diastolic blood pressures</Name>
    </Outcome>
    <Outcome Id="21129714v032o9">
      <Type Id="21129714v032o9oType">unknown</Type>
      <Name>the incidence of postdischarge AF</Name>
    </Outcome>
    <Outcome Id="21129714v032o10">
      <Type Id="21129714v032o10oType">unknown</Type>
      <Name>left ventricular function</Name>
    </Outcome>
  </Outcomes>
</Study>
```

BIBLIOGRAPHY

[1] A Aronson. Effective mapping of biomedical text to the umls metathesaurus: the metamap program. *Proceedings of the AMIA Symposium*, Jan 2001.

[2] Amit Bagga and Breck Baldwin. Algorithms for scoring coreference chains. In *In The First International Conference on Language Resources and Evaluation Workshop on Linguistics Coreference*, pages 563–566, 1998.

[3] Florian Boudin, Jian-Yun Nie, JoanC Bartlett, Roland Grad, Pierre Pluye, and Martin Dawes. Combining classifiers for robust pico element detection. *BMC Medical Informatics and Decision Making*, 10:1–6, 2010.

[4] Eric Brill and Jun Wu. Classifier combination for improved lexical disambiguation. In *Proceedings of the 17th international conference on Computational linguistics - Volume 1*, COLING '98, pages 191–195, Stroudsburg, PA, USA, 1998. Association for Computational Linguistics.

[5] YongGang Cao, Feifan Liu, Pippa Simpson, Lamont Antieau, Andrew Bennett, James J. Cimino, John Ely, and Hong Yu. Askhermes: An online question answering system for complex clinical questions. *J. of Biomedical Informatics*, 44(2):277–288, April 2011.

[6] Md. Faisal Mahbub Chowdhury and Alberto Lavelli. Disease mention recognition with specific features. In *Proceedings of the 2010 Workshop on Biomedical Natural Language Processing*, pages 83–90, Uppsala, Sweden, July 2010. Association for Computational Linguistics.

[7] Grace Yuet-Chee Chung. Sentence retrieval for abstracts of randomized controlled trials. *BMC Med Inform Decis Mak*, Jan 2009.

[8] Grace Yuet-Chee Chung. Towards identifying intervention arms in randomized controlled trials: Extracting coordinating constructions. *Journal of Biomedical Informatics*, 42(5):790–800, Oct 2009.

[9] Hal Daumé III. Notes on CG and LM-BFGS optimization of logistic regression. Paper available at `http://pub.hal3.name#daume04cg-bfgs`, implementation available at `http://hal3.name/megam/`, August 2004.

[10] Martin Dawes, Pierre Pluye, Laura Shea, Roland Grad, Arlene Greenberg, and Jian-Yun Nie. The identification of clinically important elements within medical journal abstracts: Patient-population-problem, exposure-intervention, comparison, outcome, duration and results (pecodr). *Information in Primary Care*, 15:9–16, 2007.

[11] Berry de Bruijn, Simona Carini, Svetlana Kiritchenko, Joel Martin, and Ida Sim. Automated information extraction of key trial design elements from clinical trial publications. *AMIA 2008 Symposium Proceedings*, pages 1–5, Jul 2008.

[12] Dina Demner-Fushman and Jimmy Lin. Answering clinical questions with knowledge-based and statistical techniques. *Computational Linguistics*, 33(1), 2007.

[13] Jenny Rose Finkel, Trond Grenager, and Christopher Manning. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL 2005)*, pages 363–370, 2005.

[14] Marcelo Fiszman, Thomas C. Rindflesch, and Halil Kilicoglu. Abstraction summarization for managing the biomedical research literature. In *In proceedings of the HLT/NAACL 2004 workshop on computational lexical semantics*, pages 76–83, 2004.

[15] Marie J. Hansen, Nana Ø Rasmussen, and Grace Yuet-Chee Chung. Extracting number of trial participants from abstracts of randomized controlled trials. *Proceedings of Tromsø Telemedicine and eHealth Conference*, pages 1–5, Jul 2008.

