5-2009

BIOMEDICAL LANGUAGE UNDERSTANDING AND EXTRACTION (BLUE-TEXT): A MINIMAL SYNTACTIC, SEMANTIC METHOD

Parsa Mirhaji
The University of Texas Health Science Center at Houston, Houston, Texas, USA

Dissertation

BIOMEDICAL LANGUAGE UNDERSTANDING AND EXTRACTION (**BLUE**-TEXT)

A MINIMAL SYNTACTIC, SEMANTIC METHOD

by

Parsa Mirhaji, MD

May, 2009

APPROVED:

Jiajie Zhang, PhD

Madurai Sriram Iyengar, PhD

Todd R. Johnson, PhD

, MD, PhD

Vipul Kashyap, PhD

BIOMEDICAL LANGUAGE UNDERSTANDING AND EXTRACTION (**BLUE-TEXT**)

A MINIMAL SYNTACTIC, SEMANTIC METHOD

A

DISSERTATION

Presented to the Faculty of
The University of Texas
School of Health Information Sciences
at Houston
in Partial Fulfillment
of the Requirements

for the Degree of

Doctor of Philosophy

By

Parsa Mirhaji, MD

Committee Members:

Jiajie Zhang, PhD
Madurai Sriram Iyengar, PhD
Todd R. Johnson, PhD
, MD, PhD
Vipul Kashyap, PhD

DEDICATION

This work is dedicated to my Father, who first introduced me to the magic and love of reading, and to my mother, who believed and unconditionally supported my endeavors in life.

PREFACE

This work is funded in its entirety by the Telemedicine and Advanced Technology Research Center (TATRC/USAMRMC), through “the Texas Technology and Training for Terrorism and Trauma (T5)” and “the Texas Science, Humanitarian Intervention, Education and Leadership in Disasters (TexSHIELD)” programs.

BIOMEDICAL LANGUAGE UNDERSTANDING AND EXTRACTION (**BLUE-TEXT**)

A MINIMAL SYNTACTIC, SEMANTIC METHOD

Parsa Mirhaji, MD, PhD
The University of Texas School of Health Information Sciences, 2008

Dissertation Advisor: Jiajie Zheng, Madurai S. Iyengar, Todd R. Johnson, Jack W. Smith

Clinical text understanding (CTU) is of interest to health informatics because critical clinical information frequently represented as unconstrained text in electronic health records are extensively used by human experts to guide clinical practice, decision making, and to document delivery of care, but are largely unusable by information systems for queries and computations. Recent initiatives advocating for translational research call for generation of technologies that can integrate structured clinical data with unstructured data, provide a unified interface to all data, and contextualize clinical information for reuse in multidisciplinary and collaborative environment envisioned by CTSA program. This implies that technologies for the processing and interpretation of clinical text should be evaluated not only in terms of their validity and reliability in their intended environment, but also in light of their interoperability, and ability to support information integration and contextualization in a distributed and dynamic environment.

This vision adds a new layer of information representation requirements that needs to be accounted for when conceptualizing implementation or acquisition of clinical text processing tools and technologies for multidisciplinary research.

On the other hand, electronic health records frequently contain unconstrained clinical text with high variability in use of terms and documentation practices, and without commitment

to grammatical or syntactic structure of the language (e.g. Triage notes, physician and nurse notes, chief complaints, etc). This hinders performance of natural language processing technologies which typically rely heavily on the syntax of language and grammatical structure of the text.

This document introduces our method to transform unconstrained clinical text found in electronic health information systems to a formal (computationally understandable) representation that is suitable for querying, integration, contextualization and reuse, and is resilient to the grammatical and syntactic irregularities of the clinical text. We present our design rationale, method, and results of evaluation in processing chief complaints and triage notes from 8 different emergency departments in Houston Texas. At the end, we will discuss significance of our contribution in enabling use of clinical text in a practical bio-surveillance setting.

TABLE OF CONTENTS

List of Tables xii

List of Figures xiii

List of Appendices xv

Preface 1
Outline of this dissertation 2

Chapter 1: Background 4
Current Approaches to Natural Language Processing: 5
Challenges of Clinical Text Understanding 8
A. Complexities of Natural Language: 8
B. Output representation 11
Chapter Summary: 14

Chapter 2: Prior art 17
LSP: The Linguistic String Project 17
MedLEE: A Medical Language Extraction and Encoding System 20
MPLUS: A Probabilistic Medical Language Understanding System 23
Ong and Wang: Object-Oriented Approach to Medical Text Understanding 26
Taira et al, 2001: Automatic Structuring of Radiology Free-Text Reports 29
Medsyndikate: Extraction of medical information from findings reports 32
Discussion: 36
A: Syntactic Analysis 37
B. Knowledge Representation 40
C: Output Representation Framework 46
D: Knowledge and Context Representation. 48
E: Encoding to Standard Vocabularies: 49
F: Evaluation 49
Chapter Summary; A Gap Analysis: 51
A: Syntactic Analysis: 51
B: Terminological Knowledge: 52
C: Semantic Knowledge 53
D: Domain Knowledge and Context Representation 53
E: Knowledge Representation Framework 53
F: Output Representation 54

Chapter 3: A Minimal Syntactic, Semantic Approach to Biomedical Language Undersrtanding and Extraction (BLUE-Text). 57
Motivation and History 57
Overview of the Method 59
Why the Semantic Web? 62
System Knowledgebases (Ontologies) 63
A: Syntactic Knowledge (Lexicon and Terminological Knowledge) 64

