How sick are you?
Methods for extracting textual evidence
to expedite clinical trial screening

DISSERTATION

Presented in Partial Fulfillment of the Requirements for
the Degree Doctor of Philosophy in the
Graduate School of The Ohio State University

By

Chaitanya Shivade, B.E., M.S.

Graduate Program in Computer Science & Engineering
The Ohio State University

2016

Dissertation Committee:

Prof. Eric Fosler-Lussier, Advisor
Prof. Albert M. Lai, Co-advisor
Prof. Marie-Catherine de Marneffe
Prof. Alan Ritter
© Copyright by

Chaitanya Shivade

2016
Clinical trials are instrumental in translating outcomes of scientific research into medical practice. Enrollment of patients that meet requirements of a significant sample size, within a desired time span, is limited by the speed and efficiency of screening patients for these trials. The current process of eligibility screening involves repeated reading of clinical notes to evaluate patients against intricate eligibility criteria. Significant time, human effort and financial resources are consumed to accomplish this task. This dissertation analyzes possible reasons that limit the efficiency of the current clinical trial screening workflow and presents automated methods to facilitate and expedite the enrollment process. This includes extracting specific medical concepts from clinical notes, the study and reasoning of vague language used by healthcare professionals, and finally the identification of criteria-relevant text in clinical notes. We also show that active learning and semi-supervised learning techniques help in overcoming the challenge of limited and expensive training data in this domain.
To my family
ACKNOWLEDGMENTS

Completing this degree has been a truly life-changing experience for me. It has been a long journey and many have helped me along the way.

My thanks first go to my wonderful advisors Eric Fosler-Lussier and Albert M. Lai for guiding me through my doctoral studies. To Eric: You have been a fantastic mentor. You have shaped my abilities as a researcher, and taught me to think clearly. I always got good feedback from you: constructive critique on shortcomings and encouragement of positives. Never was I confused about what you thought about anything I discussed with you. I consider myself to be fortunate, to have had the opportunity of working with you.

To Albert: You are the nicest person any student could ask as an advisor. Thank you for giving me this opportunity to contribute in the field of clinical informatics through your grant. Everything fell in place in my life as a PhD student after I started working with you. I have greatly benefited from your domain expertise, superior writing skills, and overall mentorship.

I would also like to thank Marie-Catherine de Marneffe who is nearly my third advisor. You have been great. Research discussions in your office have been enjoyable interactions that I will never forget. You made me think deeper, asked the right questions, and encouraged me to work harder. You taught me how to organize my thoughts and pen them down in a compelling way. Thank you for teaching me so many things.
This dissertation does not represent all of the research I undertook at Ohio State. I would like to express my gratitude to other faculty members. To Jay Ramanathan: It was a pleasure being your student during the first two years of grad school. You are an inspiration and I would consider it a miracle if I can be even half as enthusiastic as you are. To Rajiv Ramnath: It was great working you and your contagious smile. I deeply admire your vision as a researcher and administration skills as a manager. To Courtney Hebert: You taught me the basics of clinical informatics and mentored me through the first big research study I undertook. Thank you for believing in me and my capabilities. To Alan Ritter: Thank you for being on my thesis committee and giving me feedback.

I was very fortunate to have amazing colleagues in Computer Science, Biomedical Informatics, and Linguistics. I would like to thank Rohit Prabhavalkar, Preethi Jyothi, and Yanzhang(Ryan) He, for setting the bar of success in the lab so high. Special thanks to Preethi Raghavan for guiding me during her time at Ohio State and thereafter at IBM Research in the summer of 2015. To SLaTe lab members Yi Ma, Joo-Kyung Kim, Sirui Xu, Deblin Bagchi, Denis Griffis, Adam Stiff, and Andrew Plummer for listening, for paying attention, and letting me bounce off ideas. Playing chess, writing on a white-board, brewing coffee, and a walk to get lunch will never be the same without you all. To folks from the Linguistics department in the Clippers reading group, especially the faculty: Micha Elsner, Michael White, and William Schuler for giving valuable feedback for my research updates. To Marcelo Lopetegui and Barbara Lara: For all the support in my research and for being great friends. To Kelly Regan: For being nice, reviewing my work for clinical sanity, and for chatting when I was bored. To Pranav Malewadkar and Siddhi Bora: For being awesome labmates, believing in my ideas, and for all the fun times at the BMI lab.
My time at Ohio State has been a series of firsts in a roller-coaster ride. This ranges from my first publication, a small research award, and this dissertation, to my first ballroom dance lesson, football game, and even a sky-dive. This also includes cold winters, self-doubt, criticisms, paper rejections, and failed experiments. It is in times like these that two places on campus comforted me the most: the recreation and physical activity center and the library. These are exceptional facilities maintained by the university and I will always treasure my times spent here.

Funding for my research came from multiple sources. I was supported as a TA by the CSE department in my first year. Thereafter, the Center for Enterprise Transformation and Innovation (CETI) led by Jay and Rajiv supported my research for one year. Finally, the majority of my dissertation research was funded by the National Library of Medicine of the National Institutes of Health under award number R01LM011116.

Many thanks are due to all my friends who kept me sane through my time in OSU. These have been friends in Columbus and others who are close despite being far. There are too many to list in entirety and I shall not risk it here. Finally, I am grateful to my family – Aai, Baba, Didi, and Shirish for their unwavering support through all these years. And last, but not the least, to my nephew Manas, and niece Saanvi for the wonderful greeting cards, peek-a-boo on Skype, and putting a smile on my face.
VITA

March 15, 1986 .......................... Born in Pune, Maharashtra, India

July, 2008 ................................. B.E., Computer Engineering
University of Pune, Pune, India

May, 2015 ................................. M.S., Computer Science
The Ohio State University, Columbus,
Ohio, USA

PUBLICATIONS

Journal Articles


Conference Papers and Books


**FIELDS OF STUDY**

Major Field: Computer Science and Engineering

Studies in Artificial Intelligence: Prof. Eric Fosler-Lussier
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Introduction</td>
<td>1</td>
</tr>
<tr>
<td>1.1</td>
<td>Motivation</td>
<td>1</td>
</tr>
<tr>
<td>1.2</td>
<td>Thesis outline</td>
<td>3</td>
</tr>
<tr>
<td>1.3</td>
<td>Contributions</td>
<td>5</td>
</tr>
<tr>
<td>2.</td>
<td>Background</td>
<td>9</td>
</tr>
<tr>
<td>2.1</td>
<td>Electronic phenotyping: patient cohort identification</td>
<td>9</td>
</tr>
<tr>
<td>2.2</td>
<td>Gradability and biomedical texts</td>
<td>12</td>
</tr>
<tr>
<td>2.3</td>
<td>Textual entailment</td>
<td>15</td>
</tr>
<tr>
<td>3.</td>
<td>Gradable terms in biomedical text</td>
<td>20</td>
</tr>
<tr>
<td>3.1</td>
<td>Corpora</td>
<td>21</td>
</tr>
<tr>
<td>3.1.1</td>
<td>Corpus of clinical notes</td>
<td>21</td>
</tr>
<tr>
<td>3.1.2</td>
<td>Corpus of scientific abstracts</td>
<td>22</td>
</tr>
</tbody>
</table>
3.2 Identification of gradable adjectives ........................................ 23
3.3 Usage characterization ......................................................... 25
  3.3.1 Presence of gradable adjectives ........................................ 25
  3.3.2 Usage pattern ............................................................. 26
3.4 Reasoning using numerical quantities ...................................... 27
  3.4.1 Systolic Function ........................................................ 28
  3.4.2 Anemia ................................................................. 30
  3.4.3 Platelet count ........................................................ 31
  3.4.4 Renal Function ......................................................... 32
  3.4.5 Evaluation ............................................................. 34
3.5 Clustering and ranking gradable terms on a scale ....................... 34
  3.5.1 Extraction using structural patterns .................................. 37
  3.5.2 Clustering ............................................................. 37
  3.5.3 Ranking ............................................................... 38
  3.5.4 Evaluation of clustering .............................................. 39
  3.5.5 Evaluation of ranking ................................................ 46
  3.5.6 Extension to adverbs .................................................. 50
3.6 Conclusion ........................................................................... 51

4. Textual entailment search for extracting eligibility evidence .......... 54
  4.1 Challenges ................................................................. 55
    4.1.1 Nature of clinical data .............................................. 55
    4.1.2 Detecting negated concepts ....................................... 56
    4.1.3 Sentence boundary detection ...................................... 57
  4.2 Dataset Creation ............................................................ 59
    4.2.1 Criteria selection ..................................................... 59
    4.2.2 Annotation process .................................................. 61
  4.3 Baseline Systems : A rule-based approach .............................. 63
    4.3.1 Evaluation ............................................................ 66
    4.3.2 Limitations ............................................................ 67
  4.4 Summary ........................................................................ 68

5. Leveraging Unlabeled Data ..................................................... 69
  5.1 Dataset Creation ............................................................ 70
    5.1.1 Hypotheses generation .............................................. 70
    5.1.2 Entailment annotations .............................................. 72
  5.2 Challenges with clinical data .............................................. 73
  5.3 Self-training ................................................................. 74
  5.4 Active learning ............................................................. 75
  5.5 Baselines ....................................................................... 76
5.6 Supervised learning system ................................................. 77
5.7 Experiments ................................................................. 78
  5.7.1 Supervised baselines with RTE-5 data ....................... 78
  5.7.2 Supervised baselines with clinical data .................... 79
  5.7.3 Semi-supervision with clinical data ......................... 80
  5.7.4 Semi-supervision with RTE-5 data ......................... 82
  5.7.5 Active learning with clinical data ......................... 83
  5.7.6 Active learning with RTE-5 data ......................... 85
5.8 Discussion ............................................................... 86
5.9 Conclusion ............................................................... 88

6. Final Thoughts ............................................................ 90
  6.1 Summary of Contributions ........................................ 90
  6.2 Future Directions ..................................................... 94
  6.3 Conclusions .......................................................... 97

Appendices:

A. Extending NegEx with kernel methods for negation detection .... 99
  A.1 Kernel Methods ......................................................... 99
    A.1.1 NegEx Features Kernel .................................... 100
    A.1.2 Augmented Bag of Words Kernel ..................... 100
  A.2 Datasets .............................................................. 100
  A.3 Experiments .......................................................... 102
    A.3.1 Within dataset evaluation ................................ 102
    A.3.2 Cross dataset evaluation ................................ 103
    A.3.3 Semi-supervised approach ............................ 104
  A.4 Dependency Tree Kernels ........................................ 105
  A.5 Discussion .......................................................... 107
  A.6 Conclusion .......................................................... 109

B. Evaluation of Sentence Boundary Detection ......................... 110
  B.1 Experimental Method .............................................. 111
  B.2 Results ............................................................... 112
    B.2.1 Analysis by terminal character ....................... 113
  B.3 Conclusion .......................................................... 114

C. List of modifiers for gradable adjectives .......................... 115
# LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.2 Spearman’s Correlation between clinical notes for semantic type modification by gradable adjectives.</td>
<td>27</td>
</tr>
<tr>
<td>3.3 Evaluation of probabilistic models to predict gradable terms for numerical values in the data.</td>
<td>34</td>
</tr>
<tr>
<td>3.4 Example comparison of automatically derived clusters against gold standard clusters from WordNet.</td>
<td>41</td>
</tr>
<tr>
<td>3.5 AMT-based evaluations of cluster accuracy and pairwise ranking accuracy of systems that vary in the source of clustering data, source of strength counts, and part of speech. For comparison, the approach used by de Melo and Bansal (2013) achieves a pairwise ranking accuracy of 76.1% on the non-polysemous WordNet clusters.</td>
<td>43</td>
</tr>
<tr>
<td>3.6 Examples of automatically derived adjective clusters from PubMed abstracts.</td>
<td>45</td>
</tr>
<tr>
<td>3.7 Examples of automatically derived adverb clusters from PubMed abstracts.</td>
<td>51</td>
</tr>
<tr>
<td>3.8 Example rankings for adjectives and adverbs from PubMed data.</td>
<td>52</td>
</tr>
<tr>
<td>4.1 Criteria for textual entailment dataset.</td>
<td>61</td>
</tr>
<tr>
<td>4.2 Agreement for the dataset.</td>
<td>64</td>
</tr>
<tr>
<td>4.3 Comparative performance (F1) all four methods.</td>
<td>66</td>
</tr>
</tbody>
</table>

xiii
5.1 Example question hypotheses mappings. ............... 71

5.2 Comparison of class distribution for clinical datasets with RTE-5 Search Task. ........................................... 72

5.3 Performance of supervised baselines on the RTE-5 Search Task Test Set. . . 79

5.4 Performance of supervised baselines on clinical data (* indicates statistical significance) compared to Lucene and other baselines. ................. 80

5.5 Results for self-training on test sets. (* indicates statistical significance) . . 83

5.6 Distribution of partial and complete entailments in the corpus. ............... 86

A.1 Number of affirmed and negated concepts in each dataset. ............... 101

A.2 Within dataset performance of kernels based on F1-score using 10-fold cross validation. Bold results indicate significant improvements over the baseline (p<0.05, McNemar’s test). ................................. 103

A.3 Cross dataset performance on the i2b2Test mod dataset given different training datasets. ...................................................... 103

A.4 Semi-supervised models on the i2b2Test mod dataset. ......................... 104

B.1 Precision(P), Recall (R), and F1 score(F) of sentence boundary detection task, evaluated for each tool on each dataset. The best results for each dataset are highlighted in bold. ................................. 112

B.2 Distribution of terminal sentence characters among the four corpora, by character type. ...................................................... 113
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Organization of the thesis.</td>
<td>4</td>
</tr>
<tr>
<td>3.1</td>
<td>Probabilistic modeling of adjectives describing systolic function.</td>
<td>29</td>
</tr>
<tr>
<td>3.2</td>
<td>Probabilistic modeling of descriptions for anemia.</td>
<td>31</td>
</tr>
<tr>
<td>3.3</td>
<td>Probabilistic modeling of descriptions for variations in platelet count.</td>
<td>32</td>
</tr>
<tr>
<td>3.4</td>
<td>Probabilistic modeling of descriptions for variations in creatinine.</td>
<td>33</td>
</tr>
<tr>
<td>3.5</td>
<td>Comparison of CBOW and skip-gram for clustering of adjectives.</td>
<td>40</td>
</tr>
<tr>
<td>3.6</td>
<td>Instructions shown to AMT workers for evaluation of clustering.</td>
<td>42</td>
</tr>
<tr>
<td>3.7</td>
<td>Instructions shown to AMT workers for evaluation of ranking.</td>
<td>47</td>
</tr>
<tr>
<td>4.1</td>
<td>Criteria selection</td>
<td>62</td>
</tr>
<tr>
<td>4.2</td>
<td>Working screen</td>
<td>63</td>
</tr>
<tr>
<td>5.1</td>
<td>Excerpt from a sample clinical note.</td>
<td>74</td>
</tr>
<tr>
<td>5.2</td>
<td>Precision and Recall</td>
<td>81</td>
</tr>
<tr>
<td>5.3</td>
<td>F1-Score.</td>
<td>82</td>
</tr>
<tr>
<td>5.4</td>
<td>Active learning with clinical Data</td>
<td>84</td>
</tr>
<tr>
<td>5.5</td>
<td>Active learning with RTE-5 Data</td>
<td>85</td>
</tr>
</tbody>
</table>
5.6 Comparison of sampling strategies for active learning with clinical data development set. ................................................. 87
CHAPTER 1: INTRODUCTION

An inference is a conclusion reached on the basis of evidence and reasoning. We, as human beings are skilled with inferencing and engage in this mental process effortlessly on a daily basis. In the field of Natural Language Processing (NLP), inference can be defined as the process of concluding the truth of a textual statement based on (the truth of) another given piece of text (Dagan et al., 2013). Humans are also able to perform inferences when the evidence is underspecified, hedged, or vague. Overall, the NLP community has focused on inferences involving the literal meaning on sentences paying little attention to advanced tasks, most of which are limited to newswire text. Since textual inference is dependent on language, there is significant variation in its application across domains. Medicine is a domain with a distinct sublanguage and thus NLP systems are required to adapt to this domain-specific notion of inference to perform better (Friedman et al., 2002). This dissertation explores the use of clinical data from electronic health records (EHR) for textual inference.

1.1 Motivation

Clinical trials play an important role in medical research. These are research studies that investigate new interventions to prevent, detect, or treat a disease. These interventions may be in the form of new drugs, surgical procedures, devices, or novel ways to use existing treatments. The successful completion of a trial is dependent on achieving a significant
sample size of patients enrolled into the trial within a limited time period. However, enrollment involves the crucial step of eligibility determination, which is extremely challenging and time-consuming, often mandating manual review of the EHR (Penberthy et al., 2010; Joseph and Dohan, 2009). Such reviews can involve repeated readings of a patient’s EHR for multiple trials, across every visit to a hospital. This limits the number of patients that can be evaluated. The delay has several undesirable results including: inadequate statistical analyses of outcomes and premature closure of trials, longer trial duration, higher costs of drug production, and loss of accreditation of research centers that perform these studies (Winn, 1994).

The eligibility requirements of a patient, for a clinical trial, are specified in the form of inclusion and exclusion criteria. These are detailed descriptions of the characteristics a patient must or must not have in order to participate in the trial. Patients can be prescreened for eligibility by referring to either structured or unstructured data in the EHR, or a combination of both. Structured data consists of information such as diagnosis codes, laboratory results, medication orders, procedure information, and problem lists. Although these are useful, Köpcke et al. (2013) found that there was a significant gap (65%) between the structured data documented for patient care and the data required for eligibility assessment. Unstructured data in the form of clinical notes frequently contain nuances of clinical presentation and care that are critical for making an eligibility screening determination that the structured data does not. They also remain the preferred means of documentation for physicians (Rosenbloom et al., 2011). Moreover, the criteria are specified in natural language – a format not amenable for computational processing. Hence, not all criteria can be translated into queries (SQL) over structured data. Thus, the use of clinical notes from
the EHR is imperative to assist with determining patient eligibility. Since these lengthy reviews of the EHR are expensive in terms of time and effort, it would be desirable to speed up this step in the eligibility screening workflow.

1.2 Thesis outline

The time spent in resolving a particular eligibility criterion depends on the language used in it. Some criteria are simple and very straightforward to resolve. For instance, the criterion “Age 18 to 85 years old” is very objective and can be resolved using a simple database query. In contrast, consider the criterion “Patients maintained on a high dose of insulin.” Interpreting such statements is a problem since, usage of the term *high* makes it unclear – what was the exact amount of insulin used. Chapter 3 presents our work in understanding the meaning of such inexact terms. However, there are criteria which demand context beyond the understanding of individual words that comprise the criterion statement. For example, resolving the criterion: “Patients with mild symptoms of pneumonia” requires gathering evidence from the text in a clinical narrative to derive a conclusion. We present our work in addressing this problem in Chapter 4. Development of computational models to automate the above task requires labeled data in the form of human annotations. Since obtaining such annotations is expensive, it is desirable to develop methods that consume minimal labeled data and leverage unlabeled data in the process. Chapter 5 discusses our explorations in achieving this goal. Figure 1.1 shows a visual representation of the thesis outline.

To expand on the above outline, gradable terms can be defined as predicative expressions whose domains can be partially ordered according some property that permits grading (Kennedy, 1999). In the above examples, the terms *high* and *mild* express grading of a
dosage amount and the intensity of pneumonia symptoms respectively. The term *high* conveys an imprecise but intuitive associated with dosage amount - a measurable numerical quantity. Gradable terms inherently possess a degree of vagueness and are used in a language to express such imprecision and epistemic uncertainties (Kennedy, 2007), making it difficult to derive a definite meaning from their usage. The property of grading means they can participate in a scale of comparison. Continuing with the above example, the terms (*mild* < *moderate* < *severe*) represent an ordering for a scale that compares intensity of symptoms. Chapter 3 presents a comprehensive characterization of gradable terms usage with a simple model to ground their meaning. We demonstrate the use of computational models to understand their meaning when used as descriptions of a numerical quantity, and to derive an ordering among them, when they participate in a scale of comparison.
Deriving eligibility inference for a criterion such as “Patients with mild symptoms of pneumonia” cannot be based on a measurable numerical quantity. It demands more context and involve putting together multiple pieces of evidence in a clinical note(s). This is a complex task, and automating it is non-trivial. A first step towards this automation is identification of sentences that serve as evidence to derive a conclusion with respect to a given hypothesis. In our case, the conclusion is eligibility inference of a patient for a given criterion. In Chapter 4, we describe the creation of a dataset for evaluation of this task followed by evaluation of simple rule-based methods to address it.

The rule-based methods to identify sentences relevant to a given eligibility criterion are baselines. They lack generalization capabilities and hence the use of machine learning approaches is suitable for such problems. We created a supervised machine learning model for the task that show significant improvement over baselines. However, the amount of labeled data available to us for training such a model is limited. Moreover, creation of an entailment dataset for the clinical domain is time-consuming and expensive. Therefore, Chapter 5 describes self-learning and active learning approaches as extensions to the supervised model for the task.

Finally, Chapter 6 concludes this dissertation with a summary of contributions and a discussion of possible extensions to our work. We demonstrate the use of NLP techniques to assist the manual process of clinical trial eligibility screening and expedite the process.