[16] Mary Hickson, Aloysius L. D'Souza, Nirmala Muthu, Thomas R. Rogers, Susan Want, Chakravarthi Rajkumar, and Christopher J. Bulpitt. Use of probiotic Lactobacillus preparation to prevent diarrhoea associated with antibiotics: randomised double blind placebo controlled trial. *BMJ*, 335(7610):80+, July 2007.

[17] J.-D. Kim, T. Ohta, Y. Tateisi, and J. Tsujii. Genia corpus—a semantically annotated corpus for bio-textmining. *Bioinformatics*, 19(suppl 1):i180–i182, 2003.

[18] Jin-Dong Kim, Tomoko Ohta, and Jun'ichi Tsujii. Corpus annotation for mining biomedical events from literature. *BMC Bioinformatics*, 9(1):10, 2008.

[19] S Kiritchenko, B de Bruijn, S Carini, J Martin, and I Sim. Exact: automatic extraction of clinical trial characteristics from journal publications. *BMC Med Inform Decis Mak*, 10(1):56, 2010.

[20] H. W. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1-2):83–97, 1955.

[21] Seth Kulick, Ann Bies, Mark Liberman, Mark Mandel, Ryan McDonald, Martha Palmer, Andrew Schein, Lyle Ungar, Scott Winters, and Pete White. Integrated annotation for biomedical information extraction. In *Proceedings of the HLT-NAACL 2004 Workshop 'Linking Biological Literature, Ontologies and Databases: Tools for Users – BioLink 2004'*, pages 61–68, May 2004.

[22] John Lafferty, Andrew McCallum, and Fernando Pereira. Conditional random fields: probabilistic models for segmenting and labeling sequence data. In *International Conference on Machine Learning*, 2001.

[23] Andreas Laupacis, David L. Sackett, and Robin S. Roberts. An assessment of clinically useful measures of the consequences of treatment. *New England Journal of Medicine*, 318(26):1728–1733, 1988.

[24] Rober Leaman and Graciela Gonzalez. Banner: An executable survey of advances in biomedical named entity recognition. *Pacific Symposium on Biocomputing*, 13:652–663, 2008.

[25] Sein Lin, Jun-Ping Ng, Shreyasee Pradhan, Jatin Shah, Ricardo Pietrobon, and Min-Yen Kan. Extracting formulaic and free text clinical research articles metadata using conditional random fields. In *Proceedings of the NAACL HLT 2010 Second Louhi Workshop on Text and Data Mining of Health Documents*, pages 90–95, Los Angeles, California, USA, June 2010. Association for Computational Linguistics.

[26] D A Lindberg, B L Humphreys, and A T McCray. The unified medical language system. *Methods of Information in Medicine*, 32(4):281–291, 1993.

[27] Andrew McCallum and Wei Li. Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003*, pages 188–191. Association for Computational Linguistics, 2003.

[28] Andrew Kachites McCallum. Mallet: A machine learning for language toolkit. http://mallet.cs.umass.edu, 2002.

[29] Kathleen R. McKeown, Noemie Elhadad, and Vasileios Hatzivassiloglou. Leveraging a common representation for personalized search and summarization in a medical digital library. In *Proceedings of the 3rd ACM/IEEE-CS joint conference on Digital libraries*, JCDL '03, pages 159–170, Washington, DC, USA, 2003. IEEE Computer Society.

[30] J. Munkres. Algorithms for the assignment and transportation problems. *Journal of the Society for Industrial and Applied Mathematics*, 5(1):32–38, 1957.

[31] David Nadeau and Satoshi Sekine. A survey of named entity recognition and classification. *Journal of Linguisticae Investigationes*, 30:1, 2007.

[32] Y Niu and G Hirst. Analysis of semantic classes in medical text for question answering. *Proceedings of the ACL 2004 Workshop on Question Answering in Restricted Domains*, pages 54–61, 2004.

[33] Y Niu, G Hirst, G McArthur, and P Rodriguez-Gianolli. Answering clinical questions with role identification. *Proceedings of the ACL 2003 workshop on Natural language processing in biomedicine-Volume 13*, pages 73–80, 2003.