1.3 Contributions

Gradability in clinical texts: Understanding the meaning of gradable terms is a topic that has recently attracted the attention of the NLP community. These approaches use data
from the newswire domains and depend on manually curated resources such as WordNet (Fellbaum, 1998). Our contributions in this topic are twofold:

1. We present a comprehensive study of gradable terms used in biomedical texts. Using a method proposed by Hatzivassiloglou and Wiebe (2000), we learn to identify the gradable adjectives in our dataset of clinical notes. We show that these adjectives have a substantial presence in our data (30% of the total adjectives). Further, we show that there is a specific pattern in which gradable adjectives are used: some medical concepts are more likely to be modified by these adjectives than others. Finally, we focus on a specific subset of gradable adjectives associated with measurements of numerical quantities and demonstrate the use of a simple Bayesian model to ground their meaning.

2. Gradable terms such as brief, lengthy, and extended illustrate varying degrees of a scale and can therefore participate in comparative constructs. Knowing the set of words that can be compared on the same scale and the associated ordering between them (brief < lengthy < extended) is very useful for a variety of lexical semantic tasks. Current techniques to derive such an ordering rely on WordNet (Fellbaum, 1998) to determine which words belong on the same scale and are limited to adjectives. We describe a fully automated pipeline to extract gradable terms from a corpus, group them into clusters reflecting the same scale and establish an ordering among them. This methodology reduces the amount of required hand-crafted knowledge, and can infer gradability of words independent of their part of speech. Our approach infers an ordering for adjectives with comparable performance to previous work, but also for adverbs.
**Textual entailment:** It is defined as the directional relationship between pairs of textual expressions *text* and *hypothesis*, if humans reading *text* infer that the *hypothesis* is most likely true (Dagan et al., 2006). The Recognizing Textual Entailment (RTE) challenges (Dagan et al., 2013) have been very popular in the past and serve as a platform for evaluating systems developed for textual entailment. We extend research in this problem along multiple dimensions:

1. We create the first datasets for textual entailment in clinical text and describe challenges in the process. In RTE, care was taken to ensure that the hypotheses were explicit, thus limiting ambiguities, concise and easy to interpret in terms of spatial and temporal descriptions. The texts of interest in RTE challenges were newswire documents, which are fairly consistent in English grammar, spelling and punctuation. Our text of interest originates from clinical notes, which are characterized by misspellings, abbreviations, incomplete sentences, inconsistent document structure and other undesirable properties of poor text quality for any NLP task. We show that baseline methods that perform well for the RTE task fail to do so on our dataset, to demonstrate that the task of textual entailment is harder in the clinical domain. Systems developed for RTE challenges used external knowledge sources such as WordNet and Wikipedia extracts to enrich their systems. However these resources have little coverage in the biomedical domain (Bodenreider and Burgun, 2001). We investigate the utility of biomedical domain specific resources such as the Unified Medical Language System (UMLS) (Bodenreider, 2004) and UMLS::Similarity (McInnes et al., 2009) for this task.
2. We develop methods to address the problem of textual entailment search with limited labeled data by (a) a self-training-based method to leverage large amounts of unannotated text, and (b) an active learning approach to minimize the annotation costs involved. We present results for these approaches on a textual entailment dataset developed with clinical text, as well as the RTE-5 dataset, and compare them against multiple supervised baselines in both domains. We find that not only does the use of self-training result in significant improvements over a supervised baseline in the clinical domain, but it also achieves the best result among all systems that had participated in the RTE-5 search task. This work presents the first application of active learning to the problem of textual entailment. We demonstrate that this reduces the number of required annotations by more than 90% in both the clinical and RTE-5 datasets.
CHAPTER 2: BACKGROUND

This chapter discusses relevant literature and concepts that form a background for the ideas discussed in the subsequent chapters of this thesis. We start by a brief review of previous work done by the research community to introduce automation in the task of clinical trial screening. Section 2.2 introduces the concept of gradability and describes past studies aimed at understanding of this aspect of language and developing computational models for the same. Finally, Section 2.3 reviews the topic of textual entailment, and a brief discussion of the main challenges faced in this seemingly simple but challenging task.

2.1 Electronic phenotyping: patient cohort identification

The identification of patients who satisfy predefined criteria from a large population in an institution has numerous use cases, including clinical trial recruitment, outcome prediction, survival analysis, and other kinds of retrospective studies (Mathias et al., 2012; Strom et al., 2011). The ability to extract meaningful pieces of information from the EHR and consolidate them into a coherent structure provides great value for automatically identifying patient cohorts that satisfy complex criteria. Although the term ‘EHR-based phenotyping’ is popular it has not been well defined in literature and its meaning is thus wide ranging. Boland et al. (2013) define the process of identifying patients with or without a given disease as ‘EHR-based phenotyping’ or ‘EHR-driven phenotyping.’ A broader definition of

\[\text{This review has been published as Shivade et al. (2014)}\]
this term can be the identification of patient cohorts meeting certain pre-defined criteria using information found in the EHR (Shivade et al., 2014).

A variety of sources of information in the EHR are used for this task. These cover a wide range such as demographics, medications, lab reports, diagnoses, treatments, clinical notes, and genomic data. Phenotyping systems appear in varying levels of complexity. An important resource available for this task is the Unified Medical Language System (Bodenreider, 2004) popularly known as the UMLS. It is a repository of multiple biomedical vocabularies and standards, developed by the US National Library of Medicine. It is the largest available thesaurus of biomedical concepts, mapping them to semantic types and associating them through hierarchical and non-hierarchical relations. A common step in phenotyping systems that use unstructured clinical notes is the identification of these medical concepts from UMLS. MetaMap (Aronson, 2001) is a popular tool in the research community for this task.

Phenotyping systems have been as simple as checking for presence of a few ICD-9 codes (Schmiedeskamp et al., 2009) or regular expressions (Al-Haddad et al., 2010). There also have been systems that are fairly complex (Coden et al., 2009; Pathak et al., 2012). For instance, IBM’s MedTAS (Coden et al., 2009) automatically instantiates a knowledge representation model from free-text pathology reports. It is based on the open-source UIMA (Unstructured Information Management Architecture) framework (Ferrucci and Lally, 2004) and its components use natural language processing principles, machine learning and rules to discover and populate elements of the model. Phenotyping systems in the past have leveraged a variety of techniques. These include rule-based approaches (Penberthy et al., 2010), use of machine learning algorithms (Van den Bulcke et al., 2011),
statistical techniques (Fine et al., 2010), NLP-based approaches (Bejan et al., 2012) or a combination of one or more of these (Liu et al., 2012).

Although cohort identification is a generic task, it has often being viewed as a problem of identifying patients with a specific diagnosis. EHR-based methods aim at automatic identification of patient cohorts for specific diseases. Moreover, the proportion of studies undertaking this task is also skewed towards diseases such as breast cancer, diabetes and heart failure. Conway et al. (2011) surveyed popular EHR-based phenotyping algorithms for fourteen different diseases and discovered that “while the surface forms of the document differ significantly, the underlying logic used is more homogeneous, with heavy reliance on nested Boolean logic, complex temporality and ubiquitous ICD-9 codes.” Initiatives such as the Electronic Medical Records and Genomics (eMERGE) network or the Cross Institutional Translational Research (CICTR) project and the Strategic Health IT Advanced Research Projects (SHARP) are national level programs aimed at promoting research using EHR across multiple institutions (Rea et al., 2012). However, these programs target generalizability of specific cohort identification algorithms across multiple institutions.

Although automation can be introduced into the task of cohort identification, manual intervention is imperative due to the nature of the medical domain. Often, certain eligibility criteria describing a cohort cannot be automatically processed to identify patient cohorts. Domain experts therefore perform this task manually. Since clinical notes are the preferred means of documentation for physicians (Rosenbloom et al., 2011), a common task performed by physicians is to seek evidence for a given hypothesis about the patient in the numerous clinical notes present in the patient’s EHR. This is a generic task independent of the phenotype under consideration. There has been recent interest in the community to pursue research along these lines. Notably, Ni et al. (2015) assemble popular tools in the
domain to create a system for automated clinical trial screening. They make use of the
cTAKES framework (Savova et al., 2013), in conjunction with Lucene, NegEx (Chapman
et al., 2001) and present an information retrieval based evaluation of the developed system.
The results are promising on a set of 13 trials and the authors also describe methods to
scale the system across larger datasets. There is however, a lack of systems which explic-
itly identify textual evidence for a given criterion within clinical notes in a patient’s EHR.
It is along these lines, that we explore some ideas in the subsequent chapters.

2.2 Gradability and biomedical texts

Expressions used in a language are said to be vague if they do not convey a precise
meaning. Sentences using vague expressions do not give rise to precise truth conditions
(Kennedy, 2007). Consider the following sentence: “The patient was maintained on a high
dose of insulin.” Interpreting such statements is a problem since it is unclear what was
the exact amount of insulin used. Gradability (Sapir, 1944; Lyons, 1977) is a semantic
property that allows a word to describe the intensity of a measure in context, and thus
enables comparative constructs. In the above example, the word high is said to be gradable
since it conveys the meaning associated with the measure—amount.

Gradable adjectives inherently possess a degree of vagueness and are used in a language
to express epistemic uncertainties (Kennedy, 2007). While judgments are strong in extreme
cases, there exist borderline cases, where it is difficult to ascribe an adjective. In the above
example, some amounts of insulin would be considered as a high dose by all, other amounts
would never be considered a high dose, but there is a middle range where it can be difficult
for even experts to judge, if it is a high dose: different experts may have differing thresholds
for what constitutes a high dose.
Broadly, gradable adjectives can be classified into two categories based on their interpretation as measure functions (Bartsch, 1975; Kennedy, 1999). Adjectives such as tall, heavy, and expensive can be viewed as measurements that are clearly associated with a numerical quantity (height, weight, cost). In contrast, adjectives like clever, beautiful, and naive are more complex and underspecified for the exact feature being measured. An important property of gradability is that it identifies different degrees of the quality the word denotes. For example, adjectives such as large, huge and gigantic present different degrees of size or volume. Similarly, adverbs such as approximately, almost and roughly present different degrees of how accurate a measurement is. Thus, one of the characteristics of gradable terms is that they participate in a scale and can be ordered along that scale: for example, good < great < excellent (Kennedy, 2007). Another characteristic is that gradable terms can appear in comparative constructions, e.g., “A is larger than B.” These are important comparative judgments comprised of a psychological process that precedes judgments of counting, e.g., “A is twice as large as B” (Sapir, 1944).

Modern NLP systems face the challenge of interpreting language as close to human perception as possible. Modeling gradable terms as well as their associated meaning and ordering is an important aspect of this challenge. Such information can be very useful for a variety of inference tasks, such as sentiment analysis (Pang and Lee, 2008) and textual inference (Dagan et al., 2006). However, current lexical resources, like WordNet (Fellbaum, 1998), lack annotations capturing the gradability of words. This weakens the notion of similarity: although words such as small and minuscule illustrate varying degrees of size, they are listed as synonyms in WordNet.

Lately, there has been a lot of interest in exploring different approaches to derive an ordering among gradable adjectives based on their semantics (Ruppenhofer et al., 2014;
Sheinman et al., 2013; Schulam and Fellbaum, 2010). de Melo and Bansal (2013) propose a novel Mixed Integer Linear Programming (MILP) based approach, publish a gold standard dataset and report the best performance on ordering scalar adjectives on this dataset. However, these approaches were limited in two ways. First, they depend on a manually created resource, such as WordNet or FrameNet (Baker et al., 1998). Lexical patterns (e.g., ‘not just $x$ but $y$’) are used both to extract words that belong to the same scale and to determine the direction of the ordering (e.g., in the above pattern, $x$ is weaker than $y$). This extraction process gives noisy results that require filtering using an electronic thesaurus. The domain of application was thus restricted to words that exist in an electronic thesaurus. Second, previous work was limited to the study of adjectives. However more recently, Ruppenhofer et al. (2015) evaluated different methods for ranking gradable adverbs by their effect on the intensity of adjectives. Sharma et al. (2015) go beyond derivation of scales and demonstrate an application for predicting movie ratings.

The phenomenon of adjectival modification in biomedical discourse has also been a subject of interest. Through empirical observations, Chute and Elkin (1997) classified frequent modifiers for medical concepts into two types: clinical modifiers (e.g., chronic, severe, acute) and administrative qualifiers (e.g., history of, no evidence of, status post). Bodenreider and Pakhomov (2003) extended this idea and compared adjectival modifications in biomedical literature and patient records. They found that while patient records contain markers for uncertainty (e.g., possible, probable) and non-specific symptoms (e.g., low back pain, discomfort), scientific articles are precise about attributes of organisms or age-groups (e.g., human, canine, neonatal).

Although association between adjectives and numerical quantities has been a topic of research in some studies (Aramaki et al., 2007; Davidov and Rappoport, 2010; Iftene and
very few studies have investigated grounding the meaning of adjectives to numerical quantities. de Marneffe et al. (2010) investigated the problem of interpreting implied answers to yes/no questions when the response is not explicit. Specifically, they investigated question-answer pairs in which the question contains an adjective and the answer contains a numerical measure. For example, predicting the correct yes/no answer in (1) involves interpreting a numerical quantity (age) with respect to the gradable adjective *little*.

1. Q. Are your kids little?
   A. I have a 7 year-old and a 10 year-old.

The authors created logistic regression models for each adjective by querying the web with appropriate keywords (“little kids”) and its antonyms (“not little kids”), so that both positive and negative instances can be learned.

Narisawa et al. (2013) explore a closely related problem of learning *numerical common sense* for the task of RTE in Japanese text. They study a broad set of cases that require semantic inference over numerical expressions. They query the web to gather instances of pairs of numerical quantities and corresponding contexts and propose two approaches. The distribution based approach concludes the numerical quantity to be *large* or *small* if it appears in the top or bottom five percent of the distribution generated for the numerical quantity and *normal* if it is in between. The cue-based approach relies on explicit textual cues (e.g., *as large as*, *only*) for associating a judgment about a numerical expression.

### 2.3 Textual entailment

Textual entailment is the task of automatically determining whether a natural language *hypothesis* can be inferred from a given piece of *text*. Identifying hypotheses and texts that
have the different surface forms, but express the same meaning, is essential to many natural language processing (NLP) tasks including question answering (QA), summarization and machine translation. Textual entailment has been popular in the form of two sub-tasks: recognition and search. Entailment recognition is where, given a text-hypothesis pair, the system needs to classify whether an entailment relationship holds between them (Dagan et al., 2006). It should be noted that textual entailment is not the same as pure logical entailment and has a more relaxed definition. A text entails the hypothesis if a human reading the text would infer that the hypothesis is most likely true. In contrast, the problem of entailment search is to find all sentences in a corpus that entail a given hypothesis (Dagan et al., 2013; Bentivogli et al., 2009).

We study the problem textual entailment search in the clinical domain; specifically in unstructured clinical notes like discharge summaries, history and physical and progress notes. Searching for specific information about a patient, from across hundreds of clinical notes, to help physicians make better, informed clinical decisions, is a challenging task. Entailment search can be applied here to automate the process of examining multiple clinical notes for evidence about a patient-specific statement. This has multiple applications in patient recruitment for clinical trials (Joseph and Dohan, 2009), search, and question answering from clinical notes.

Recognizing Textual Entailment (RTE) shared tasks (Dagan et al., 2013) conducted annually from 2006 up until 2011 have been the primary drivers of textual entailment research in recent years. Because of the initial focus of RTE on entailment recognition, most studies – including recent work of Bowman et al. (2015) – have investigated this type of entailment. It was in RTE-5 (Bentivogli et al., 2009) that the task of entailment search was
first introduced as a pilot. In this type of entailment, the goal of a system is to automatically find, within a set of documents on a topic, all sentences entailing a given hypothesis. Subsequently, RTE-6 (Bentivogli et al., 2010) and RTE-7 (Bentivogli et al., 2011) featured entailment search as the primary task, but constrained the search space to only those candidate sentences that were first retrieved by Lucene, an open source search engine. Based on the 80% recall from Lucene in RTE-5, the organizers of RTE-6 and RTE-7 deemed this filter to be an appropriate compromise between the size of the search space and the cost and complexity of the human annotation task.

Following recent work by Bowman et al. (2015), there has been interest in applying deep neural networks for the problem of recognizing textual entailment (not search). They release the Stanford Natural Language Inference (SNLI) corpus, a large collection (570K) of sentence pairs labeled for entailment, contradiction, and semantic independence. Along with the SNLI corpus, they propose use of long-short term memory units (LSTMs (Hochreiter and Schmidhuber, 1997)) for the problem. Rocktäschel et al. (2015) have shown an improvement over their model using LSTMs with neural attention mechanisms that leverage entailments of pairs of words and phrases. Bowman et al. (2016) propose a faster model architecture, equivalent to TreeLSTMs (Tai et al., 2015) to demonstrate significant gains on the SNLI task.

Marelli et al. (2014b) argue that datasets developed for the RTE challenges have fewer occurrences of important semantic variations such as contextual synonymy, lexical variation phenomena, active/passive, syntactic alternations, and impact of negation, which are necessary to make entailment decisions. In order to fill this gap, they created the SICK (Sentences Involving Compositional Knowledge) dataset for entailment recognition, which

\[\text{http://lucene.apache.org}\]
has also become very popular. The SemEval 2014 task 1 (Marelli et al., 2014a) evaluates systems on this dataset. More recently, Yin et al. (2015) demonstrate the use of an attention-based convolutional neural network for various applications including answer selection, paraphrase identification, and textual entailment on the SICK dataset. Hassanzadeh et al. (2015) use this dataset to train a regression model and show its application for extracting evidence for therapeutic decisions from scientific literature, to practice of evidence based medicine. Their model achieves encouraging results on the biomedical text when compared against human judgment.

Successful RTE systems in the past (Mirkin et al., 2009; Jia et al., 2010; Tsuchida and Ishikawa, 2011) relied on the given training data to either train supervised classifiers, or to develop effective rules for detecting entailing sentences. Operating under the assumption that more labeled data would improve system performance, some researchers sought to augment their training data with automatically obtained labeled pairs (Burger and Ferro, 2005; Hickl et al., 2006; Hickl and Bensley, 2007; Zanzotto and Pennacchiotti, 2010; Celekiyilmaz et al., 2009).

Burger and Ferro (2005) automatically created an entailment recognition corpus using the news headline and the first paragraph of a news article as near-paraphrases. Their approach has an estimated accuracy of 70% on a held out set of 500 pairs. The primary limitation of the approach was that it only generated positive training examples. Hickl et al. (2006) improved upon this work by including negative examples selected using heuristic rules (e.g., sentences connected by although, otherwise, and but). On RTE-2 their method achieves accuracy improvements of 10%. However, Hickl and Bensley (2007) achieved only a 1% accuracy improvement on RTE-3 using the same method, suggesting that it is not always as beneficial.
Zanzotto and Pennacchiotti (2010) created an entailment corpus using Wikipedia data. They hand-annotated original Wikipedia entries, and their associated revisions for entailment recognition. Using a previously published system for RTE (Zanzotto and Moschitti, 2006), they show that their expanded corpus does not result in improvement for RTE-1, RTE-2 or RTE-3.

Celikyilmaz et al. (2009) address the same issue of the lack of extensive labeled data by creating an entailment corpus. They reuse text-hypothesis pairs from RTE challenges in addition to manually-annotated pairs from a newswire corpus. To obtain unlabeled text-hypothesis pairs, the authors used Lucene to query a large newswire corpus with news headlines. Instead of choosing the first sentence of the document as in previous studies, they used the first and twentieth sentence of the search result as representatives of positive and negative entailment respectively. The accuracy of their QA system reported on the TREC04 dataset increased with the addition of unlabeled data.

Research involving computational modeling of properties exhibited by gradable terms is interesting and slowly gaining popularity in the NLP community. We encounter such inference as a challenge in the real-world problem of clinical trial screening. We discuss ideas to overcome this challenge in Chapter 3. This work seeks to understand the meaning of individual gradable terms used in an eligibility criterion. However, complex criteria require gathering evidence from larger units of contexts, such as sentences. In Chapter 4, we present this evidence gathering as a supervised machine learning problem. Chapter 5 presents improvements to the model through techniques for expanding the existing training data set through semi-supervised means, and reducing annotation costs through active learning strategies.
CHAPTER 3: GRADABLE TERMS IN BIOMEDICAL TEXT

Gradable terms can be defined as predicative expressions whose domains can be partially ordered according some property that permits grading (Kennedy, 1999). The semantic property of gradability (Sapir, 1944; Lyons, 1977) enables a word to engage in comparative constructs and accept modifying expressions to act as intensifiers or diminishers. These modifiers are often adverbs such as *slightly, very, somewhat* and occasionally nouns such as *bit high*. In this chapter, we discuss two properties of gradable terms in the biomedical domain: usage of gradable terms as measurements of numerical quantities, and participation of gradable terms that represent a scale of comparison. In the first part, we conduct a comprehensive study of gradable adjectives used in clinical text. Using a method proposed by Hatzivassiloglou and Wiebe (2000), we identify the gradable adjectives in our dataset of clinical notes. We found that these adjectives have a substantial presence (30%) in our data. Further, we show that there is a specific pattern in which gradable adjectives are used: some medical concepts are more likely to be modified by these adjectives than others. In Section 3.4, we demonstrate the use of a simple computational model to ground the meaning of gradable terms associated with measurements of numerical quantities.

In Section 3.5, we study another important property of gradable terms, namely the formation of a scale of meaning and the presence of an ordered relationship among them. We propose a fully automated pipeline that uses structural patterns to extract gradable terms

---

3Work discussed in this chapter has been published as Shivade et al. (2015a)
from a corpus, cluster them into groups that reflect the same semantic scale of comparison, and finally rank them using MILP (Mixed Integer Linear Programming) technique proposed by de Melo and Bansal (2013) to establish an ordering among them. We also explore how the technique performs on domain-specific (biomedical) text, deriving scales for domain-specific terms that might not exist in thesauri. Our approach achieves a comparable performance to previous studies on scalar adjectives, and can be reliably extended to adverbs.

3.1 Corpora

In our investigation of gradability with the biomedical domain, we used two types of corpora for studying each of the two properties discussed above. While association of gradable terms and numerical quantities was studied using a corpus of clinical notes, the property of exhibiting semantic scales was studied using a corpus of abstracts from biomedical articles. We describe both the corpora here, in detail.

3.1.1 Corpus of clinical notes

We used 58,880 clinical notes on Chronic Lymphocytic Leukemia (CLL), 2,652 notes on prostate cancer (PC) and 14,378 notes on Methicillin-resistant Staphylococcus Aureus (MRSA) representing three different cohorts from our institution as a corpus for our study. Thus we had a total of 75,910 notes with an average word count of 1,476 words per note. In addition, we also had access to 8,192 echocardiograms, which are cardiology reports mostly containing semi-structured data with few lines of free text (avg. word count = 64). All clinical notes were from adult patients collected for a period from 2005 to 2010 with necessary approval of the institutional review board at our institution.
These notes are written by healthcare professionals communicating different aspects of patient care and therefore correspond to different note types. For instance, “Progress Notes” are written by physicians documenting periodic developments in the condition of patients, their diagnosis, and treatment. “Operative Notes” are written by surgeons documenting the pre-operative diagnosis, description of the procedure, and the post-operative condition. Our corpus consists of notes belonging to 98 different note types. The name of each note type is mentioned in the first few lines of a templated document header and often has multiple lexical variations. For instance, a “Progress Note” can be an “Inpatient Progress Note” or an “Outpatient Progress Note.” These names were manually normalized to 18 note types, and confirmed by a physician for correctness. Each note from our dataset was thus mapped to one of these normalized types.

Clinical notes have a typical structure: the content is often organized in sections (e.g., “History of Present Illness” followed by “Physical Examination” and ending with “Assessment and Plan”). The beginning of a section is formatted as distinct text with the section name in capital letters followed by a newline character. We used a simple rule-based system to identify section headers and map the contents of a note to these sections. As with note types, section names also had multiple lexical variations (e.g., “Physical Examination” can be “Physical Exam” or “Physical Assessment” or simply “Exam”). Our corpus had 587 section names which were normalized to 17 note sections with a physician’s approval.

3.1.2 Corpus of scientific abstracts

We determined clusters of gradable terms representing the same scale by extracting them from PubMed, a large domain-specific corpus of biomedical texts. It is a free resource developed and maintained by the National Center for Biotechnology Information at the
National Library of Medicine. It provides access to scientific abstracts, full text articles and associated resources. We used 10,875,982 freely available abstracts (not full text articles) from PubMed as our corpus. This corresponds to 88,303,272 sentences in total, where the average length of a sentence is 28 words (including punctuations).

3.2 Identification of gradable adjectives

First, we want to automatically identify gradable adjectives in our corpus of clinical notes and ascertain the extent to which they occur. We re-implemented the method described in (Hatzivassiloglou and Wiebe, 2000), a log linear regression model that learns the weights associated with two features: 1) Number of times an adjective is used in comparative and superlative constructs, and 2) Number of times an adjective is modified by terms that intensify or diminish the semantic meaning of adjectives (mostly adverbs such as very, little, somewhat, etc. and a few nouns such as bit, etc.). (Hatzivassiloglou and Wiebe, 2000) manually created a list of 73 such intensifiers. Their model was generated using the 1987 Wall Street Journal Corpus (Marcus et al., 1993) and tested on a hand curated gold standard dataset of 453 adjectives (235 gradable and 218 non-gradable) created using the Collins Birmingham University International Language Database dictionary, which is annotated for gradable and non-gradable adjectives.

We developed a logistic regression model with the two features described above. For the first feature, a morphology analysis component was developed to identify inflections of adjectives from their base form. This consisted of identifying adjectives in their comparative form using simple parts-of-speech tagging (Toutanova et al., 2003) and regular expression based rules. Although the test set used in Hatzivassiloglou and Wiebe (2000) is available, the list of 73 (de)intensifiers and adverbial modifications was not. We therefore
compiled this list using ten fold cross validation to capture the second feature. In each fold of training, we found adverbs and nouns modifying the gradable adjectives using the Stanford Dependency Parser (version 2.0.4) de Marneffe et al. (2006). For example: For the sentence: “The patient was very serious.” the dependency relation $\text{advmod}(\text{serious, very})$ enabled us to conclude that the adverb very modified the adjective serious. During a single training fold, we found adverbs and nouns modifying gradable adjectives across all sentences and noted the frequencies of each modification. We then determined the best subset by choosing an optimal threshold ($\tau = 81$) for the most frequent modifiers through cross validation (see Appendix C for a complete list). This gave us the second feature for gradability.

Although the method was developed on newswire text, we found that it worked surprisingly well for our clinical corpus. We trained the model on clinical notes and evaluated it on the test set published by Hatzivassiloglou and Wiebe (2000). Of the 453 adjectives in that gold standard test set, we found that 61 adjectives (e.g. wealthy, zesty) were not present in our corpus, resulting in a total of 392 adjectives (217 gradable and 175 non-gradable). Table 3.1 outlines (does not compare) the performance of classification in the two studies. Since the F-score of our model is reasonably high, we use it to identify the gradable adjectives in our corpus. In addition to the 392 adjectives present in the test set, the model identifies 1,709 gradable adjectives in our data. These were domain-specific words such as therapeutic, retroperitoneal, edematous, common adjectives such as acute, febrile, gentle, pale, and also some interesting compositions such as well-nourished, low-normal, and near-complete.
<table>
<thead>
<tr>
<th>Study</th>
<th>Corpus</th>
<th>Gradable</th>
<th>Non gradable</th>
<th>Precision</th>
<th>Recall</th>
<th>F-Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>H &amp; W(2000)</td>
<td>1987 WSJ</td>
<td>235</td>
<td>218</td>
<td>94.1</td>
<td>82.1</td>
<td>87.7</td>
</tr>
<tr>
<td>Our study</td>
<td>Clinical notes</td>
<td>217</td>
<td>175</td>
<td>99.5</td>
<td>84.3</td>
<td>91.3</td>
</tr>
</tbody>
</table>

Table 3.1: Performance of gradable adjective identification on the test set from Hatzivas-siloglou and Wiebe (2000).

3.3 Usage characterization

Vagueness induced by gradable adjectives has been studied by researchers in the past. We want to investigate how frequently such language appears in clinical notes, and if there are certain situations where these terms are more likely to be used. In the following sections, we show that not only do gradable adjectives have a substantial presence in clinical text, but there is also a definite pattern in their usage.

3.3.1 Presence of gradable adjectives

Using the model described in the previous section, we found all gradable adjectives present in our corpus. The percentage of adjectives identified as gradable in the notes across the 18 normalized note types was calculated. This percentage is fairly consistent across different note types, $\mu = 30.85\%, \sigma = 4.9\%$.

In addition to examining the distribution of gradable adjectives across notes types, we performed a finer analysis by calculating their percentage across different sections in a note. The percentage of adjectives identified as gradable across the 17 normalized sections was calculated. Again, it is fairly consistent ($\mu = 31.45\%, \sigma = 6.2\%$) across different sections.
3.3.2 Usage pattern

In this section, we present statistics that characterize the usage of gradable adjectives in describing medical concepts of different semantic types in clinical notes. A major component of the UMLS (Bodenreider, 2004) is the Semantic Network which assigns a semantic type to every concept. A semantic type is a high-level category (e.g., “Sign or Symptom,” “Pharmacological Substance,” “Plant,” “Enzyme”) analogous to named-entity types and there are 133 such semantic types in the 2013AA version of the UMLS.

Using the Stanford Dependency Parser, we identified medical concepts that were modified by a gradable adjective in our corpus and looked up their semantic types. For example: in *extreme fatigue*, the gradable adjective *extreme* modifies the term *fatigue* which has the semantic type “Sign or Symptom,” while in *severe stenosis*, the adjective *severe* modifies the term *stenosis* which has the semantic type “Disease or Syndrome.”

We hypothesized that gradable adjectives modify certain nouns more often than others. In order to test this hypothesis, we calculated how often nouns of a particular semantic type are modified by gradable adjectives. These frequencies were calculated for the three sets of clinical notes corresponding to three different diagnoses (CLL, PC, and MRSA) in our corpus. Nouns from a certain semantic types were very frequently described using gradable adjectives (e.g., “Finding,” “Therapeutic or Preventive Procedure,” “Disease or Syndrome”), and hence had high frequency values in all three datasets. Similarly, nouns from a few semantic types were never described by gradable adjectives (e.g., “Reptiles,” “Professional Society”).

We confirmed this finding by sampling each dataset into five equal folds and repeating the frequency calculations. The observations for frequency variations were consistent for every fold across each dataset. We performed a simple add-one Laplace smoothing
Table 3.2: Spearman’s Correlation between clinical notes for semantic type modification by gradable adjectives.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>CLL</th>
<th>PC</th>
<th>MRSA</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLL</td>
<td>1.00</td>
<td>0.93</td>
<td>0.90</td>
</tr>
<tr>
<td>PC</td>
<td>0.93</td>
<td>1.00</td>
<td>0.91</td>
</tr>
<tr>
<td>MRSA</td>
<td>0.90</td>
<td>0.91</td>
<td>1.00</td>
</tr>
</tbody>
</table>

to account for low frequency semantic types across datasets. Since the size of the three datasets were significantly different, we normalized the frequencies by the sum of frequencies across all semantic types within each dataset. The normalized frequency values represent the probability of a semantic type being modified by gradable adjectives in a dataset. We then computed the Spearman’s correlation for these 133 probabilities across each pair of datasets and found that there was a high correlation between them (Table 3.2). This high correlation across all three diagnoses suggests a definite pattern for the usage of gradable adjectives in clinical text.

### 3.4 Reasoning using numerical quantities

Gradable adjectives are widely studied as implicit or explicit measurements of certain quantities (Bartsch, 1975; Kennedy, 1999). Moreover, they also participate in a scale. For example, the adjectives (*warm* < *hot* < *scorching*) represent a scalar relationship and implicitly measure temperature. While judgments to associate an adjective with extreme values are very strong, those for borderline cases are difficult. In the above example, consider the first two adjectives. Some values of temperature are definitely considered *warm* and others are definitely considered *hot* (and yet not scorching). But there is always a set of values in between which can be either *warm* or *hot*. In order to capture this intuition, we
created a probabilistic model using Bayes rule:

$$P(\text{gradable}|\text{value}) = \frac{P(\text{value}|\text{gradable}) \cdot P(\text{gradable})}{P(\text{value})}$$ (3.1)

Clinicians frequently document their assessments for a patient along with evidence to support their claim, e.g., “Mild anemia, Hgb 8.2.” This sentence has a medical concept “anemia” being described by a gradable adjective mild on the basis of the measurement of a numerical value - hemoglobin. For several medical concepts, we extracted using regular expressions, instances where an assessment for a medical concept was made using a gradable term, along with a numerical evidence to support the claim. In the following subsections, we demonstrate that we can ground the meaning of gradable terms using the above model.

### 3.4.1 Systolic Function

Systolic function is a measure of how well the lower left pumping chamber of the heart sends blood to the rest of the body. It is measured using a numerical quantity called left ventricular ejection fraction (LVEF) which is documented in an echocardiogram. There is variation among physicians defining the precise threshold for a normal ejection fraction (Sanderson, 2007). While normal values range from 55 to 65, values less than 30 imply that the systolic function is severely compromised. We extracted LVEF values from the echocardiogram reports and their corresponding descriptions of systolic function. Posterior probabilities $P(\text{gradable}|LVEF)$ were calculated using equation (1) which resulted in a plot as shown in Figure 3.1.

From the 8,192 echocardiogram reports, we found six gradable adjectives in association with LVEF values. While the adjectives severe, mild and moderate are associated with systolic dysfunction, the adjectives low-normal, normal and hyperdynamic are associated
with systolic function. Although there is discussion in the clinical community regarding qualitative descriptions for ejection fraction (Radford, 2005), there is variation in these recommendations. Moreover, certain terms though used frequently (e.g. low-normal) are never a part of such guidelines.

An interesting observation can be made regarding Figure 3.1, drawing an analogy from the concept of WordNet dumbbells (Sheinman et al., 2012). A WordNet dumbbell is a representation involving an antonym pair (e.g. small and large) as two ends of a semantic scale with semantically similar adjectives arranged in a radial fashion around each adjective. The antonym acting as a centroid and its synonyms as members of a cluster represent words that most likely participate in the same scale. For example, the antonym pair (small, large) results in the dumbbell with clusters (small, tiny, pocket-size, smallish) and (large, gigantic, monstrous, huge) at the two ends. WordNet dumbbells have been used in the past (Sheinman et al., 2013; de Melo and Bansal, 2013) to group gradable adjectives belonging to the same scale. It can be seen that the analogous dumbbell consisting of (severe, mild,
moderate) and (low-normal, normal, hyperdynamic) can be constructed using the modified terms systolic dysfunction and systolic function respectively.

The model captures essential aspects of gradability very well. The scalar relationships (severe < moderate < mild) and (low-normal < normal < hyperdynamic) can be inferred by imposing an order on the mean values for the posterior distributions of these adjectives. Strong judgments for extreme cases and uncertainty for borderline cases can be observed in the form of flat peaks for specific intervals and overlapping distributions for mid-range values.

3.4.2 Anemia

Hemoglobin is a protein in the red blood cells (RBCs) that contains iron and carries oxygen from the lungs to the rest of the body. Anemia is a blood disorder, operationally defined as a reduction in the hemoglobin content of blood caused by a decrease in the RBCs below a reference interval of healthy individuals. The range of normal hemoglobin values for the laboratories at our institution is from 11.7 to 15.5. We found the two adjectives severe and mild to be most commonly used for describing anemia. A number of notes also mentioned anemia with no modifier at all. Figure 3.2 shows the posterior probabilities calculated for the three modifications of anemia: mild, no adjective, and severe using the model outlined in equation 3.1.

It is interesting to note that when physicians refer to anemia without an adjective, it is neither severe nor mild, and has a value in between. As with systolic function, we can infer the ordinal relationship (severe anemia < anemia < mild anemia), considering the mean values for the posterior distributions of these adjectives. Also, strong judgments for extreme values and uncertainty for borderline cases are evident through flat peaks and
overlapping distributions respectively. We also found the adjective *moderate* being used in our data for describing anemia for hemoglobin values between *mild* and *severe*. However, it had few occurrences and hence we did not include *moderate* in our model. Other adjectives describing anemia such as *significant*, *marked*, *slight* and *pernicious* were also found in the data but with low frequency counts.

### 3.4.3 Platelet count

Platelets (also known as thrombocytes) are colorless blood cells that help the process of blood clotting. There are about 150,000 to 450,000 platelet per microliter of blood in the human body (Erkurt et al., 2012). While the condition resulting from a lower than normal platelet count is known as *thrombocytopenia*, the condition resulting from a higher than normal platelet count is referred to as *thrombocytosis*. Since the notion of *low* and *high* counts is gradable, we treat equivalent descriptions of thrombocytopenia and thrombocytosis as gradable. In addition we also extracted instances of clinical notes where the platelet...
count was referred to as *normal*. Using these three descriptions, we applied the Bayes rule explained in Equation 3.1.

Figure 3.3 shows posterior probabilities calculated for these three descriptions of platelet count. As with previous examples, we can infer the ordinal relationship (*thrombocytopenia* $<$ *normal* $<$ *thrombocytosis*) by considering the mean values of their posterior distributions.

### 3.4.4 Renal Function

Creatinine is a chemical made by the body and is used to supply energy to the muscles. Creatinine is removed from the body by the kidneys and released through urine. If kidney function (or renal function) is not normal, creatinine level in the body increases (*Israni and Kasiske, 2011*). Abnormal renal function is referred to through different terminologies such as renal insufficiency, renal failure, and renal dysfunction. The vagueness introduced by the use of these gradable terms is also evident in clinical literature. *Hsu and Chertow (2000)* in their paper titled “Chronic renal confusion: insufficiency, failure, dysfunction, or disease”
propose a set of laboratory values to classify patients as mild, moderate and advanced degrees of chronic renal insufficiency to “facilitate communication among nephrologists and other physicians and provide a framework for comparison of populations.” It should be noted that linguistic ambiguity is not the only reason for this confusion and also has medical explanations which are beyond the scope of discussion of our work.

This problem was acknowledged by the medical community. More than 30 new definitions were proposed (Bellomo et al., 2004) and a new standard is now in place (Khwaja, 2012). However, our data is older (from 2005 to 2010) and has frequent occurrences of these terms. We extracted instances for the gradable terms “normal renal function,” “renal failure,” “renal insufficiency,” “renal failure” and the corresponding creatinine values mentioned by physicians in the text. Further, we computed posterior probabilities for $P(\text{gradable}|\text{creatinine})$ using our model (Figure 3.4). The range of normal creatinine values is between 0.60 to 1.10 for the laboratories at our institution. In comparison with other examples discussed so far, it can be seen from the plot that there is a greater confusion in the use of these terms. This is especially evident in the interval [2,3]. Again, this confirms
<table>
<thead>
<tr>
<th>Medical concept</th>
<th>Number of data points</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Systolic function</td>
<td>10,201</td>
<td>90.4</td>
</tr>
<tr>
<td>Anemia</td>
<td>12,711</td>
<td>88.3</td>
</tr>
<tr>
<td>Platelet count</td>
<td>14,234</td>
<td>94.6</td>
</tr>
<tr>
<td>Renal function</td>
<td>16,309</td>
<td>74.8</td>
</tr>
</tbody>
</table>

Table 3.3: Evaluation of probabilistic models to predict gradable terms for numerical values in the data.

with the property of uncertainty for borderline cases. However, an ordering (normal < dysfunction < insufficiency < failure) can still be inferred.

3.4.5 Evaluation

We evaluated the model to determine if it fits the data well. Using leave one out cross validation, we tested if the model was able to predict the adjective for a given numerical value. The gradable mentioned in each text extract was regarded as the gold standard prediction label. While creating a model, we ensured that there were at least three data points for each measurement value of the numerical quantity present in the data. This allowed us to compute priors for all values in the data. In practice, one would either need large amounts of data or employ smoothing (Kneser and Ney, 1995) to ensure prior calculations for all numerical values are possible. Accuracy is calculated across all gradable terms for each medical concept as described in previous sections (Table 4.3). The models achieve fairly high accuracies which demonstrates that our model fits the data well.

3.5 Clustering and ranking gradable terms on a scale

Previous sections in this chapter described a simple approach to reason the meaning of gradable adjectives when they are associated with numerical quantities. In this section,
we study an important property of gradable terms, namely that of demonstrating a scale of meaning. In contrast to the approach used in the previous section, we leverage textual clues to these goals. This is a two step procedure in which we first cluster the adjectives that belong to the same scale and then derive a ranking among them based on their semantic intensities.

This task has been attempted by two studies in the past. Sheinman et al. (2013) extract adjectives from the Web using lexical patterns indicative of the direction of the scalar relationship between a pair of adjectives. Two sets of patterns are defined: mild patterns in which participating words are such that the first word has a weaker semantic intensity than the second word (e.g., ‘* but not *’ – good but not great); and intense patterns, in which the first word has a stronger semantic intensity than the second word (e.g., ‘not * but still *’ – not freezing but still cold). In the first step, they assign a positive score to an adjective if it is seen as a part of the intense pattern and a negative score if seen as part of the mild pattern. In the second step, they use these scores to partition the adjectives into two subsets one representing mild and the other representing intense adjectives. They perform this partitioning recursively to obtain a complete ordering for a given cluster of adjectives from a WordNet dumbbell.

de Melo and Bansal (2013) improve upon Sheinman et al. (2013) by refining their lexical patterns, and refer to them as “strong-weak” and “weak-strong” patterns. Using frequencies of occurrence for a pair of adjectives across the strong-weak and weak-strong patterns in a corpus, they define an overall weak-strong score. They optimize for this score using mixed integer linear programming (MILP). The constraints of the MILP model two types of strength relationships: the strength relationships between two adjectives in a pair with a possible third adjective, and synonymy relationship between two adjectives based
on information from an external resource. Given a cluster of terms, the MILP produces an ordering of the cluster members using frequency counts of instances where these members are found in strong-weak and weak-strong patterns. To evaluate their approach, de Melo and Bansal construct a manually curated gold standard of 88 clusters, each with a cardinality of three or more adjectives. These 88 clusters are randomly drawn from all possible clusters that are either half of a WordNet dumbbell. Two annotators manually examined these clusters to remove words that did not belong to the same scale. Further, all pairs within these clusters were annotated for scalar relationship: is the adjective in a pair weaker than the other, stronger than the other, or of equivalent intensity. The output of the MILP was tested on these 88 clusters (569 word pairs). They achieve a pairwise accuracy of 78.2%.