[34] Jim Nuovo, Joy Melnikow, and Denise Chang. Reporting Number Needed to Treat and Absolute Risk Reduction in Randomized Controlled Trials. *JAMA: The Journal of the American Medical Association*, 287(21):2813–2814, June 2002.

[35] Hyung Paek, Yacov Kogan, Prem Thomas, Seymour Codish, and Michael Krauthammer. Shallow semantic parsing of randomized controlled trial reports. In *AMIA Annual Symp Proc. 2006*, pages 604–608, 2006.

[36] Fuchun Peng and Andrew McCallum. Information extraction from research papers using conditional random fields. *Information Processing and Management*, 42(4):963 – 979, 2006.

[37] D Pinto, A McCallum, X Wei, and WB Croft. Table extraction using conditional random fields. *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 235–242, 2003.

[38] Sampo Pyysalo, Filip Ginter, Juho Heimonen, Jari Bjorne, Jorma Boberg, Jouni Jarvinen, and Tapio Salakoski. Bioinfer: a corpus for information extraction in the biomedical domain. *BMC Bioinformatics*, 8(1):50, 2007.

[39] Matthew Richardson and Pedro Domingos. Markov logic networks. *Machine learning*, 62(1-2):107–136, 2006.

[40] W. Scott Richardson, Mark C. Wilson, Jim Nishikawa, and Robert S. A. Hayward. The well-built clinical question: A key to evidence-based decisions. *ACP Journal Club*, 123:A–12, 1995.

[41] Barbara Rosario and Marti A. Hearst. Classifying semantic relations in bioscience texts. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics (ACL 2004)*, page 430, 2004.

[42] Alan Schwartz. Evidence based medicine (ebm) and decision tools. *MedEdPORTAL*, 2006. Available from: `http://www.aamc.org/mededportal`, ID = 209.

[43] F Sha and F Pereira. Shallow parsing with conditional random fields. *Proceedings of HLT-NAACL*, Dec 2003.

[44] I Sim, B Olasov, and S Carini. The trial bank system: capturing randomized trials for evidence-based medicine. *AMIA Annual Symposium Proceedings*, 2003:1076, 2003.

[45] R.L. Summerscales, S. Argamon, S. Bai, J. Hupert, and A. Schwartz. Automatic summarization of results from clinical trials. In *Bioinformatics and Biomedicine (BIBM), 2011 IEEE International Conference on*, pages 372 –377, nov. 2011.

[46] R.L. Summerscales, S. Argamon, J. Hupert, and A. Schwartz. Identifying treatments, groups, and outcomes in medical abstracts. In *The Sixth Midwest Computational Linguistics Colloquium (MCLC 2009)*, 2009.

[47] Lorraine Tanabe, Natalie Xie, Lynne Thom, Wayne Matten, and W John Wilbur. Genetag: a tagged corpus for gene/protein named entity recognition. *BMC Bioinformatics*, 6(Suppl 1):S3, 2005.

[48] Eric Thrane and Michael Coughlin. Searching for gravitational-wave transients with a qualitative signal model: Seedless clustering strategies. *Phys. Rev. D*, 88:083010, Oct 2013.

[49] Dirk T Ubbink, Gordon H Guyatt, and Hester Vermeulen. Framework of policy recommendations for implementation of evidence-based practice: a systematic scoping review. *BMJ Open*, 3(1), 2013.

[50] T Elizabeth Workman, Marcelo Fiszman, and JohnF Hurdle. Text summarization as a decision support aid. *BMC Medical Informatics and Decision Making*, 12:1–12, 2012.

[51] R Xu, A Morgan, AK Das, and A Garber. Investigation of unsupervised pattern learning techniques for bootstrap construction of a medical treatment lexicon. *Proceedings of the Workshop on BioNLP*, pages 63–70, 2009.

[52] Rong Xu, Yael Garten, Kaustubh S. Supekar, Amar K. Das, Russ B. Altman, and Alan M. Garber. Extracting subject demographic information from abstracts of randomized clinical trial reports. *MEDINFO 2007*, 2007.