We improve upon previous studies along the following lines: first, we introduce structural patterns in the form of parse tree regular expressions (Levy and Andrew, 2006) for extracting gradable terms. Unlike previous studies which rely on web-scale corpora, this allows us to reliably extract terms of interest over full sentences. Moreover, since parse trees contain part-of-speech information, we are able to extend our approach from adjectives to adverbs without any modification. Second, previous approaches relied on manually curated resources such as WordNet to identify terms that belong to the same scale. We eliminate this dependency by automatically clustering gradable terms that belong to the same scale. Lastly, we investigate clustering and ranking of gradable terms in the biomedical domain. The following sections describe these tasks in detail.
3.5.1 Extraction using structural patterns

We use Tregex (Levy and Andrew, 2006), which enables pattern matching on parse trees based on syntactic relationships and regular expression matches on nodes. Using Tregex, we transform de Melo and Bansal’s weak-strong and strong-weak lexical patterns into structural patterns. For example, one way of expanding the lexical pattern ‘* but not *’ into a structural Tregex pattern for adjectives is ‘ADJP<(ADJP<JJ) $(CC<but)$(RB<not)$ (ADJP<JJ)).’ Similarly, a structural pattern for adverbs can be written as ‘ADVP<(ADVP<RB) $(CC<but)$(RB<not)$ (ADVP<RB)).’

Introducing tree patterns requires parsing a corpus: while this additional step in the pipeline might lead to error propagation, the advantages of the structural patterns are that (i) they are more robust than the lexical ones and (ii) restricting results to a desired part-of-speech comes for free. In the experiments reported here, we use the Stanford parser v3.3.1 (Klein and Manning, 2003).

3.5.2 Clustering

In order to determine a ranking of words based on their semantic intensity, the first step is to determine words that belong to the same scale of meaning. We achieve this by automatically clustering words that belong to the same scale. As the clustering algorithm, we use the MATLAB (2014) implementation of K-means++ (Arthur and Vassilvitskii, 2007), a hard clustering algorithm with cosine similarity as a distance metric. Following Hatzi-vassiloglou and McKeown (1993), we use context vectors to represent the words to cluster.

4The choice of a hard-clustering algorithm was mostly for implementation convenience, but carries with it the issue that polysemous words can only appear in one semantic cluster. We leave the issue of deriving a soft clustering approach that works with context vectors, a separate research problem in its own right, to future work.
They make use of standard context vectors for clustering adjectives, where context for every adjective comprises of nouns it modifies across all sentences in a corpus.

However, recent work shows promise for context vectors embedded in a compressed semantic space that are derived using neural networks: Baroni et al. (2014) compare standard context vectors with embedded vectors for a wide range of lexical semantic tasks and found embedded vectors to yield better results. We therefore generate context vectors and compare the utility of both skip-gram and continuous bag of words (CBOW) representations using the \texttt{word2vec} tool (Mikolov et al., 2013) for our task. These two representations have demonstrated varying degrees of success in different NLP tasks (Baroni et al., 2014). Given a window size $w$, the CBOW model predicts the current word given the neighboring words as context. In contrast, the skip-gram model predicts the neighboring words given the current word. We used $w = 5$ and found CBOW to yield better results for our task. Thus the terms extracted from a corpus by the structural patterns are automatically clustered, and these clusters are used as an input to the ranking algorithm.

### 3.5.3 Ranking

Once the terms have been clustered, the second step is to provide a ranking between the cluster members. To do so, we use the MILP implementation provided by de Melo and Bansal (2013). This method computes an overall weak-strong score for a pair of adjectives based on the frequency of that pair in the matches for weak-strong and strong-weak patterns. The MILP then uses these scores among all relevant pairs of adjectives belonging to the same scale, capturing complex interactions to infer an ordering among them.
3.5.4 Evaluation of clustering

In order to evaluate the automatic clustering procedure that uses K-means++ and word vectors, we start with the gold standard provided by de Melo and Bansal (2013): as mentioned above, their data set has 88 gold standard clusters, corresponding to 346 adjectives, annotated by humans for scale ordering. One problem with evaluating a hard clustering algorithm is that the same word may appear in multiple WordNet synsets, corresponding to multiple clusters (soft clustering). We therefore made a “hard cluster version” of the de Melo & Bansal dataset by removing any adjectives that occur in multiple clusters, and then eliminating any singleton clusters. This resulted in a gold standard set of 256 adjectives belonging to 84 clusters.

We clustered the 256 adjectives from the gold standard data subset into 84 clusters: the representation for each adjective was a neural embedding derived using the word2vec tool trained on our PubMed data. We experimented with both the skip-gram and continuous bag of words (CBOW) models to derive vectors of dimension sizes varying from 200 to 800 in increments of 100. To choose the right dimensionality and the best model, we evaluated the quality of the automatically derived 84 clusters against the gold standard. As a metric of evaluation for cluster quality, we follow Hatzivassiloglou and McKeown (1993) and use F1 calculated by comparing equivalence relations generated by the clusters (as implemented in LingPipe (Lingpipe, 2008)). We found that the CBOW model gave clusters closer to the gold standard than the skip-gram model. We found that a dimension size of 600 for the vectors yielded clusters with a maximum F1 score of 57%. Thus, we were able to fix the parameters for our clustering task. Figure 3.5 summarizes the results of this experiment.

In their study, Hatzivassiloglou and McKeown (1993) evaluate the results of their clustering on a small set of 21 adjectives. They presented the 21 adjectives to 9 annotators.
and asked them to partition these adjectives such that each partition contain adjectives that belong to the same scale. They report a best F1 score of 48% but note that such score does not seem to reflect the quality of the clustering. We observe the same problem: our automatically derived clusters have a different organization for words that belong to the same cluster than the gold standard clusters, but in a way that seems intuitive. Some differences between our automatically derived clusters and the gold standard clusters are illustrated in Table 3.4. For example, the adjectives *false* and *misleading* belong to the same cluster in both the gold standard as well as the automatic clustering output. However, the automatic clustering groups the adjectives *false* and *misleading* together with *unreliable* and *wrong*, whereas the gold standard groups *false* and *misleading* with *deceptive* and *fraudulent*. Both clusterings are plausible though. The adjectives *fraudulent* and *deceptive* become part of new clusters in our automatic clustering. It could be argued that the gold standard cluster “deceptive, false, fraudulent, misleading” represents different degrees of
Table 3.4: Example comparison of automatically derived clusters against gold standard clusters from WordNet.

<table>
<thead>
<tr>
<th>Gold standard clusters</th>
<th>Automatic clusters</th>
</tr>
</thead>
<tbody>
<tr>
<td>deceptive, false, fraudulent, misleading</td>
<td>false, misleading, unreliable, wrong</td>
</tr>
<tr>
<td>evil, immoral, sinful, wrong</td>
<td>desperate, humiliated, immoral, insane, sinful</td>
</tr>
<tr>
<td>dangerous, risky, suicidal, unreliable</td>
<td>dangerous, harmful, toxic</td>
</tr>
</tbody>
</table>

“trickery,” whereas the automatic cluster “false, misleading, unreliable, wrong” represent different degrees of “wrongness.” Thus, although both clusters contain different adjectives, they group adjectives that are on the same scale of a different meaning.

Therefore, to evaluate the quality of the automatic clustering, we sampled 50 clusters containing three or more adjectives (corresponding to a total of 190 adjectives) from all the generated clusters and obtained annotations using Amazon Mechanical Turk (AMT), a crowdsourcing platform that has been shown useful for a number of NLP tasks (Snow et al., 2008). Annotators (workers, in AMT parlance) were presented with 15 clusters in each worker session, whose members were each associated with a checkbox. For each cluster, workers had to uncheck the adjectives that did not belong to the same scale. The nature of the annotation task does involve inherent subjectivity which cannot be avoided. We tried to minimize this by giving detailed instructions with accompanying examples to achieve coherent annotations. A screenshot of the instruction screen is shown in Figure 3.6.

To make sure workers were paying attention to the task, 2 clusters among the 15 clusters they saw were clusters for which we a priori knew which adjectives should be removed (e.g., beautiful, pretty, and rainy where rainy had to be unchecked). Most workers did the task well: we only had to discard annotations from 4 worker sessions (out of 140). We
Instructions

- You will be shown a group of words with a checkbox next to each word. We want a group to contain words that belong to the same scale. We ask you to identify for us words that are in currently in a group, but should not be. Please uncheck words that do not belong to the same group. Words in the same group belong to the same scale for measuring a quantity. For example: The words "large," "huge," and "gigantic" are on the same scale and measure the same quantity "size." You have to provide answers for 10 groups presented to you.

- There are three possibilities:

<table>
<thead>
<tr>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>All words belong to the same group, you don't have to uncheck anything.</td>
<td>hot, warm, scorching</td>
</tr>
<tr>
<td>Some words do not belong to the group, you have to uncheck them.</td>
<td>hot, happy, scorching</td>
</tr>
<tr>
<td>None of the words belong to the same group, you have to uncheck all of them.</td>
<td>hot, magical, slippery</td>
</tr>
</tbody>
</table>

- In case you identify that words can be grouped in more than one way, choose to keep the larger group. For example, if the group is (weak, beautiful, delicate, fragile) then prioritize the group (weak, delicate, fragile) over (beautiful, delicate).

If you would like to participate in the experiment, please read the consent form and print it for your records. Then check the box below to give your consent.

**BOX:** ☐ I have read and understood the consent form and agree to participate in this experiment.

**Please either check the box above to consent to participate, or return the HIT.**
ended up with annotations from 8 to 10 workers per cluster. To create a gold standard, we retained in each cluster only those words that were ascertained to be in the same cluster by 6 or more annotators.

For each cluster, we calculated an accuracy score equivalent to the number of correct adjectives (determined to be on the same scale by the annotators) divided by the total number of adjectives in the generated cluster. This accuracy was averaged across all 50 clusters, and yielded a final micro-averaged accuracy of 74.36% as seen in Table 3.5.

We also desired to see how well our approach does on terms that are not specifically in WordNet, but present in a domain-specific corpus such as PubMed. Since there was no gold standard reflecting ideal clustering of data, we explored heuristic measures to choose parameters for our clustering step. We used CBOW vectors over skip-gram vectors since these were more effective in the previous experiment. Since the true value for number of clusters \( k \) was unknown, we chose \( k \) such that the average cardinality of a cluster was three.

Table 3.5: AMT-based evaluations of cluster accuracy and pairwise ranking accuracy of systems that vary in the source of clustering data, source of strength counts, and part of speech. For comparison, the approach used by de Melo and Bansal (2013) achieves a pairwise ranking accuracy of 76.1% on the non-polysemous WordNet clusters.
The value of $k$ was found to be the same ($k = 375$) for all clustering experiments conducted using vector dimension sizes varying from 200 to 800 in increments of 100. To choose the right dimension size $d$ of the CBOW vectors for this fixed value of $k$, we obtained clusters for incremental values of $d$ from 200 to 800 in increments of 100. We determined the number of identical clusters obtained using a particular value of $d$ with its next increment. The lowest value of $d$ which resulted in a maximum number of identical clusters with its next increment was chosen: $d = 400$.

Using vectors of 400 dimensions, we obtained 375 adjective clusters with cardinality varying from 1 to 9. Since these clusters were derived from our biomedical dataset, they comprised of domain-specific adjectives, which are quite unfamiliar even to native English speakers. We manually partitioned the clusters into two sets: (i) containing domain-specific words, and (ii) containing words used in day-to-day English (henceforth referred to as “regular” terms). Examples of clusters from both sets are summarized in Table 3.6. The clusters we obtain look reasonable, grouping together adjectives that pertain to the same scale. The first cluster of domain-specific adjectives qualifies the nouns corresponding to different types of protein with varying degree of specificity, the second cluster contains different qualifications of a tumor, and adjectives in the third cluster qualify different parts of a living cell. For the regular adjective clusters, the clusters look intuitive too, except for the first cluster. The adjectives male and female are not scalar, but match the structural patterns, and are grouped together with adjectives describing age qualifications, due to a strong context overlap in which these words are used.

We randomly sampled 25 clusters from each set, “regular adjectives” and “domain-specific adjectives”, for our evaluation. For regular adjectives, we obtained a clustering
Clusters of domain-specific adjectives
- cytokine, gm-csf, ifn-gamma, il-10, il-12, il-2
- benign, malignant, metastatic, neoplastic, squamous
- mitochondrial, nuclear, ribosomal

Clusters of regular adjectives
- female, male, middle-aged, older, young, younger
- accurate, precise, reliable, reproducible, robust
- additive, insignificant, negligible

Table 3.6: Examples of automatically derived adjective clusters from PubMed abstracts.

accuracy of 86.26% for 25 clusters across 101 regular adjectives using AMT. This is substantially better than the performance of clustering in the previous experiment. We believe that this is due to the fact that the adjectives in the dataset used in the previous experiment originate from WordNet and contain many adjectives (e.g., handsome, crazy, spicy), which are less likely to be found in scientific abstracts. Therefore, the context vectors learned for these words are possibly less accurate compromising the clustering quality.

For the domain-specific adjectives, the annotations require specialized skills. PubMed hosts scientific articles from different disciplines of biological sciences. We obtained annotations from three annotators specializing in disciplines of Biomedical Informatics, Biochemistry, and Nursing. To create the gold standard, a word was retained or discarded from a cluster if two or more annotators agreed on it. We obtained a clustering accuracy of 64.3% for 25 clusters across 101 domain-specific adjectives.
3.5.5 Evaluation of ranking

Since our end goal is to establish an ordering among scalar adjectives, we evaluated the ranking obtained for adjectives within each cluster. We used both datasets: WordNet based clusters and those derived from biomedical abstracts. For evaluating ranking in WordNet based clusters, we use the automatically derived clusters (rather than the WordNet dumbbells) as input to the MILP algorithm. To determine the performance of the ranking produced by the MILP algorithm, we use AMT to obtain pairwise ranking annotations for all unique adjective pairs within a cluster. Workers were presented with 15 word pairs in each worker session. For each pair \((a_1, a_2)\), the worker had to pick one of four options: (1) \(a_1\) is stronger than \(a_2\), (2) \(a_2\) is stronger than \(a_1\), (3) both are equally strong, and (4) \(a_1\) and \(a_2\) are not comparable. Option (4) was present because our clusters possibly contained adjectives that are not on the same scale. A screenshot of the instruction screen is shown in Figure 3.7. As in the previous task for getting annotations for clusters, we inserted two items with a clear ranking (e.g., hot, hotter) for every set of 15 pairs to avoid random annotations. Each set was annotated by 10 workers. All workers passed all the checks and we did not discard any annotations for this task. To create a gold standard we assigned each pair one of four labels, weaker, stronger, equal, or not comparable. A value was assigned based on a majority vote. In case of a tie, the pair was assigned a label of being equal.

In order to compute a ranking, the MILP needs two inputs: 1) the cluster of terms that are on the same scale, and 2) the counts for how many times all pairs of adjectives in that cluster satisfied the weak-strong and strong-weak patterns (henceforth referred to as “strength counts”) as discussed in Section 3.5.1. Thus, for every pair of words (both adjectives or both adverbs) \((a_1, a_2)\) a weak-strong score \(W_1\) is computed where \(a_2\) is stronger than \(a_1\) across all weak-strong patterns and a weak-strong score \(W_2\) is computed when \(s_1\)
**Instructions**

Pairs of evaluative adjectives can often (but not always) be ordered by **strength** or **intensity**. For example, "excellent" is stronger than "good", and "terrible" is stronger than "bad".

In this task, you provide annotations to help linguists determine how pairs of adjectives relate to each other in this way.

**Your task:** For each of the following 15 pairs of adjectives, choose the stronger member of the pair, or determine that both members are of equal strength/intensity.

To make sure that you are paying attention to the task, in some cases, one member of the pair is clearly stronger or both members are clearly of equal strength. Make sure you get them right!

At the end of the experiment, you will answer a brief survey about your language background. We estimate that the entire task should take less than 5 minutes.

If you would like to participate in the experiment, please read the [consent form](#) and print it for your records. **Then check the box below to give your consent.**

**BOX:** [ ] I have read and understood the consent form and agree to participate in this experiment.

**Please either check the box above to consent to participate, or return the HIT.**

**Language survey:** At the end of the task, we ask you to fill out a brief survey about your language background. You will be paid no matter what answers you give in this survey (so please answer truthfully).

---

Figure 3.7: Instructions shown to AMT workers for evaluation of ranking.
is stronger than $a_2$. Similarly, strong-weak scores $S_1$ and $S_2$ are computed. Using these, an overall weak-strong score is calculated as follows:

$$
\text{score}(a_1, a_2) = \frac{(W_1 - S_1) - (W_2 - S_2)}{\text{cnt}(a_1) \cdot \text{cnt}(s_2)}
$$

(3.2)

The goal of the ordering task is to place words $a_1, a_2, \ldots, a_n$ on a linear scale $[0, 1]$. An assignment $x_i \in [0, 1]$ to a word $a_i$ is based on its pairwise scores $\text{score}(a_i, a_j)$. A positive score means $a_j$ is stronger than $a_i$ and hence obtaining $x_j > x_i$ would be desirable. Similarly, a negative score would mean wanting $x_j < x_i$. Therefore, intuitively the goal is to maximize the objective $\sum_{i,j} \text{sgn}(x_j - x_i) \cdot \text{score}(a_i, a_j)$. This optimization is achieved by imposing various constraints, the details of which can be found in de Melo and Bansal (2013).

In the first experiment of ranking adjectives, we ran the full pipeline used by (de Melo and Bansal, 2013) on the 256 adjective (84 hard cluster) subset of their gold standard. Thus, this experiment uses hand-corrected WordNet dumbbells to determine adjectives on the same scale of semantic intensity, followed by the MILP using strength counts from the Google N-gram corpus, to determine the ranking. Their pipeline resulted in a pairwise accuracy of 76.1% which serves as a baseline for comparison. In the second experiment of ranking adjectives, we used the 50 automatically derived adjective clusters as an input for the MILP. Since these adjectives originate from WordNet dumbbells, we refer to them as “WordNet adjective clusters.” We determined the ranking for adjectives within these clusters using strength counts obtained from our PubMed corpus. We obtained an accuracy of 69.23% across 105 pairs. The strength counts for all adjectives in these clusters, from Google N-grams corpus, used in the experiments of (de Melo and Bansal, 2013) were also available to us by the authors. We repeated the previous experiment by substituting strength counts from PubMed corpus with these strength counts from the Google N-grams corpus.
and obtained an accuracy of 84.74% across 119 pairs. It appears from our experiment that pattern counts from a general corpus is a better match for determining the adjective ordering than a more-limited domain corpus, despite the limitation of Google N-grams being restricted to 5-word sequences. We think this is because of two reasons: First, Google N-grams is a very large corpus compared to the one we use. Second, our corpus consists of abstracts and not full text of scientific articles from PubMed. Hence there is less variety in the language used; capturing fewer comparative constructs than Google N-grams. However, it is interesting that we can still extract patterns from domain-specific corpora to act as constraints for the MILP process.

We also evaluated ranking on clusters obtained using PubMed data. We obtained gold standard annotations for ranking using AMT for these 25 “regular adjective” clusters derived from the PubMed corpus using the exact same methodology as for the WordNet based clusters. The strength counts for these adjectives were also derived from the PubMed corpus. We obtained an accuracy of 70.37% across 109 pairs, indicating a similar level of performance to WordNet-based clusters.

Our expert annotators for the domain-specific adjectives faced problems in assessing an ordering between adjectives in a cluster. They report that for majority of the clusters, the ordering of the words would vary given the context. For example, consider the following modifications of the domain specific adjectives of the third cluster in Table 3.6: ribosomal particles, mitochondrial compartments and nuclear compartments, representing different parts of a living cell. If we consider “number of” as the relation in context, we get (ribosomal > mitochondrial > nuclear) as an ordering since the number of ribosomal particles

The MILP does not produce a strength relationship between a pair of adjectives if there are no strength counts for this pair. Hence, we observe a difference in the number of pairs for which accuracy is determined in the two ranking experiments.
is greater than number of mitochondrial compartments, and number of mitochondrial com-
partments is greater than number of nuclear compartments. However, if we consider “size of” as the context, the ordering is reversed.

3.5.6 Extension to adverbs

A novelty of our approach is that we can also apply the technique to other parts of speech (e.g., adverbs). The structural patterns we describe in Section 3.5.1 also enable us to extract candidate scalar adverbs. We follow a similar approach to adjectives: extract adverbs, derive strength counts and rank them using the MILP.

We used CBOW vectors to perform clustering and derived $k = 300$ and $d = 250$. As with the adjective clusters, we found that there were also domain-specific adverbs, illustrated in Table 3.7. Again, the clusters obtained look reasonable. The first cluster of domain-specific adverbs describes relative position of a body part, the second cluster corresponds to adverbs describing identity of a gene that may have an observable effect, the third cluster represents temporal descriptions that relate an event to child birth. The clustering of regular adverbs is accurate, except for the third cluster where inappropriately was found to be an outlier based on our annotations. We followed a similar approach to the adjective experiment, creating two partitions for domain-specific and regular adverbs and sampling 25 clusters from each. Annotations for regular adverbs were obtained from AMT while annotations for domain-specific adverbs were obtained from 3 domain experts. The annotation process for both clusters of adverbs was identical to that of adjectives. We obtained a micro-averaged accuracy of 89.36% for 25 clusters across 104 regular adverbs and a 53.8% for 25 clusters across 89 domain-specific adverbs.
Clusters of domain-specific adverbs
anteriorly, caudally, distally, proximally
chromosomally, clonally, genetically, phenotypically
neonatally, prenatally, postnatally

Clusters of regular adverbs
always, certainly, inevitably, invariably, universally
marginally, modestly, slightly, somewhat
excessively, inappropriately, overly

Table 3.7: Examples of automatically derived adverb clusters from PubMed abstracts.

As in the case of adjectives, our annotators for domain-specific adverbs faced a challenge in ranking adverbs due to lack of context. Therefore we do not report results on ranking of adverbs. We obtained gold standard annotations for ranking using AMT for 25 clusters of regular adverbs derived from the PubMed corpus, using the exact same methodology as adjectives. The strength counts for these adverbs were also derived from the PubMed corpus. We obtained an accuracy of 71.00% across 38 pairs – a performance similar to the adjectives. However, we observe that there are a large number of pairs for which there are no strength counts, and the MILP does not generate a ranking. Table 3.8 shows sample results for ranking adjectives and adverbs from the PubMed data.

3.6 Conclusion

In this chapter, we present a data-driven approach to evaluate two properties exhibited by gradable adjectives in biomedical documents. As a first step, we identified gradable adjectives in clinical text by re-implementing a previously published model for newswire text.
Table 3.8: Example rankings for adjectives and adverbs from PubMed data.

<table>
<thead>
<tr>
<th>Accuracy</th>
<th>POS</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Good</td>
<td>Adjective</td>
<td>serious &lt; life-threatening &lt; fatal</td>
</tr>
<tr>
<td></td>
<td>Adverb</td>
<td>considerably &lt; significantly &lt; dramatically</td>
</tr>
<tr>
<td>Average</td>
<td>Adjective</td>
<td>common &lt; frequent = prevalent</td>
</tr>
<tr>
<td></td>
<td>Adverb</td>
<td>slightly &lt; modestly &lt; marginally</td>
</tr>
<tr>
<td>Bad</td>
<td>Adjective</td>
<td>useful &lt; helpful</td>
</tr>
<tr>
<td></td>
<td>Adverb</td>
<td>continuously = regularly</td>
</tr>
</tbody>
</table>

This model performs surprisingly well with our clinical data. Adjectives identified using this model have a substantial presence in clinical notes across multiple types of documents, written by different healthcare professionals. Analysis of the frequencies of these adjectives and their association with clinical concepts from UMLS revealed that there is a specific pattern for their usage. We showed that a simple Bayesian model can be used effectively to ground the meaning of gradable terms when they are used to describe medical concepts involving measurement of numerical quantities. The second property we study is that of scalar relationship among gradable terms. We present an approach to order gradable term that replaces WordNet-based clusters with automatically derived word clusters, replaces lexical patterns with structural patterns, and show that the approach has utility for not only discovering adjective patterns, but also adverb patterns in biomedical text. Our experimental results show that automatic clustering of gradable words produces promising results. We observe that while automatic ranking based on semantic intensity can be successful established between regular terms, doing so for domain-specific terms requires knowledge of context. The goal of the work presented in this chapter was to develop techniques to derive evidence from individual gradable terms in clinical notes. The following chapter explores...
methods to extract evidence from larger units of context, namely sentences, to assist the task of eligibility criterion resolution in clinical trial screening.
CHAPTER 4: TEXTUAL ENTAILMENT SEARCH FOR EXTRACTING ELIGIBILITY EVIDENCE

Textual entailment is defined as the directional relationship between pairs of textual expressions, text and hypothesis, if humans reading text infer that the hypothesis is most likely true (Dagan et al., 2006). This is a generic NLP task and can be an important step in multiple complex tasks such as question-answering, search, and summarization. In this chapter, we discuss our findings for textual entailment in context of a real-world application, namely clinical trial screening.

Clinical trials are research studies that investigate new interventions such as new drugs, devices, surgical procedures to treat, prevent, or detect a disease. In order to enroll patients in these research studies, they are required to satisfy stringent eligibility criteria to ensure their wellness with the new clinical intervention. These criteria are specified in natural language and are a part of the clinical trial protocol. The current process of eligibility determination is extremely time-consuming requiring a manual review of the patient’s electronic health record (EHR) (Penberthy et al., 2010). This involves reading of clinical notes in the EHR and making an inference concerning whether or not the patient meets the eligibility criterion. Thus, the problem of clinical trial screening is formulated as the task of textual entailment where an eligibility criterion represents the hypothesis and the clinical note corresponds to the text.6

6Work discussed in this chapter has been published as Shivade et al. (2015c)
4.1 Challenges

As discussed in Chapter 2, the Recognizing Textual Entailment (RTE) challenges have been a series of popular shared tasks which serve as the primary venue to evaluate systems developed for textual entailment. However, the RTE tasks have been known to be hard problems in NLP. None of systems in RTE-1 could perform better than the baseline and accuracy was between 0.5 and 0.6 on a balanced dataset of positive and negative entailments, which steadily increased from 0.75 for RTE-2 to 0.8 for RTE-3. RTE-4 and RTE-5 which had a 3-way task to determine entailment between text-hypothesis pairs as “Yes,” “No” and “Unknown,” noticed a fall in accuracy, with the best accuracy being 0.68 for both challenges. The RTE-5 pilot, RTE-6 and RTE-7 challenges differed by centering on a search-based task. The best F1 scores for these tasks were 0.455, 0.48 and 0.48 respectively.

4.1.1 Nature of clinical data

There are a few important differences between the RTE challenges and our task. First, the hypotheses in the RTE challenges were manually curated based on the text against which entailment was to be ascertained. Care was taken to ensure that the hypotheses were explicit (thus limiting ambiguities), as well as concise, and easy to interpret in terms of spatial and temporal descriptions. In textual entailment for clinical trials, the criteria and the notes reflect data in the real world. The criteria therefore do not obey the above desirable properties. Second, the texts of interest in RTE challenges were newswire documents, which are fairly consistent and well formed in terms of English grammar, spelling and punctuation (Meystre et al., 2008). Our text of interest originates from clinical notes, which are characterized by misspellings, abbreviations, incomplete sentences, inconsistent
document structure and other properties undesirable for any NLP task. Third, there is a significant word overlap between the text and the hypothesis in these challenges which increased steadily from 69.25% in RTE-1 to 77.14% in RTE-5 for the positive entailment relation. Therefore, it was observed that systems taking word overlap into account had better chances of performing well. Our data collection shows a very low word-overlap between the criteria and the relevant sentences ($\mu=17\%$) making the task much more challenging. Lastly, the RTE challenges aimed at open-domain textual inference, while our task is very domain-specific. Systems in the RTE challenges used external knowledge sources such as WordNet (Fellbaum, 1998), Wikipedia extracts, FrameNet (Baker et al., 1998), etc. to enrich their systems. Ablation tests of participating systems indicated that they had a significant impact on the system performance. However, such resources are domain-independent and hence have very little coverage in the biomedical domain (Bodenreider and Burgun, 2001).

4.1.2 Detecting negated concepts

A simple approach for identifying information relevant to an eligibility criterion in the patient's EHR is to use a standard search engine which uses text in the eligibility criterion as a query and returns relevant sentences from the clinical notes as the result. Most information retrieval approaches determine a document to be relevant to a concept based on the presence of that concept in the document. However, these approaches fall short if these concepts are negated, leading to a number of false positives. This is an important problem especially in the clinical domain. For example, the sentence: “The scan showed no signs of malignancy” has the concept ‘malignancy’ which was looked for in the patient, but was not observed to be present. The task of negation detection is to identify whether
a given concept is negated or affirmed in a sentence. NegEx (Chapman et al., 2001) is a rule-based system developed to detect negated concepts in the clinical domain and has been extensively used in the literature.

Negation has been a popular research topic in the medical domain in recent years. NegEx along with its extensions (South et al., 2006; Chapman et al., 2013) is one of the oldest and most widely used negation detection systems because of its simplicity and speed. An updated version - ConText (Harkema et al., 2009) was also released to incorporate features such as temporality and experiencer identification, in addition to negation. These algorithms are designed using simple rules that fire based on the presence of particular cues, before and after the concept. However, as with all rule-based systems, they lack generalization. Shortage of training data discouraged the use of machine learning techniques in clinical natural language processing (NLP) in the past. Yet, shared tasks (Uzuner et al., 2011) and other recent initiatives (Albright et al., 2013) are making more clinical data available. This data should be leveraged to harness the benefits offered by machine learning solutions. Recently, Wu et al. (2014) argued that negation detection is not of practical value without in-domain training and/or development, and described an SVM-based approach using hand-crafted features. We developed a kernel-based approach that maps the rule-based NegEx system into a machine learning system and extends its performance. Please refer Appendix A for more details.

4.1.3 Sentence boundary detection

In order to identify the sentences that entail the criterion, a fundamental task is that of segmenting a note into constituent sentences, formally referred to as sentence boundary detection (SBD). SBD is a non-trivial problem, since the end-of-sentence punctuation marks
are ambiguous. For example, a period can refer to an abbreviation, the end of a sentence, or a numeric decimal point. Exclamation points, question marks, quotation marks, ellipses, parentheses or a combination of these make the problem even more challenging.

Although SBD is an important task, it has not recently received much attention from the NLP community, for a variety of reasons (Gillick, 2009; Read et al., 2012). Previous work in SBD is mostly restricted to the news domain and limited datasets such as the Wall Street Journal (WSJ) corpus (Marcus et al., 1993). These studies discuss problems with SBD assuming properties of good quality text, such as proper formatting, grammatical sentence construction, and the absence of incorrect spellings. Clinical text, however, does not possess these desirable properties (Meystre et al., 2008). The challenges associated with SBD are thus elevated in the clinical domain, since basic assumptions of sentence construction and punctuation do not hold true; for example, only 50% of the sentences in our clinical notes corpus end in periods, compared to 99% in the GENIA corpus. It is likely that models developed for SBD on open domain data may not generalize well to clinical texts. Popular shared tasks in the clinical domain, such as the i2b2 challenges (Uzuner et al., 2011, 2012; Sun et al., 2013), provide participants with gold-standard sentence boundaries to evaluate isolated NLP tasks. Previous studies in the open domain have shown that the impact of error propagation originating from SBD into other NLP tasks is substantial (Walker et al., 2001; Kiss and Strunk, 2006). Despite this, SBD has been a neglected task in the biomedical NLP community, with very few studies reviewing or addressing the problem. Therefore, the problem of SBD in the clinical domain deserves a comprehensive assessment with existing NLP tools and directions for development of appropriate solutions. We conducted a qualitative and quantitative evaluation of multiple popular NLP tools used by the biomedical community for the task of SBD. We found that models trained using newswire text
fail to perform on clinical data. It is therefore essential to use a model that is trained using clinical text. Details of this study can be found in Appendix B.

4.2 Dataset Creation

In order to evaluate the performance of an automated system that can identify text relevant to eligibility criteria in clinical notes, we created a gold standard dataset, as outlined below. The 2014 i2b2 challenge (Stubbs et al., 2015) had two tracks with a training set of 790 notes with annotations for protected health information and risk of heart disease, respectively. The third track evaluated the usability of systems developed for the i2b2 challenges. The fourth track allowed participants to demonstrate novel use of the data made available through the challenge. This work was a submission to this fourth track, proposing the use of NLP for expediting clinical trial eligibility screening. Since a limited amount of time was available to attempt the challenge, 80 notes were annotated for four criteria (20 notes per criterion) by our team of annotators. Every annotator marked all full sentences in a note that were relevant to an eligibility criterion.

4.2.1 Criteria selection

The National Library of Medicine and the National Institutes of Health maintain the website www.clinicaltrials.gov, a registry and a database of publicly and privately supported clinical studies across the globe. The 2014 i2b2 challenge provided notes for coronary artery disease (CAD) patients from three cohorts: 1) Patients with no CAD, 2) Patients who developed CAD over the course of provided notes for that patient, and 3) Patients who had CAD from the beginning of their record. Therefore, we downloaded study record data from clinicaltrials.gov for all trials associated with the search term “coronary
artery disease” as XML files (5054 trials in the May 2014 download). We extracted eligibility criteria for these trials and segmented them into individual sentences using a combination of the LingPipe (Lingpipe, 2008) sentence chunking module trained on MEDLINE biomedical abstracts and user defined rules to identify sentence boundaries. Each sentence was assumed to be a single criterion. The annotation team considered the following factors while selecting the criteria: 1) The resolution of the criterion would require a healthcare professional to refer to the notes of the patient, 2) It would be hard to assess patient eligibility for the criterion by referring only to the structured data associated with the patient’s EHR, and 3) The criterion is commonly used across clinical trials for CAD. The criteria were selected to reflect realistic situations where clinical trial coordinators spend substantial time reading notes to assess eligibility, and would benefit from an NLP system such as the one proposed in this study.

The segmented criteria sentences described above were analyzed using MetaMap (Aronson, 2001) to identify concepts from the Unified Medical Language System (UMLS). Ignoring concepts such as “patient,” “study,” “trials,” and “subject,” we found that medical history (C0262926) and lesion (C0221198) were the most common concepts in these criteria. We chose one eligibility criterion associated with each of these two concepts (Criteria 1 and 2). We also found that angina was the most common concept in the semantic category “Sign or Symptom” across the eligibility criteria for CAD trials. It is a common practice among physicians to specify angina using grades as specified by the Canadian Cardiovascular Society Angina Grading Scale (Campeau, 1976). We chose classification of patients into Class 1 (Criterion 3) and Class 3 angina (Criterion 4) as the other two eligibility criteria in our study. These criteria are shown in Table 4.1.
<table>
<thead>
<tr>
<th>No.</th>
<th>Criterion statement</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Known location of a coronary artery lesion.</td>
</tr>
<tr>
<td>2</td>
<td>History of revascularization procedure.</td>
</tr>
<tr>
<td>3</td>
<td>Ordinary physical activity such as walking and climbing stairs does not cause angina. Angina with strenuous or rapid or prolonged exertion at work or recreation.</td>
</tr>
<tr>
<td>4</td>
<td>Marked limitation of ordinary physical activity. Walking one or two blocks on the level and climbing one flight of stairs in normal conditions at normal pace.</td>
</tr>
</tbody>
</table>

Table 4.1: Criteria for textual entailment dataset.

### 4.2.2 Annotation process

Our annotation team consisted of three senior undergraduate nursing students, a second-year graduate-entry nursing student, and two physicians. The annotators for a single corpus were always the same and consisted of three students and one physician. We developed detailed annotation guidelines for the annotators for each criterion in consultation with the physician. The annotation process was carried out using CLINical Trials Criteria ANnotator (CLINICIAN), a tool developed at our institution. CLINICIAN is a secure web-based tool and has a simple user interface. After having logged in, annotation for a single note has the following workflow: the user selects a criterion of choice (screenshot shown in Figure 4.1) to annotate notes for. This brings up a webpage (screenshot shown in Figure 4.2) that displays the criterion statement at the top and a note beneath it. The user first determines whether the note has any text that is relevant to the criterion and marks the note as “Not relevant” otherwise. If found to be relevant, the user highlights all complete sentences in the note that contribute towards determining whether the patient meets the criterion, based on that particular note. After selecting all relevant sentences, the user
clicks on one of the three buttons “Yes,” “No” or “Maybe,” indicating an assessment of whether the patient meets the criterion. This action brings up the next note and the user repeats the workflow.

In order to ensure that these guidelines covered all cases, we carried out four training rounds. Each round involved annotations of ten notes per criterion (40 in total). All four annotators worked on the same set of 40 notes. Each round was followed by a discussion about the correctness of these annotations. Decisions associated with correctness of annotations were made through group discussions led by the physician. The annotation guidelines were updated after every training round to restrict differences among annotators. After completing the training, we calculated reliability of agreement among the four annotators based on annotations obtained for 20 notes per criterion (80 in total). We ensured that notes from the training rounds were not used in the agreement testing set. We used Fleiss’ kappa (Fleiss, 1971) as a metric of agreement, since more than two annotators were involved. We calculated three metrics of agreement. As outlined in the previous section, if
a note was found to be relevant, it was annotated for criterion eligibility. Thus, every note could be relevant (“Yes,” “No,” “Maybe”) or “Not Relevant.” The first metric calculated agreement on whether a note was relevant. The second metric calculated agreement for assessment of the criterion across all four answers. All notes were segmented into constituent sentences using the LingPipe sentence chunking module trained on MEDLINE biomedical abstracts. We calculated agreement on sentences found to be relevant by the annotators. A sentence was considered relevant only if the annotator highlighted it. Table 4.2 shows that the annotators achieve high agreement considering overall performance across all three metrics.

4.3 Baseline Systems: A rule-based approach

We implemented two lexical and two semantic baseline methods to understand the challenges associated with the task. We also evaluated the performance Lucene – an open source system for textual entailment. These algorithms are applied at a sentence level in
Table 4.2: Agreement for the dataset.

<table>
<thead>
<tr>
<th>Metric</th>
<th>Overall</th>
<th>Criterion 1</th>
<th>Criterion 2</th>
<th>Criterion 3</th>
<th>Criterion 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Note relevance</td>
<td>0.858</td>
<td>0.751</td>
<td>0.883</td>
<td>0.810</td>
<td>0.948</td>
</tr>
<tr>
<td>Criterion Assessment</td>
<td>0.795</td>
<td>0.769</td>
<td>0.818</td>
<td>0.611</td>
<td>0.846</td>
</tr>
<tr>
<td>Sentence relevance</td>
<td>0.799</td>
<td>0.758</td>
<td>0.784</td>
<td>0.825</td>
<td>0.813</td>
</tr>
</tbody>
</table>

every clinical note to determine a relevance score for every sentence with a criterion. These algorithms were applied to pairs of text and hypotheses, where text is a sentence N in the note and hypothesis is a sentence C defining the criterion. This ignores document-level features such as co-references, negations, document structure, etc. All note sentences having a positive score are considered to be relevant to the criterion. Performance is calculated using precision, recall and F-score of the system on a per-sentence basis.

1. **Lexical Matching**: This is one of the basic algorithms for textual inference that computes a score based on lexical overlap by matching the lemma for each word in C with the lemma for some word in N, ignoring stop words. The score is also incremented if a word in C is a synonym or an antonym of the word in N as determined by WordNet.

2. **Lucene**: Lucene is an open source text search engine; given a query of search terms, it returns a set of documents ranked by score of relevance. This system served as a pure information retrieval baseline in the search-based RTE challenge (Bentivogli et al., 2009), where every sentence is a document and the search terms in the query are all words in the hypothesis. Lucene can be configured using parameters that analyze the query terms and the text to be searched, in different ways. Configuration
parameters and evaluation of this approach by considering a limited number of top-ranked sentences was carried out as per the challenge guidelines.

3. **Concept Matching:** The UMLS brings together many biomedical vocabularies and standards. It is the largest thesaurus of biomedical concepts, mapping them to semantic types and associating them through hierarchical and non-hierarchical relations. We used MetaMap (Aronson, 2001) to identify UMLS concepts in every pair of C and N. A score was computed for every sentence N as the number of concepts in N that were exactly identical to a concept in C.

4. **UMLS Similarity:** UMLS::Similarity is a freely available open source tool (McInnes et al., 2009) that can be used to obtain similarity or relatedness between any two concepts from the UMLS. Given one or more ontologies and one or more hierarchical relations of interest, the tool treats the UMLS as a graph, with concepts as nodes and relations as edges. The tool has a library of different measures that compute similarity between two concepts using different graph-based properties. As in the previous approach, we identified UMLS concepts in every pair of C and N using MetaMap. A similarity score was computed between every pair of concepts in a given pair of C and N. The score for a sentence N was the sum of similarity scores its constituent concepts share with the concepts in a criterion C.

SNOMED-CT is the regarded as the most comprehensive healthcare terminology in the world. Pedersen et al. (2007) demonstrated that similarity measures between clinical concepts, computed using different metrics, had high correlation with physicians and human coders. They used parent-child relationships between concepts in SNOMED-CT to define the graph and computed similarity scores. We used the same relations, but on the version
<table>
<thead>
<tr>
<th>Method</th>
<th>Criterion 1</th>
<th>Criterion 2</th>
<th>Criterion 3</th>
<th>Criterion 4</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lexical Matching</td>
<td>0.368</td>
<td>0.086</td>
<td>0.182</td>
<td>0.070</td>
<td>0.176</td>
</tr>
<tr>
<td>Concept Matching</td>
<td>0.182</td>
<td>0.101</td>
<td>0.156</td>
<td>0.088</td>
<td>0.132</td>
</tr>
<tr>
<td>Lucene</td>
<td>0.400</td>
<td>0.051</td>
<td>0.161</td>
<td>0.050</td>
<td>0.166</td>
</tr>
<tr>
<td>UMLS Similarity Matching</td>
<td>0.150</td>
<td>0.367</td>
<td>0.475</td>
<td>0.075</td>
<td>0.267</td>
</tr>
</tbody>
</table>

Table 4.3: Comparative performance (F1) all four methods

2013AA of the UMLS. Of the different measures available in the tool, we used the metric described in Wu and Palmer (1994) to calculate similarity between a concept in the criterion and a concept in the note sentence. We chose this metric since the score is bounded between 0 and 1 and considers both the depth at which the two concepts are located in the graph as well as the distance between them. We varied the threshold for the similarity score to control the obtained precision and recall. We chose a value of 0.7 since it resulted in the best overall performance across all criteria.

4.3.1 Evaluation

We evaluated the performance of all four approaches by using standard information retrieval metrics of precision, recall and F1-score on a per sentence basis. Table 4.3 summarizes the performance of each method across every criterion based on the F1-score. We found that the semantic methods performed better than the lexical methods. While lexical methods had a higher precision, semantic methods had a higher recall. The UMLS similarity-based matching gave the best results. Most errors occurred because criterion statements had a low lexical overlap with the annotated sentences. Concepts present in the annotated sentences are related to concepts in the criterion but implemented methods fail short to capture these relations.
The UMLS::Similarity tool provides implementations of a number of similarity measures capturing different relationships between two concepts. This includes path-based measures, information content-based measures, and corpus-based measures. We conducted an exhaustive evaluation of all similarity metrics considered at different values of threshold tuned using cross validation. For corpus-based measures, we also analyzed the effect of the amount of data used to compute the similarity. The interested reader is referred to Shivade et al. (2015c) for these evaluations.

### 4.3.2 Limitations

The primary limitation of this study is the size of data used in these experiments. We used only 20 notes per criterion (80 notes in total) in our experiments. A larger dataset with annotations for relevant sentences might yield more accurate and robust results. Our annotation team consists of nursing students led by a physician. Having physicians or experts from the domain (in this case, cardiologists) as annotators could improve upon the quality of annotations. Moreover, the UMLS::Similarity tool is used in conjunction with a reference terminology to compute similarity between UMLS concepts. In this work, we followed previous work (Pedersen et al., 2007) and used SNOMED-CT as the reference terminology. Recent studies (Wu et al., 2012) have investigated the problem of choosing the right UMLS terminology for NLP tasks clinical notes. Our work also analyzes this particular problem for the 2014 i2b2 task but is not a part of this dissertation (Shivade et al., 2015d). Determining the right terminology or a combination of terminology sets to conclude similarity between concepts remains an unexplored topic of research.
4.4 Summary

In this chapter, we described the problem of textual entailment in the context of clinical trial recruitment. The objective of this work was to evaluate baseline methods that can identify sentences from clinical notes that are relevant to different clinical trial eligibility criteria. To do this, we constructed a dataset and showed that high agreement was achieved between annotators with varying levels of clinical expertise. This suggests that there is general systematic reasoning employed by humans for performing this task. We evaluated baseline computational methods from previous work for similar tasks and found that there is significant room for future investigation. We found that semantic methods gave better results than lexical methods and appear to be a promising direction for improvement. The biomedical domain has rich knowledge sources, such as the UMLS, which can be leveraged to perform intelligent inference in clinical text. While these are promising results as a first step in this direction, use of a learning algorithm to combine different approaches discussed in this chapter is an obvious next step. However, building a machine learning model requires annotated data which is expensive and hence scarce in the medical domain. Leveraging unlabeled data and building models using minimum training data would therefore be desirable. We discuss development of a machine learning model and the use of semisupervised and active learning techniques to achieve these goals in the following chapter.
CHAPTER 5: LEVERAGING UNLABELED DATA

The RTE challenges were designed so that solutions developed for this task could be applied to multiple practical problems. In the previous chapter we formulated the problem of clinical trial screening as a textual entailment task. We found that rule-based approaches have reasonable performance and knowledge from medical thesauri is helpful. In this chapter, we present the design of a supervised learning system for the clinical entailment task and then demonstrate how unlabeled data can be used to further improve performance. This is an important consideration since getting annotated data is time consuming and expensive in the clinical domain. We address the problem of limited, expertly annotated data in the clinical domain by (a) a self-training-based method to leverage large amounts of unannotated clinical text, and (b) an active learning approach to minimize the annotation costs involved. We present results for these approaches on a textual entailment dataset developed with clinical text, as well as the RTE-5 dataset, and compare them against multiple supervised baselines in both domains. The use of self-training results in significant improvements over a supervised baseline in the clinical domain, and also achieves the best result among all systems that had participated in the RTE-5 search task. We further demonstrate that active learning reduces the number of required annotations by more than 90% in both the clinical and RTE-5 datasets.
5.1 Dataset Creation

We begin by first outlining our annotated entailment dataset creation process. Unlike the data used in the previous chapter, this dataset is from a different institution. Moreover, it is largely driven by a clinical QA application (Ferrucci, 2012), under development. The work described in this paper is part of a larger effort to improve QA performance along the lines of Harabagiu and Hickl (2006) where they demonstrate the effectiveness of textual entailment within their QA application.

Given a corpus of text, creating an entailment search dataset involves two parts: hypotheses generation, and entailment annotation. The text corpus used in our system is the set of de-identified patient records associated with the QA application. The hypotheses originate from the questions of a QA dataset, which is quite similar to the process followed by RTE-1 through RTE-5 in creating hypotheses from TREC QA questions. The documents in the text corpus are then annotated with entailment relations for each of these hypotheses. RTE-6 and RTE-7 compromise on recall to reduce the annotation effort by filtering candidate sentences using a Lucene search. The nature of applications in the clinical domain places high priority on recall. Therefore, we do not filter sentences from our documents (i.e. clinical notes), but instead annotate the entire texts.

5.1.1 Hypotheses generation

We follow the methodology approved by RTE-1 through RTE-5 for converting questions from a QA dataset into entailment hypotheses. For a given question, the annotators first chose a relevant text snippet (T) that was suggested by a QA system as a candidate answer (Dagan et al., 2006). They then turned the question into an affirmative sentence with the candidate answer “plugged in” to form the hypothesis (H). For example, given
Question Hypotheses

<table>
<thead>
<tr>
<th>Question</th>
<th>Hypotheses</th>
</tr>
</thead>
<tbody>
<tr>
<td>When was the patient diagnosed with dermatomyositis?</td>
<td>The patient was diagnosed with dermatomyositis two years ago.</td>
</tr>
<tr>
<td>Any creatinine elevation?</td>
<td>Creatinine is elevated.</td>
</tr>
<tr>
<td></td>
<td>Creatinine is normal.</td>
</tr>
<tr>
<td>Why were x-rays done on the forearm and hand?</td>
<td>X-rays were done on the forearm and hand for suspected fracture.</td>
</tr>
</tbody>
</table>

Table 5.1: Example question hypotheses mappings.

the question, “Who is Ariel Sharon?” and the candidate answer “Israel’s Prime Minister, Ariel Sharon, visited Prague” (T), the hypothesis H is formed by turning the question into the statement “Ariel Sharon is Israel’s Prime Minister”, producing a positive entailment pair (H,T). In our QA efforts we use a clinical question set, which consists of questions about a set of patients and whose answers have been annotated within the Electronic Health Records (EHRs) of those patients. The questions are primarily seeking information that a physician may need to seek before or during a patient encounter. Examples are “When was the patient’s last colonoscopy?” or “What medications is she taking to control her hypertension?”

For generating hypotheses, the human annotators followed a process similar to that of RTE and converted questions into assertive sentences, with one difference. Neither did the annotators use a QA system, nor did they refer the patient’s EHR to get candidate answers. Instead, in order to generate hypotheses, they plugged in clinically-plausible answers that may or may not be correct for a given patient. Table 5.1 shows example hypotheses and their source questions.
Table 5.2: Comparison of class distribution for clinical datasets with RTE-5 Search Task.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Entailment</th>
<th>Non-entailment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train</td>
<td>293 (4.1%)</td>
<td>6,733</td>
</tr>
<tr>
<td>Dev</td>
<td>324 (4.0%)</td>
<td>7,768</td>
</tr>
<tr>
<td>Test</td>
<td>596 (5.6%)</td>
<td>9,870</td>
</tr>
<tr>
<td>RTE-5 Dev</td>
<td>810 (4.0%)</td>
<td>19,294</td>
</tr>
<tr>
<td>RTE-5 Test</td>
<td>800 (4.6%)</td>
<td>16,480</td>
</tr>
</tbody>
</table>

5.1.2 Entailment annotations

To generate entailment annotations, we paired a hypothesis with every sentence in a clinical note. Annotators were then asked to determine if the note sentence allowed them to conclude an entailment relationship with the hypothesis. For example, the text: “The appearance is felt to be classic for early MS.” entails the hypothesis: “She has multiple sclerosis”. To limit the number of annotations, hypotheses mapped from a given question were only paired with sentences from notes that contained an answer to the question as per the QA annotations. Two annotators who are medical experts participated in creating the dataset. We observed high inter-annotator agreement on a sample of 2883 text-hypothesis pairs (Cohen’s kappa = 0.81).

At the end of the annotation process, we had three datasets: training, development and test. A total of 243 hypotheses were annotated against sentences from 380 clinical notes, to generate 24,371 text-hypothesis pairs (see Table 5.2 for details).Datasets for the textual entailment search task are highly skewed towards the non-entailment class. Our datasets, while smaller than the RTE datasets, maintain a similar class imbalance.
5.2 Challenges with clinical data

Clinical narratives are written by healthcare professionals in a practice setting. These are substantially different from newswire text, which is consistent and well-formed in terms of English grammar, spellings and punctuation. Clinical notes are characterized by misspellings, abbreviations, incomplete sentences, inconsistent document structure, embedded semi-structured content (see Figure 5.1), and other undesirable properties of poor text quality. This poses a significant challenge to NLP systems (Meystre et al., 2008). Owing to the lack of punctuation, identifying sentence boundaries is not straightforward in clinical text. We used the approach described in (McCord et al., 2012) to identify sentence boundaries and calculated sentence length of each sentence in terms of number of words. The distribution for sentence length is significantly different for the development set of our clinical text ($\mu = 7.74$ and $\sigma = 7.85$) as compared to that of the development set for RTE-5 pilot data ($\mu = 21.4$ and $\sigma = 10.59$). This illustrates that clinical text and newswire text in RTE data are substantially different.

Another characteristic our annotators found in our data was that two or more sentences together were allowed to conclude entailment, but either one by itself was not sufficient to do so. For example: Consider the hypothesis: “The patient has a history of diabetes.” The progress note (see Figure 5.1) that allowed annotators to conclude that this hypothesis was true, had a section titled “Past Medical History”. Listed within the section are multiple disorders, including diabetes, with one-sentence descriptions for each. In order to conclude that the hypothesis is true, the annotators required evidence from both the section heading and the sentence that stated diabetes. Our parser identified each line in Figure 5.1 as a separate sentence. In this case our annotators annotated entailment relationship for the section
heading “Past Medical History”, and the sentence that mentioned diabetes “2. Diabetes mellitus”, and marked no entailment relationship for other sentences.

This characteristic is very similar to the phenomenon of partial entailment discussed by Levy et al. (2013), where a hypothesis can be broken down into one or more elements and text entails some of the elements, but not the hypothesis. It should be noted that the phenomenon of partial entailments is not found in RTE data, and makes the task of textual entailment search even harder. We discuss partial entailments in our dataset in Section 5.8.

5.3 Self-training

One of the primary goals of this research is to exploit unlabeled data, with the hope of augmenting the limited annotated data available to train our machine learning model. To that end, we describe in this section, our self-training algorithm for learning from unlabeled data.

Self-training is a single-view weakly supervised algorithm that has been widely used in the NLP community (Søgaard, 2013). In a standard supervised setup a model $M$ is trained using the training set $L$, and tested it on the test set $T$. In self-training, an existing model
labels unlabeled data. This newly labeled data is then treated as truth, and combined with the actual labeled data to train a new model.

Although we had limited number of labeled text-hypothesis pairs, we had a large number of unlabeled clinical notes. We paired the hypotheses from our training data with sentences from a set of randomly sampled subset of these unlabeled notes. Using these set of unlabeled text-hypothesis pairs $U$, we ran the following training regime: A model was created using the training data $L$, and tested on the unlabeled data $U$. We then identified all pairs where the posterior probability for the entailment class (note: we ignore the non-entailment class) was greater than a chosen threshold $\tau$. These pairs were then added to the training set $L$, which resulted in a bootstrapped training set $L'$. A new model $M'$ was then trained on $L'$ and tested on $T$. This was repeated iteratively until all pairs from $U$ were exhausted.

5.4 Active learning

In this section, we describe our active learning approach to minimize the amount of labeled data required to train the machine learning models.

Active learning is a popular training paradigm in machine learning (Settles, 2012) where a learning agent interacts with its environment in acquiring a training set, rather than passively receiving an i.i.d. sample from some underlying distribution. The purpose of developing models using this training paradigm is to achieve the best possible generalization performance at the least cost, where cost is usually measured as a function of the number of labeled examples. Active learning has also been a topic of interest for a variety of problems in NLP (Olsson, 2009).
Active learning is an iterative process, and typically works as follows: A model $M$ is trained using a minimal training dataset $L$. A query framework is used to identify an instance from an unlabeled set $U$ that, if added to $L$, will result in maximum expected benefit. Gold standard annotations are obtained for this instance and added to the original training set $L$ to generate a new training set $L'$. In the next iteration, a new model $M'$ is trained using $L'$ and used to identify the next most beneficial instance to be added to the training set $L'$. This is repeated until a certain stopping criterion is met.

The above approach is often simulated using a training dataset $L$ of reasonable size. The initial model $M$ is created using a subset $A$ of $L$. Further, instead of querying a large unlabeled set $U$, the remaining training data ($L - A$) is treated as an unlabeled dataset and queried for the most beneficial addition.

We carried out active learning in this simulated setting using a popular querying framework known as uncertainty sampling (Lewis and Gale, 1994). In this framework, the model $M$ trained using $A$, queries the instances in ($L - A$) for instance(s) it is least certain for a prediction label. For probabilistic classifiers the most uncertain instance is defined as the one where posterior probability for a given class is nearest to 0.5. To estimate the effectiveness of this framework, it is always compared with a random sampling framework, where random instances from the training data are incrementally added to the model.

5.5 Baselines

We compared our system with a number of baselines. These are entailment systems that participated in previous RTE challenges and are freely available for download. TIE: The Excitement Open Platform (EOP) is an initiative (Magnini et al., 2014) to make
tools for textual entailment freely available\textsuperscript{7} to the NLP community. The implementation for the Textual Inference Engine (TIE) (Wang and Zhang, 2009), a maximum entropy based entailment classifier, is made available through EOP.

**EDITS**: The Edit Distance Textual Entailment Suite (EDITS) by Kouylekov and Negri (2010) is an open source textual entailment system that is made available through the EOP platform.

**Lucene**: We used the exact same configuration as specified by the organizers (Bentivogli et al., 2009), but with a newer version of Lucene (v5.0) for our experiments.

These systems were built for data from RTE challenges. Therefore, they lack domain-specific knowledge and do not address issues described in Section 5.2. Therefore, we developed an entailment search system using supervised learning for clinical data.

### 5.6 Supervised learning system

The system uses the following feature sets, as input to a logistic regression with ridge estimator, implemented in Weka (Hall et al., 2009).

**Simple Similarity Features**: We discussed lexical overlap and UMLS concept overlap as two baselines in the previous chapter. These are cast as two simple similarity features in the learning system.

**Document Features**: Clinical notes typically have a structure and the content is often organized in sections (e.g. *History of Illness* followed by *Physical Examination* and ending with *Assessment and Plan*). We identified the section in which each note sentence was located and used them as features in the classifier.

**Paraphrase Features**: A number of features that attempt to measure text similarity were

\textsuperscript{7}http://hltfbk.github.io/Excitement-Open-Platform/
inspired by “passage scorers” in a QA system described in Murdock et al. (2012). The purpose of these scorers is to determine if a candidate answer passage is a paraphrase of a given factoid question. The pair of texts are matched in many different ways by these scorers, each producing one or more “match” features for the machine learning model. An example of these scorers is one that measures the overlap between a question and passage in terms of the number of common bi-grams. Another scorer measures the overlap between the dependency parse of the two. Yet another scorer computes the textual alignment between the two as a measure of their similarity. A total of 84 binary features were produced by our implementation of these passage scorers for the entailment task.

**UMLS Similarity Features**: Following results from the previous chapter, we used concept similarity scores obtained using the UMLS::Similarity tool as features in the model. We follow Pedersen et al. (2007) who have shown that using Systematized Nomenclature of Medicine - Clinical Terms (SNOMED-CT) as the ontology with parent-child relationships between concepts is useful for similar tasks with clinical text in the past.

### 5.7 Experiments

We conducted experiments using the baseline systems on both the RTE and clinical data. We found that systems that performed well on RTE data gave poor results on the clinical dataset. However our supervised system had a robust performance, and outperformed others in both domains. Further, we conducted experiments using self-training and active learning on both the datasets. The following sections describe these results in detail.

#### 5.7.1 Supervised baselines with RTE-5 data

Our problem setting is most similar to the RTE-5 Search Task. We trained all baselines on this task’s development set and tested on the task’s test set. Lucene was used as a
baseline for this task, considering the first 5/10/15/20 sentences top-ranked by the search engine for a given hypothesis. The configuration with the top 10 documents performed the best. None of the participating systems were able to beat this baseline, including our supervised system.

Of the 20 runs across eight teams that participated in this task, the median F1-Score was 30.14 and the best system Mirkin et al. (2009) achieved an F1-Score of 45.59. Table 5.3 summarizes the results of our experiments on the test set. EDITS and TIE perform slightly above the median and our system would have ranked third in the challenge.

### 5.7.2 Supervised baselines with clinical data

We trained all baselines using training data and calculated the performance of these systems on the development and test sets, (described in Section 5.1) respectively. Table 5.4 summarizes these results. We observe that entailment systems that have a respectable performance with RTE data fail to perform well on clinical data. In contrast, our system has a robust performance across domains and significantly outperforms the baselines on clinical data.\(^8\) The primary contributor of this is the set of paraphrase features in our model. In addition, our model also has a number components tailored for the clinical domain. These

---

\(^8\) McNemar’s test (\(p<0.05\)) for all experiments

---

<table>
<thead>
<tr>
<th>System</th>
<th>Precision</th>
<th>Recall</th>
<th>F1-Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lucene</td>
<td>46.9</td>
<td>47.5</td>
<td><strong>47.2</strong></td>
</tr>
<tr>
<td>EDITS</td>
<td>22.4</td>
<td>57.3</td>
<td>32.2</td>
</tr>
<tr>
<td>TIE</td>
<td>66.4</td>
<td>20.6</td>
<td>31.4</td>
</tr>
<tr>
<td>Our system</td>
<td><strong>77.4</strong></td>
<td>26.0</td>
<td>38.9</td>
</tr>
</tbody>
</table>

Table 5.3: Performance of supervised baselines on the RTE-5 Search Task Test Set.
Table 5.4: Performance of supervised baselines on clinical data (* indicates statistical significance) compared to Lucene and other baselines.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Development</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Precision</td>
<td>Recall</td>
</tr>
<tr>
<td>Lucene</td>
<td>20.3</td>
<td>36.5</td>
</tr>
<tr>
<td>EDITS</td>
<td>8.6</td>
<td>6.7</td>
</tr>
<tr>
<td>TIE</td>
<td>1.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Our system</td>
<td>42.4</td>
<td>20.6</td>
</tr>
</tbody>
</table>

range from fundamental tasks such as sentence chunking, abbreviation expansion, to advanced tasks such as recognition of UMLS concepts. Domain specific features such as capturing semantic relationships between UMLS concepts also contribute to better performance.

The performance of all systems on the clinical data is noticeably low as compared to the RTE data. An obvious difference in the two domains is that of training data size (see Table 5.2). However, obtaining annotations for textual entailment search is expensive, particularly in the clinical domain. Semi-supervised and active learning techniques are attractive solutions in such situations. We discuss our experiments using these methods in the following sections.

5.7.3 Semi-supervision with clinical data

The performance (F1-Score) obtained in a supervised setup on the development and the test sets is summarized in Table 5.4. We consider the performance of our model in this setup as the baseline, and investigate if the use of self-training is beneficial. Although we had a limited number of annotated clinical notes, we had a large number of unlabeled ones. We randomly sampled 10,000 clinical notes from our data, and paired each sentence with every
hypotheses created for the training dataset. This gave us a total of 623,600 text-hypothesis pairs. The threshold $\tau$ determines the confidence of our model for a text-hypothesis pair being classified to the entailment class. This threshold was varied incrementally from 0.1 to 0.9 in steps of 0.1. The best $\tau$ was tuned on the development set, and used to determine performance on the test set.

Figure 5.2 shows changes in precision and recall on the development set as threshold $\tau$ changes. We observe that as $\tau$ decreases from 0.9 to 0.1, the precision decreases, while the recall increases.

The F1-score (Figure 5.3) increases as the threshold decreases, reaches a peak value and then decreases again. On the development set, the best performance (F1=33.1) is obtained at $\tau = 0.2$ which is significantly greater than the performance obtained with the supervised baseline (F1=27.8). On the test set, this threshold of $\tau = 0.2$ results in a performance (F1=36.2) which is also significantly greater than the baseline (F1=22.6). The comparative performance on the test set is outlined in Table 5.5.
The variation in precision and recall as the threshold changes is very intuitive. When the threshold is low, there are more instances being added (n=10,799 at $\tau = 0.1$) into the training set. Therefore, recall is likely to benefit, since the model is exposed to a larger variety of text-hypothesis pairs. However, the precision is low since noisy pairs are likely to be added. When the threshold is high, fewer instances are added (n=316 at $\tau = 0.9$) into the model. Moreover, they are ones about which the model is quite certain. Therefore, the precision is comparatively higher, but recall is lower. One can observe in Table 5.5 that gains obtained from self-training are due to recall.

### 5.7.4 Semi-supervision with RTE-5 data

We also ran our self-training approach on the RTE search task datasets. In our experiments, we consider RTE-5 *train* as the training set and RTE-6 *test* as our development set (43,845 text hypothesis pairs). We treat RTE-7 as our unlabeled set (43,845 text hypothesis pairs).
pairs) by ignoring the labels. The threshold $\tau$ was tuned on the development set tested on the RTE-5 test set.

We observed similar variations in performance with RTE data as with the clinical data. At $\tau = 0.9$ fewer instances are added to the training set from the unlabeled data (n=49), while a large number of instances (n=2,861) are added at a higher threshold of $\tau = 0.1$. The best performance (F1=51.7) was obtained at $\tau = 0.3$, on the development set. This value of $\tau$ also resulted in the best performance (F1=53.8), on the test set. Similar to the experiments in clinical data, precision increased but recall decreased as $\tau$ increased. Again, it is evident from Table 5.5 that gains obtained from self-training are due to recall. It should be noted that, the self-trained system achieves a performance of F1=53.8, which is substantially better than the best performing system, (F1=45.5) by Mirkin et al. (2009) in RTE-5.

<table>
<thead>
<tr>
<th>Data</th>
<th>Clinical</th>
<th>RTE</th>
</tr>
</thead>
<tbody>
<tr>
<td>System</td>
<td>Precision</td>
<td>Recall</td>
</tr>
<tr>
<td>Supervised Baseline</td>
<td>42.0</td>
<td>15.4</td>
</tr>
<tr>
<td>Self-training</td>
<td>33.8</td>
<td><strong>38.8</strong></td>
</tr>
</tbody>
</table>

Table 5.5: Results for self-training on test sets. (*) indicates statistical significance

### 5.7.5 Active learning with clinical data

Our training data consists of 7,026 elements (refer to Table 5.2). The performance (F1-Score) obtained using this data in a fully supervised setup on the development and test sets is summarized in Table 5.4. We consider the performance of our model in this setup as the baseline, and investigate whether use of active learning is beneficial.
Starting with a model trained using a single randomly chosen instance, we carried out active learning using uncertainty sampling, adding one instance at a time. Thus, we created 7,025 models until all training instances were added to the model. After the addition of each instance, the model was retrained and tested on a held out set. To minimize the effect of randomization associated with the first instance, we repeated the experiment ten times and averaged the performance scores across the ten runs.

We follow previous work (Settles and Craven, 2008; Reichart et al., 2008) and evaluate active learning using learning curves on the test set. Figure 5.4 shows the learning curves for clinical data. While uncertainty sampling achieves performance equal to the supervised baseline with only 470 instances, random sampling requires 2,225 from a total of 7026 instances in the training data. Thus the active learner achieves performance equal to the baseline with only 6.6% of training data.
5.7.6 Active learning with RTE-5 data

In order to confirm the benefits of active learning, we repeated the experiments with RTE-5 Search Task data. The F1-Score obtained by our model using this data in a fully supervised setup, is summarized in Table 5.3. This F1-Score is considered as the baseline. We carried out both uncertainty sampling and random sampling using the RTE-5 test set as held out data. As with clinical data, the experiment was repeated ten times and the results were averaged across the ten runs. These results are shown using learning curves in Figure 5.5. Again, we find similar trends, with both sampling strategies outperforming the baseline using less than half of the training data, and uncertainty sampling learning faster than random sampling. While uncertainty sampling achieves performance equal to the supervised baseline with only 1,169 instances, random sampling requires 2,305 from a total of 20,104 instances in the training data. Thus the active learner achieves performance equal to the baseline with only 5.8% of training data.
<table>
<thead>
<tr>
<th>Dataset</th>
<th>Partial</th>
<th>Complete</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train</td>
<td>171</td>
<td>122</td>
<td>293</td>
</tr>
<tr>
<td>Dev</td>
<td>264</td>
<td>60</td>
<td>324</td>
</tr>
<tr>
<td>Test</td>
<td>481</td>
<td>115</td>
<td>596</td>
</tr>
</tbody>
</table>

Table 5.6: Distribution of partial and complete entailments in the corpus.

5.8 Discussion

In this chapter, we presented a supervised machine learning model for textual entailment search. While the system achieved a modest performance on data from the news and clinical domains, extensions using semi-supervised and active learning techniques are promising. In this section, we outline the contributions of this paper and provide explanations for some of the results.

As is evident from performance of systems that participated in the RTE challenges, textual entailment search is a challenging task. In the clinical domain, this is further exacerbated by issues of text quality and partial entailment (see Section 5.2). Partial entailment is particularly challenging in clinical text because the evidence for the entailment relationship is distributed across one or more sentences. Our medical experts noted the occurrence of partial entailments in our data, as summarized in Table 5.6. We observe that partial entailments significantly outnumber complete entailments. Handling partial entailments is a difficult problem, which is not explicitly modeled in our current entailment system (it processes hypotheses paired with single sentences). Modeling partial entailment will be part of our future work.

In order to make the best use of the available data, we explored semi-supervised and active learning techniques. Self-training works surprisingly well for the task of textual
Figure 5.6: Comparison of sampling strategies for active learning with clinical data development set.

entailment search in both clinical and newswire domains. In fact, as illustrated in our experiments, our self-training based system would have ranked first in RTE-5, outperforming various supervised systems. A possible reason for the success of self-training may be related to the class imbalance in our datasets (Table 5.2). Since we add only positive instances into the training set at each iteration, this helps the logistic regression classifier with the class imbalance. In contrast, multiple studies in the past have reported negative findings for self-training on other tasks (Søgaard, 2013). For example, Charniak (1997) and Steedman et al. (2003) report minor improvements or significant drop in performance with self-training for parsing.
Most studies using active learning report results where the active learner matches the performance of the model using the entire labeled set, but with fewer instances of training data. We found that active learning performs better than the fully supervised model with three held out datasets: the development and test sets in clinical data and the test set from the RTE data. We analyzed the possible reason for this phenomenon by tracking the proportion of positive instances to negative instances in the training set, as active learning proceeds. Figure 5.6 illustrates this on the development set of our clinical data.

The blue solid line shows that positive instances are “consumed” by the active learner at a faster rate than the negative instances when using uncertainty sampling. In contrast, random sampling consumes both positive and negative instances at an equal rate. Thus, the active learner tries to maintain a class balance during the initial phase of the training regime. In the later phase, more negative instances are added and thus the class imbalance hurts the performance, even though more training data is being added to the model. The plot also shows that 80% of the positive examples are added to the training set with less than 50% of the data.

5.9 Conclusion

This chapter explored the problem of leveraging unlabeled data for the task of textual entailment search in the clinical domain. We outline the issues while constructing a clinical dataset, for this task. Our results show that although available systems developed for textual entailment work well on newswire text, they fail to do so on clinical data. Our system performs well in both domains in a supervised setting. Additionally, we show that unlabeled data can be used effectively to boost the performance of this system using semi-supervised and active learning approaches. The simple self-training algorithm works surprisingly well
for the task and active learning with uncertainty sampling demonstrates that optimal perfor-
mance can be achieved with less than half of the training data. These techniques leveraging
unlabeled data are shown to be effective in both news and clinical domains.
CHAPTER 6: FINAL THOUGHTS

Use of electronic health records for screening patients into clinical trials has become an active area of research in the recent years. Clinical trial screening is an extensive task that poses multiple research questions which can be pursued as individual research problems in their own right. In this dissertation we discussed solutions using natural language processing to introduce automation into the clinical trial screening workflow. Specifically, we presented techniques to extract evidence from clinical notes present in a patient’s EHR.

In the first part, we focused on use of gradable terms in clinical text and demonstrated how their meaning can be reasoned about in various ways. In the second part, we investigated the problem of finding specific sentences in clinical notes that can enable a clinician to make a decision whether a patient meets specific eligibility criteria for a trial. We also addressed two important sub-problems of detecting negated concepts and determining sentence boundaries in clinical notes. We formulated all tasks as standard machine learning problems, collected relevant data from the EHR, gathered annotations from domain experts, and evaluated the performance of computational models to perform these tasks. This dissertation is the first end-to-end study to extract criterion-specific evidence from clinical texts.

6.1 Summary of Contributions

We first address the problem of understanding the meaning of gradable terms in clinical texts. This is covered in Chapter 3. The main contribution in addressing this problem
is a detailed analysis of usage of gradable terms and development of machine learning models to capture the linguistic properties exhibited by these terms. We demonstrate our approaches through experiments on an actual dataset of clinical notes from the EHR at our institution.

**Characterizing usage of gradable terms:** We empirically evaluated use of gradable adjectives in clinical notes. We reimplemented a previously published model for identifying gradable adjectives in newswire text and found that it performs surprisingly well with our clinical data. Analysis of the frequencies of these adjectives and their association with clinical concepts from UMLS revealed that there is a specific pattern for their usage, where certain nouns are modified by gradable adjectives more often than others. Finally, we showed that a simple Bayesian model can be used effectively to ground the meaning of gradable terms when they are used to describe medical concepts involving measurement of numerical quantities.

The main contribution here is that the Bayesian model presents a data-driven approach that can help in development of clinical standards in situations where there is a need to establish a precise relationship between adjectives and measurements. Grounding the meaning of adjectives in this way also allows us to infer scalar relationship between them. This work also serves as a confirmation of theories proposed by linguists regarding properties of gradability through a complex real-world dataset.

**Discovering scales:** We present an automatic pipeline to extract, cluster, and rank gradable adjectives from a large corpus. Our experimental results show that automatic clustering of gradable words based on vector representations produces promising results. However, we also observe that context is important for establishing a semantic intensity based ranking between domain-specific words.
The core contribution here is eliminating the dependency on manually curated resources like WordNet. Evaluation of clusters obtained using clustering shows that although these clusters are not identical to WordNet synsets, they represent scales with high accuracy. Ranking of adjectives within clusters shows that our method achieves a performance comparable to an approach that relies on lexical resources. Importantly, we also show that this approach can be used without any modifications for inferring adverbial scales.

In Chapter 4, we address the problem of extracting sentences from clinical notes that serve as evidence to determine whether a patient meets a given eligibility criterion for a clinical trial. We also discuss leveraging unlabeled data for this task in Chapter 5. The main contributions here are the task formulation, dataset preparation, evaluation of baselines from the open domain, and adapting a solution to the clinical domain.

Textual entailment: The problem of extracting evidence in clinical notes specific to eligibility criteria from clinical trials is modeled as a textual entailment task. We create the first entailment dataset for the clinical domain and evaluate baselines from the RTE challenges for this task. Through an exhaustive evaluation of UMLS::Similarity – a tool that uses the graphical structure of UMLS ontologies to infer similarity among various concepts – we show that such knowledge-based resources are very useful for the task and contribute significantly towards the task performance.

An important contribution here is demonstrating that the problem of textual entailment is harder in the clinical domain. Methods that work very well in the open domain do not perform as reliably in the clinical domain. We point out various challenges in this process ranging from issues with text quality to real-world characteristics such as low lexical overlap between text-hypothesis pairs. Our approach is generic and not limited to specific note-types or trials for specific diseases.
**Semi-supervised learning:** We developed a supervised learning model for the task of textual entailment. This linear model generalizes well across both the newswire and clinical domains. However, gathering annotations for textual entailment is time consuming and very expensive in the clinical domain. Therefore, we showed that a self-training approach that bootstraps examples from an unlabeled dataset can be used effectively to boost the performance of the supervised model.

The main contribution here is demonstrating that a robust model for textual entailment that works across domains can be developed. We show that unlabeled data can improve performance using the simple self-training method. This result is of great value to the community where obtaining annotations is very hard.

**Active learning:** While improvement in performance using labeled data is a nice result, it is also desirable to reduce the initial annotation effort required to develop a learning model. We empirically demonstrate this through experiments using uncertainty sampling. The datasets for textual entailment are skewed with very few positive examples. Our experiments show that more than 80% of positive examples can be revealed through active learning with less than 50% of the data.

The applicability of active learning is an important result since active learning has never been tried for the task of textual entailment even in the newswire domain. We conduct active learning in a simulated fashion where we feed the learning model with one instance at a time from the pre-annotated training data. The larger benefit of this result is however in the clinical domain where annotations are expensive since they require domain experts.

**Negation detection:** Detecting negated concepts in clinical notes has been carried out using rule-based systems. We demonstrate that such a rule-based system can be cast into a machine learning model by coupling it with bag of words features. Our experiments show
that the learning system captures more cases of negation, contributing to improved recall and thus an improved F1 score. We show that a semi-supervised approach also improves the performance.

The main contribution here is the use of a machine learning approach for this important task in clinical NLP. Rule-based approaches have been popular in the past since the availability of clinical data was rare. However, clinical datasets have become available over time. This work shows a simple adaptation of a rule-based system into a learning system. Given the performance of machine learning algorithms over rule-based systems for other NLP tasks, this is a direction in which a state of the art solution will evolve.

**Evaluation of sentence boundary detection:** The task of textual entailment is a sentence level task. Implicit in the problem definition is the assumption that a clinical note can be reliably segmented into constituent sentences. However, this is a non-trivial task in the clinical domain since authors of these documents make inconsistent use of periods and other punctuation. Systems developed for this task in the newswire domain perform poorly in the clinical domain. We conducted an evaluation of tools for sentence boundary detection across different datasets.

We identify an important shortcoming of multiple studies conducted in the biomedical NLP community. This is the use of off-the-shelf tools developed using newswire text on clinical data. This is an important contribution, since sentence boundary detection is a fundamental task for any NLP problem and results in many downstream errors.

### 6.2 Future Directions

The study of gradable terms presented in this work can be pursued in multiple directions for future work.
**Extending inference based on numerical quantities:** The Bayesian model proposed in this work is very simple. It works when adjectives ascribed to a medical concept depend on a numerical quantity, e.g. adjectives for anemia depend on hemoglobin measurements. We also discussed cases where the model breaks due to its dependence on two or more variables, e.g. descriptions of bone marrow cellularity depend on both age and cellularity values. It would be worth investigating if all dependent variables that influence the scale of comparison can be automatically inferred. A model would have to be designed that can consider an arbitrary number of dependent variables and their values to infer the appropriate adjective.

**Soft clustering:** We use K-Means clustering as a clustering algorithm in our work to group together gradable terms that belong to the same scale. Since K-means is a hard clustering algorithm allows each term to participate only in one scale. This inhibits modeling of polysemous words. For example, the adjective *great* can participate only in one of the two scales *good, great, excellent* and *large, great, huge, gigantic*. It would be worth investigating if a soft clustering algorithm can be used effectively to address this problem.

**Ranking of biomedical terms:** We found through our study that although clusters of biomedical terms (e.g. *mitochondrial, nuclear, ribosomal*) represent words on the same scale, it is hard to rank them unless more context is provided. This is an important limitation of our current work. It would be interesting to know if there exists a clustering that allows ranking of terms without providing further context. If not (which is often the case), the exact nature of the context required to infer ranking would have to be investigated.

**Other extensions:** We showed that the approach to infer clustering and ranking of gradable adjectives can be extended to adverbs. The weak-strong and strong-weak patterns used in this study followed previous work. These patterns have been hand-curated by
linguists in the past. It would be interesting to see if other patterns can be inferred using automatic pattern extraction techniques. If this step can be automated, the method can be possibly extended across multiple parts of speech and even multiple languages. Our work is also limited to unigrams. It would be interesting to see if phrases such as bigrams consisting of adverb-adjective pairs (e.g., somewhat unclear $< $ quite hard $< $ very difficult) exhibit properties of gradability.

**Textual entailment approaches:** Textual entailment is a hard problem attempted by multiple studies through research in the past. We argued that it is an even harder problem in the clinical domain with real-world data. Current research in the NLP community uses deep learning approaches for many problems to achieve state of the art results. Research on word level tasks has been satisfactory. However, directions of research for phrase level or sentence level tasks is not clear. Moreover, such methods require large amounts of data. This is not possible in the clinical domain. Therefore, development of models that can represent phrases or sentences as vectors with minimal training data would be ideal.

**Ranking and combining evidence:** We attempted the task of identifying sentences that entail an eligibility criterion in a single clinical note. The actual task of clinical trial screening is conducted on the entire EHR of the patient. This involves extracting multiple sentences, across multiple criteria. Hence, it would be necessary to rank evidence gathered across multiple notes for the same criterion to make a binary decision of eligibility with respect to that criterion. Decisions across multiple such criteria have to be aggregated to make a decision for an entire trial.
6.3 Conclusions

The number of clinical trials conducted across the globe has grown by leaps and bounds in the past decade. According to clinicaltrials.gov, the number of registered studies has grown from 5,633 in 2000 to 206,675 in 2015. Similarly, the adoption of EHRs is also on the rise. As of 2015, up to 95 percent of all eligible and critical access hospitals have demonstrated meaningful use of certified health IT through participation in the Centers for Medicare and Medicaid Services (CMS) EHR incentive programs.\footnote{Office of the National Coordinator for Health Information Technology. \textit{Hospitals Participating in the CMS EHR Incentive Programs}, Health IT Quick-Stat 45. \url{dashboard.healthit.gov/quickstats/pages/FIG-Hospitals-EHR-Incentive-Programs.php} February 2016.} Owing to the critical nature of clinical trials, and its impact on patient safety, while human intervention cannot be completely eliminated, introduction of automation into the screening workflow is a need of the hour.

This dissertation presents use of different NLP techniques to extract evidence from clinical notes to assist the task of clinical trial eligibility screening. We learn to identify, characterize, and ground the meaning of gradable terms used in clinical texts when they are used in association with measurement of numerical quantities. We propose an automatic pipeline to extract gradable adjectives, group them into clusters that represent the same scale and derive an ordering among adjectives within the scale. We also extend this method to gradable adverbs. When inferring patient eligibility requires more context, we demonstrate a method for extracting sentence-level evidence from clinical notes relevant to the eligibility criteria. We model this problem as a textual entailment task and evaluate different rule-based systems as well as a machine learning model for the task. Since obtaining annotations is time-consuming and expensive, we also present a self-training approach that
can leverage unlabeled data to boost the performance of the machine learning model. Finally, we also show that active learning with uncertainty sampling can significantly reduce the amount of training data required to develop a learning model for the entailment task.
APPENDIX A: EXTENDING NEGEX WITH KERNEL METHODS FOR NEGATION DETECTION

NegEx is a popular rule-based system used to identify negated concepts in clinical notes. This system has been reported to perform very well by numerous studies in the past. In this paper, we demonstrate the use of kernel methods to extend the performance of NegEx. A kernel leveraging the rules of NegEx and its output as features, performs as well as the rule-based system. An improvement in performance is achieved if this kernel is coupled with a bag of words kernel. Our experiments show that kernel methods outperform the rule-based system, when evaluated within and across two different open datasets. We also present the results of a semi-supervised approach to the problem, which improves performance on the data.\(^\text{10}\)

A.1 Kernel Methods

Our approach uses kernel methods to extend the abilities of the NegEx system. A kernel is a similarity function \(K\), that maps two inputs \(x\) and \(y\) from a given domain into a similarity score that is a real number (Hofmann et al., 2008). Formally, it is a function \(K(x, y) = \langle \phi(x), \phi(y) \rangle \rightarrow R\), where \(\phi(x)\) is some feature function over instance \(x\). For a function \(K\) to be a valid kernel, it should be symmetric and positive-semidefinite. In this section, we describe the different kernels we implemented for the task of negation detection.

\(^{10}\)This work has been published as Shivade et al. (2015b)
A.1.1 NegEx Features Kernel

The source code of NegEx\textsuperscript{11} reveals rules using three sets of negation cues. These are termed as pseudo negation phrases, negation phrases and post negation phrases. Apart from these cues, the system also looks for a set of conjunctions in a sentence. We used the source code of the rule-based system and constructed a binary feature corresponding to each cue and conjunction, and thus generated a feature vector for every sentence in the dataset. Using the LibSVM (Chang and Lin, 2011) implementation, we constructed a linear kernel which we refer to as the NegEx Features Kernel (NF).

A.1.2 Augmented Bag of Words Kernel

We also designed a kernel that augmented with bag of words the decision by NegEx. For each dataset, we constructed a binary feature vector for every sentence. This vector is comprised of two parts, a vector indicating presence or absence of every word in that dataset and augment it with a single feature indicating the output of the NegEx rule-based system. We did not filter stop-words since many stop-words serve as cues for negated assertions. The idea behind constructing such a kernel was to allow the model to learn relative weighting of the NegEx output and the bag of words in the dataset. Again, a linear kernel using LibSVM was constructed: the Augmented Bag of Words Kernel (ABoW).

A.2 Datasets

A test set of de-identified sentences, extracted from clinical notes at the University of Pittsburgh Medical Center, is also available with the NegEx source code. In each sentence, a concept of interest has been annotated by physicians with respect to being negated

\textsuperscript{11}From https://code.google.com/p/NegEx/
<table>
<thead>
<tr>
<th>Dataset</th>
<th>Affirmed</th>
<th>Negated</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>NegCorp</td>
<td>1885</td>
<td>491</td>
<td>2376</td>
</tr>
<tr>
<td>i2b2Train$_{mod}$</td>
<td>4476</td>
<td>1533</td>
<td>6009</td>
</tr>
<tr>
<td>PH subset</td>
<td>(1862)</td>
<td>(635)</td>
<td>(2497)</td>
</tr>
<tr>
<td>BID subset</td>
<td>(2614)</td>
<td>(898)</td>
<td>(3512)</td>
</tr>
<tr>
<td>i2b2Test$_{mod}$</td>
<td>8618</td>
<td>2594</td>
<td>11212</td>
</tr>
</tbody>
</table>

Table A.1: Number of affirmed and negated concepts in each dataset.

or affirmed in the sentence. The concepts are non numeric clinical conditions (such as symptoms, findings and diseases) extracted from six types of clinical notes (e.g., discharge summaries, operative notes, echo-cardiograms).

The 2010 i2b2 challenge (Uzuner et al., 2011) on relation extraction had assertion classification as a subtask. The corpus for this task along with the annotations is freely available for download.\textsuperscript{12} Based on a given target concept, participants had to classify assertions as either present, absent, or possible in the patient, conditionally present in the patient under certain circumstances, hypothetically present in the patient at some future point, and mentioned in the patient report but associated with someone other than the patient. Since we focus on negation detection, we selected only assertions corresponding to the positive and negative classes from the five assertion classes in the corpus, which simulates the type of data found in the NegEx Corpus. The i2b2 corpus has training data, partitioned into discharge summaries from Partners Healthcare (PH) and the Beth Israel Deaconess (BID) Medical Center. This gave us datasets from two more medical institutions. The corpus also has a test set, but does not have a split corresponding to these institutions.

\textsuperscript{12}From https://www.i2b2.org/NLP/DataSets/
Using the above corpora we constructed five datasets: 1) The dataset available with the NegEx rule-based system, henceforth referred to as the NegCorp dataset; 2) We adapted the training set of the i2b2 assertion classification task for negation detection, the i2b2Train\textsubscript{mod} dataset; 3) The training subset of i2b2Train\textsubscript{mod} from Partners Healthcare, henceforth referred to as the PH dataset; 4) The training subset of i2b2Train\textsubscript{mod} from the Beth Israel Deaconess Medical Center, henceforth referred to as the BID dataset; and 5) The adapted test set of the 2010 i2b2 challenge, henceforth referred to as the i2b2Test\textsubscript{mod} dataset. Table A.1 summarizes the distribution for number of affirmed and negated assertions in each dataset.

### A.3 Experiments

We implemented the kernels outlined in Section A.1 and evaluated them within different datasets using precision, recall and F1 on ten-fold cross validation. We compared the performance of each model against the NegEx rule-based system as baseline.

#### A.3.1 Within dataset evaluation

As can be seen in Table A.2, the NegEx Features Kernel performed similarly to the baseline (the improvement is not significant). However, the ABoW kernel significantly outperformed the baseline ($p<0.05$, McNemar’s test). Joachims et al. (2001) showed that given two kernels $K_1$ and $K_2$, the composite kernel $K(x, y) = K_1(x, y) + K_2(x, y)$ is also a kernel. We constructed a composite kernel adding the kernel matrices for the ABoW and NF kernels, which resulted in a further (but not significant) improvement.
<table>
<thead>
<tr>
<th>System</th>
<th>Dataset NegCorp</th>
<th>Dataset BID</th>
<th>Dataset PH</th>
</tr>
</thead>
<tbody>
<tr>
<td>NegEx (baseline)</td>
<td>94.6</td>
<td>84.2</td>
<td>87.3</td>
</tr>
<tr>
<td>NF Kernel</td>
<td>95.6</td>
<td>87.3</td>
<td>87.5</td>
</tr>
<tr>
<td>ABoW Kernel</td>
<td><strong>97.0</strong></td>
<td><strong>90.6</strong></td>
<td><strong>89.9</strong></td>
</tr>
<tr>
<td>ABoW+ NF Kernel</td>
<td><strong>97.3</strong></td>
<td><strong>92.4</strong></td>
<td><strong>91.3</strong></td>
</tr>
</tbody>
</table>

Table A.2: Within dataset performance of kernels based on F1-score using 10-fold cross validation. Bold results indicate significant improvements over the baseline (p<0.05, McNemar’s test).

<table>
<thead>
<tr>
<th>System</th>
<th>Precision</th>
<th>Recall</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>NegEx (baseline)</td>
<td>89.6</td>
<td>79.9</td>
<td>84.5</td>
</tr>
<tr>
<td>ABoW trained on</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NegCorp</td>
<td>89.9</td>
<td>79.3</td>
<td>84.2</td>
</tr>
<tr>
<td>PH</td>
<td>89.4</td>
<td><strong>88.1</strong></td>
<td><strong>88.7</strong></td>
</tr>
<tr>
<td>BID</td>
<td>89.2</td>
<td><strong>89.9</strong></td>
<td><strong>89.7</strong></td>
</tr>
<tr>
<td>i2b2Train_{mod}</td>
<td>89.9</td>
<td><strong>90.0</strong></td>
<td><strong>90.0</strong></td>
</tr>
</tbody>
</table>

Table A.3: Cross dataset performance on the i2b2Test_{mod} dataset given different training datasets.

### A.3.2 Cross dataset evaluation

In order to test the generalizability of our approach, we evaluated the performance of the ABoW kernel against the baseline. We trained the ABoW kernel on different datasets and tested them on the i2b2Test_{mod} dataset. Table A.3 summarizes the results of these experiments.
We found that the ABoW kernel significantly outperformed the baseline when trained on datasets that were generated from the same corpus, namely PH and BID. A kernel trained on \( \text{i2b2Train}_{\text{mod}} \), i.e., combining the PH and BID datasets performs better than the individually trained datasets. These experiments also tested the effect of training data size (PH < BID < \( \text{i2b2Train}_{\text{mod}} \)) on the kernel performance. We observed that the performance of the kernel increases as the size of the training data increases, though not significantly. The kernel trained on a dataset from a different corpus (NegCorp) performs as well as the baseline.

### A.3.3 Semi-supervised approach

We tried a semi-supervised approach to train the ABoW kernel, which we tested on the \( \text{i2b2Test}_{\text{mod}} \) dataset. We trained a kernel on the NegCorp dataset and recorded the predictions. We refer to these labels as “pseudo labels” in contrast to the gold labels of the \( \text{i2b2Train}_{\text{mod}} \) dataset. We then trained a semi-supervised ABoW kernel, ABoW\(_{ss}\) on the \( \text{i2b2Train}_{\text{mod}} \) dataset to learn the pseudo labels for this predicted dataset. Finally, we tested ABoW\(_{ss}\) on the \( \text{i2b2Test}_{\text{mod}} \) dataset. Table A.4 summarizes the results of these experiments. For ease of comparison, we restate the results of the ABoW kernel, ABoW\(_{gold}\) trained on the gold labels of the \( \text{i2b2Train}_{\text{mod}} \) dataset.

<table>
<thead>
<tr>
<th>System</th>
<th>Precision</th>
<th>Recall</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>NegEx</td>
<td>89.6</td>
<td>79.9</td>
<td>84.5</td>
</tr>
<tr>
<td>ABoW(_{ss})</td>
<td>89.7</td>
<td>82.1</td>
<td>85.7</td>
</tr>
<tr>
<td>ABoW(_{gold})</td>
<td>89.9</td>
<td>90.0</td>
<td>90.0</td>
</tr>
</tbody>
</table>

Table A.4: Semi-supervised models on the \( \text{i2b2Test}_{\text{mod}} \) dataset.
These results demonstrate that the kernel trained using a semi-supervised approach performs better than the baseline ($p<0.05$, McNemar’s test), but performs worse than a kernel trained using supervised training. However, supervised training is dependent on gold annotations. Thus, the semi-supervised approach achieves good results without the need for annotated data. This is an important result especially in the clinical domain where available annotated data is sparse and extremely costly to generate.

### A.4 Dependency Tree Kernels

Dependency tree kernels have been showed to be effective for NLP tasks in the past. Culotta and Sorensen (2004) showed that although tree kernels by themselves may not be effective for relation extraction, combining a tree kernel with a bag of words kernel showed promising results. Dependency tree kernels have also been explored in the context of negation extraction in the medical domain. Recently, Bowei et al. (2013) demonstrated the use of tree kernel based approaches in detecting the scope of negations and speculative sentences using the BioScope corpus (Szarvas et al., 2008). However, the task of negation scope detection task is different than that of negation detection. Among other differences, an important one being the presence of annotations for negation cues in the Bioscope corpus. Sohn et al. (2012) developed hand crafted rules representing subtrees of dependency parses of negated sentences and showed that they were effective on a dataset from their institution.

Therefore, we implemented a dependency tree kernel similar to the approach described in Culotta and Sorensen (2004) to automatically capture the structural patterns in negated assertions. We used the Stanford dependencies parser (version 2.0.4) (de Marneffe et al.,...
2006) to get the dependency parse for every assertion. As per their representation (de Marneffe and Manning, 2008) every dependency is a triple, consisting of a governor, a dependent and a dependency relation. In this triple, the governor and dependent are words from the input sentence. Thus, the tree kernel comprised of nodes corresponding to every word and every dependency relation in the parse. Node similarity was computed based on features such as lemma, generalized part-of-speech, WordNet (Fellbaum, 1998) synonymy and the UMLS (Bodenreider, 2004) semantic type obtained using MetaMap (Aronson, 2001) for word nodes.

Node similarity for dependency relation nodes was computed based on name of the dependency relation. A tree kernel then computed the similarity between two trees by recursively computing node similarity between two nodes as described in (Culotta and Sorensen, 2004). The only difference being, unlike our approach they have only word nodes in the tree. The kernel is hence a function $K(T1, T2)$ which computes similarity between two dependency trees $T1$ and $T2$. See (Culotta and Sorensen, 2004) for why $K$ is a valid kernel function. However, we got poor results. In experiments involving within dataset evaluation, the tree kernel gave F1 scores of 77.0, 76.2 and 74.5 on NegCorp, BID and PH datasets respectively. We also tried constructing composite kernels, by adding kernel matrices of the tree kernel and the ABoW kernel or NF kernel, hoping that they captured complimentary similarities between assertions. Although performance of the composite kernel was better than the tree kernel itself, there was no significant gain in the performance as compared to those of the reported kernels.
A.5 Discussion

We observe that while the precision of all the classifiers is almost constant across all the set of experiments, it is the recall that changes the F1-score. This implies that the kernel fetches more cases than the baseline. The bag of words contributes towards the increase in recall and thus raises performance.

It is instructive to look at sentences that were misclassified by NegEx but correctly classified by the ABoW\textsubscript{gold} system. The NegEx rule-based system looks for specific phrases, before or after the target concept, as negation cues. The scope of the negation is determined using these cues and the presence of conjunctions. False positives stem from instances where the scope is incorrectly calculated. For example, in “No masses, neck revealed lymphadenopathy”, the concept ‘lymphadenopathy’ is taken to be negated. The issue of negation scope being a shortcoming of NegEx has been acknowledged by its authors in Chapman et al. (2001). There were certain instances where the NegEx negation cues and the target concept overlapped. For example, in “A CT revealed a large amount of free air”, the target concept ‘free air’ was wrongly identified by NegEx as negated. This is because ‘free’ is a post negation cue, to cover cases such as “The patient is now symptom free”. Similarly, with ‘significant increase in tumor burden’ as the target concept, the sentence “A staging CT scan suggested no significant increase in tumor burden” was wrongly identified as an affirmation. Since the closest negation cue was ‘no significant,’ NegEx would identify only concepts after the phrase ‘no significant’ as negated. We also found interesting cases such as, the “Ext: cool, 1 + predal pulses, - varicosities, - edema.” where the concept ‘varicosities’ is negated using the minus sign.

We studied cases where NegEx made the right decision but which were incorrectly classified by our system. For example, in the assertion “extrm - trace edema at ankles, no
cyanosis, warm/dry”, the kernel incorrectly classified “trace edema” as negated. In “a bone scan was also obtained to rule out an occult hip fracture which was negative”, the concept “occult hip fracture” was incorrectly classified as affirmed. We found no evident pattern in these examples.

The tree kernel was constructed to automatically capture subtree patterns similar to those hand-crafted by Sohn et al. (2012). Although, it resulted in a poor performance, there are a number of possibilities to improve the current model of the kernel. Clinical data often consists of multi-word expressions (e.g., “congestive heart failure”). However, the word nodes in our dependency tree kernel are unigrams. Aggregating these unigrams (e.g., identification using MetaMap, followed by use of underscores to replace whitespaces) to ensure they appear as a single node in the tree could give dependency parses that are more accurate. Similarity for nodes involving dependency tree relations; similarity in our kernel is a binary function examining identical names for dependency relations. This could be relaxed by clustering of dependency relations and computing similarity based on these clusters. We followed Culotta and Sorensen (2004) and used WordNet synonymy for similarity of word nodes. However, open-domain terminologies such as WordNet are known to be insufficient for tasks specific to the biomedical domain (Bodenreider and Burgun, 2001). This could be coupled with domain specific resources such as UMLS::Similarity (McInnes et al., 2009) for a better estimate of similarity. Finally, since learning structural patterns is a complex task achieved by the tree kernel; training with a larger amount of data could result in improvements.
A.6 Conclusion

We demonstrate the use of kernel methods for the task of negation detection in clinical text. Using a simple bag of words kernel with the NegEx output as an additional feature yields significantly improved results as compared to the NegEx rule-based system. This kernel generalizes well and shows promising results when trained and tested on different datasets. The kernel outperforms the rule-based system primarily due to an increase in recall. We also find that for instances where we do not have additional labeled training data, we are able to leverage the NegEx Corpus as a bootstrap to perform semi-supervised learning using kernel methods.
APPENDIX B: EVALUATION OF SENTENCE BOUNDARY DETECTION

The sentence is a fundamental unit in NLP research. A typical NLP pipeline in the biomedical domain consists of identifying medical concepts, parsing, part of speech tagging, negation identification, word sense disambiguation, etc. as pre-processing tasks. Most of these begin with the fundamental task of identifying sentences, or what is more formally referred to as sentence boundary detection (SBD). SBD is a non-trivial problem, since the end-of-sentence punctuation marks are ambiguous. For example, a period can refer to an abbreviation, the end of a sentence, or a numeric decimal point. Exclamation points, question marks, quotation marks, ellipses, parentheses or a combination of these make the problem even more challenging.

In this work, we identify multiple popular NLP tools used by the biomedical community and evaluate their performance on the SBD task.\(^{13}\) Our primary focus is on comparing performance between general-domain text and clinical text. We use the toolkits off-the-shelf, i.e. without additional training or modification, and compare their performance over a variety of openly available datasets. More specifically, we evaluate cTAKES (Savova et al., 2013), Stanford CoreNLP (Manning et al., 2014), LingPipe (Lingpipe, 2008), SPECIALIST (McCray et al., 1994) and Splitta (Gillick, 2009) on the general-domain British National Corpus (bnc, 2007), transcripts from the Switchboard corpus (Godfrey and Holliman, 1993), the GENIA corpus (Kim et al., 2003) of biomedical abstracts, and a set of

\(^{13}\)This work has been published as Griffis et al. (2016)
clinical notes from the 2010 i2b2 challenge (Uzuner et al., 2011). We discuss important results here. Details of the study can be found in (Griffis et al., 2016).

B.1 Experimental Method

For BNC, Switchboard, and GENIA, we extracted a plaintext version of each document, along with sentence boundaries calculated during the extraction. For i2b2, since we obtained the corpus in plaintext format, we only calculated sentence bounds using the implicit annotations in the documents. All sentence bounds were calculated as character offsets from the beginning of the document. We used whitespace to maintain document structure when necessary, substituting one or more newline characters between paragraphs, sections, etc. We evaluated each toolkits performance on detecting sentence boundaries in each of the four corpora. We follow Read et al. (2012) by evaluating SBD using character offsets from the beginning of a document to denote its bounds. Thus, we ran each toolkit on each dataset and extracted the sentence boundaries from the results, which we compared against gold standard bounds extracted from the annotated corpora. Sentence boundaries assigned by the toolkits that were not present in the reference bounds were treated as false positives, and reference bounds that were missing in the toolkit output were considered false negatives. We note that under this design, there are no true negatives for the purposes of evaluation; negative results are reflected in decisions not made in the SBD task. Additionally, each correct bound detection counts twice (once for the end of the sentence and once for the start of the next sentence), and each error does so as well.
Table B.1: Precision(P), Recall (R), and F1 score(F) of sentence boundary detection task, evaluated for each tool on each dataset. The best results for each dataset are highlighted in bold.

<table>
<thead>
<tr>
<th>Tool</th>
<th>BNC</th>
<th>SWB</th>
<th>GENIA</th>
<th>i2b2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>P</td>
<td>R</td>
<td>F</td>
<td>P</td>
</tr>
<tr>
<td>St</td>
<td>0.89</td>
<td>0.77</td>
<td>0.82</td>
<td>0.59</td>
</tr>
<tr>
<td>$L_{GEN}$</td>
<td>0.83</td>
<td>0.65</td>
<td>0.73</td>
<td>0.58</td>
</tr>
<tr>
<td>$L_{ME}$</td>
<td>0.82</td>
<td>0.65</td>
<td>0.72</td>
<td>0.59</td>
</tr>
<tr>
<td>$S_{PSVM}$</td>
<td>-</td>
<td>0.55</td>
<td>0.30</td>
<td>0.39</td>
</tr>
<tr>
<td>$S_{PB}$</td>
<td>-</td>
<td>0.55</td>
<td>0.30</td>
<td>0.39</td>
</tr>
<tr>
<td>SLST</td>
<td>0.77</td>
<td>0.71</td>
<td>0.74</td>
<td>0.60</td>
</tr>
<tr>
<td>cT</td>
<td>0.73</td>
<td>0.75</td>
<td>0.74</td>
<td>0.67</td>
</tr>
</tbody>
</table>

B.2 Results

We report precision, recall, and F1-scores from this comparison in Table B.1. We do not report results for Splitta on BNC, because the toolkit implementation we used collapses consecutive whitespace characters, which are used to indicate document structure in our plaintext extraction of BNC. Our evaluation based on character offsets was therefore impractical for this pairing.

On a practical note, we found that the SPECIALIST and cTAKES tools could be run with no modification to the source code, but both required some post-processing to recover the character-level sentence bounds from the output. Splitta required some modifications to the source code in order to track sentence bounds at the character level, though this obviated the need for post-processing. LingPipe and Stanford CoreNLP required the most code-level work, needing API access to extract sentence bound information. The interested reader should refer to the source code for details.
Our experimental results show that all toolkits except cTAKES perform extremely well on the well-formed text in GENIA, and somewhat lower on the more mixed-domain text in BNC. The short, telegraphic sentences of Switchboard were extremely difficult for any of the toolkits to parse correctly: cTAKES performed the best, but still had a comparatively low F1-score (0.55). Finally, we saw that the clinical notes in the i2b2 corpus were even more difficult than Switchboard for every toolkit except cTAKES; its pre-training on clinical notes gave it an F1-score more than twice as high as that most of the other toolkits.

### B.2.1 Analysis by terminal character

To get a more detailed picture of SBD performance for different kinds of sentences, we further break down our results by the type of character used to end a sentence in each of the four corpora. We group characters into periods, all other punctuation, lowercase alpha characters, uppercase alpha, numeric characters, and all other characters (the “Special” category). Table B.2 shows the distribution of these different terminal character types in the four corpora.

\[
\begin{array}{ccccccc}
\text{Corpus} & \text{Period} & \text{Other} & \text{Lowercase} & \text{Uppercase} & \text{Numeric} & \text{Special} \\
& \text{Punctuation} & \text{< 1%} & \text{< 1%} & \text{< 1%} & \text{< 1%} & \text{< 1%} \\
\hline
\text{BNC} & 74\% & 15\% & 9\% & 1\% & < 1\% & < 1\% \\
\text{Switchboard} & 56\% & 33\% & 10\% & < 1\% & 0\% & < 1\% \\
\text{GENIA} & 99\% & < 1\% & < 1\% & < 1\% & 0\% & < 1\% \\
i2b2 & 51\% & 22\% & 9\% & 7\% & 9\% & 1\% \\
\end{array}
\]

Table B.2: Distribution of terminal sentence characters among the four corpora, by character type.

\[\text{14 Includes the following characters: ? ! : ; ( ) “ ’} \]
B.3 Conclusion

The task of identifying sentence boundaries is integral to many NLP applications. However, SBD has largely been treated as a solved problem in the biomedical domain, as it has been common practice to use off-the-shelf models to split sentences with minimal correction or adaptation for the specific task at hand. We describe and quantify the kinds of errors that arise from using several popular off-the-shelf SBD models on various domains, including clinical text. We find several interesting trends, primarily around domain-specific use of punctuation. In our clinical data, semicolons, colons, and newlines are heavily used and error prone, while periods caused errors in multiple corpora when used in unknown abbreviations, names, and numbers. Additionally, we note that both the ease of use of each toolkit and the additional work it performs on top of SBD varies widely, as does its runtime. Our observations indicate that SBD remains a difficult problem in the biomedical domain, and that the field will benefit from renewed effort to create or train efficient, domain-adapted models for this fundamental task.
APPENDIX C: LIST OF MODIFIERS FOR GRADABLE ADJECTIVES

actually, additionally, again, all, also, always, apparently, as, at, back, bilaterally, biopsy, bit, certainly, clearly, clinically, completely, currently, down, enough, especially, essentially, even, extremely, fairly, far, generally, here, how, however, indeed, initially, just, less, likely, little, mildly, moderately, more, most, mostly, much, no, now, obviously, only, otherwise, overall, particularly, possibly, pretty, previously, prior, probably, quite, rather, really, reasonably, recently, relatively, significantly, slightly, so, sometimes, somewhat, still, than, that, then, therefore, though, thus, too, unfortunately, up, usually, very, well, when, where
BIBLIOGRAPHY


24, 105


Ferrucci, D. (2012). Introduction to This is Watson. IBM Journal of Research and Development, 56(3.4):1:1–1:15. 70


Lingpipe (2008). LingPipe 4.1.0. 39, 60, 110


electronic health records. Journal of the American Medical Informatics Association, 21(2):221–30. 9, 10


Winn, R. J. (1994). Obstacles to the accrual of patients to clinical trials in the community setting. In *Seminars in Oncology*, volume 21, pages 112–117. 2