A.1. Syntactic Cues and Lexical Knowledge....64
A.2. The Lexicon....65
A.3. The Terminological Knowledgebase....66
A.4. Terminology Service vs. Terminological Ontology....70
B: The Semantic Knowledge:....72
Extensions of the Semantic Knowledge....74
C: The Domain Ontology (Domain Knowledge)....77
Syntactic Analysis....80
Text Preparation....80
Syntactic Analysis and Text Parsing....81
A minimal syntactic, language independent parser....82
Concept Mapping (Ontology Mapping):....86
The Conceptual Graph....89
Output Constructor....91
Evaluation....94
Introduction and Overview....94
The evaluation method....97
Chapter Summary:....102

Chapter 4: Results....105
Overview....105
The Test Set (Sample)....106
A) Extraction and Encoding....107
1. Clinical Observations....108
2. Body Site (anatomical concepts) Extraction....110
3. Modifier Concept (Qualitative and Quantitative) Extraction....110
B) Text Understanding....111
4. Anatomical sites of Clinical Observations....112
5. Quality and Quantity Modifiers of Clinical Observations....113
6. Negation of Clinical Observations....113
C) Consistency of extraction by quality of text....114
7. Quality of text and Extraction of Clinical Observations....114
8. Quality of text and assertions about Locus of the Clinical Observation....115
9. Quality of text and negation of Clinical Observations....116
D) BLUE-Text and Information Retrieval using UMLS-SN....117
10. UMLS-Semantic Types and retrieval of the Clinical Observations....118
11. UMLS-Semantic Types and retrieval of Locus assertions....119
12. UMLS-Semantic Types and Modifiers of the Clinical Observation....120
Chapter Summary:....120

Chapter 5: Discussion....122
Introduction....122
BLUE-Text Evaluation Paradigm:....124
The Gold-Standard: Good, Medium, Bad Quality of Text....127
Section A: Semantic Validity of BLUE-Text Output....130

A1. Overall Validity of the BLUE-Text Extraction Algorithm....132
A2. Interpretation of the PSA, Accuracy and Error rates: 133
A3. Extraction of Observations: 134
A4. Extraction of Body Site (Locus): 137
A5. Extraction of Modifiers:....137
A6. Quality and Quantity of Observations: 139
A7. Anatomical sites (Locus) of Clinical Observations: 139
A8. Negation of Observations: 140
A9. Modifier, Locus, Negation and The Effect of Knowledgebase: 141
A10. BLUE-Text and Good, Medium, and Bad Quality Text....143
Section B: Syntactic Validity of the BLUE-Text Output Representation 145
B1. Output Representation Framework 146
B2. Explicit (unambiguous) output representation:....147
B3. Expressive Output with Detailed Granularity....148
B4. UMLS Encoding 149
B5. Knowledge based output representation 149
B6. Conceptual Graph Vs. Task Specific Output Construction....150
Comparative analysis of the BLUE-Text validity 153
Comparative Analysis of the BLUE-Text Output 156
I. Structured (implicit) vs. Formal (explicit) Output....157
Richness, and Expressivity of the output....163
Support of Encoding to Standard Vocabularies:....169
Context and knowledge representation....171
Information Integration and Information Contextualization (reusability)....174
Is BLUE-Text a Decision Support Tool?....179
Section C: UMLS as domain knowledge and terminology 182
Section D: Limitations of BLUE-Text....183
Maturity of the OWL as the Knowledge Representation Language....184
D1. Computationally intensive process hinders performance 185
D2. Limited to Clinical Text (patient centric by design)....186
D3. Single Language (English only so far)....186
D4. Word Sense Disambiguation is modeled as variation....187
D5. Complex narratives and BLUE-Text Performance....188
Chapter Summary: Semantic, Syntactic and Pragmatic Validity of BLUE-Text....190

Chapter 6: Conclusions....193
Why text-understanding?....193
What is the BLUE-Text? 194
The BLUE-Text Evaluation Paradigm....195
The Semantic Validity of BLUE-Text....195
The Syntactic Validity of BLUE-Text....196
Future Works 197
Improve Modifier and Locus Algorithm....198
Word Sense Disambiguation and Normalization....198

Multilingual support.....199
Support Clinical Document Architecture (CDA) for information exchange.....199
References.....201

LIST OF TABLES

Table 1. Comparative Summary of Cited Systems 56
Table 2: Distance of related words in clinical text. 83
Table 3: Positive Specific Agreement and Negative Specific Agreement 100
Table 4: Quality of text and the distribution of extracted concepts and facts.....107
Table 5: BLUE-Text recall and precision rates for extraction and encoding 109
Table 6: BLUE-Text recall and precision for Subclasses of Clinical Observation 110
Table 7: BLUE-Text recall and precision rates explication of semantic relationships.....111
Table 8: BLUE-Text explication of assertions about categories of Locus 112
Table 9: BLUE-Text extraction of clinical observations by the quality of text.....115
Table 10: Quality of text and extraction of the anatomical relationships 115
Table 11: Quality of text and Negation of Clinical Observations 116
Table 12: BLUE-Text retrieval of Clinical Observations by UMLS Semantic Types 118
Table 13: BLUE-Text retrieval of anatomical assertions by UMLS Semantic Types 119
Table 14: BLUE-Text retrieval of modifier assertions by UMLS Semantic Types 120
Table 15: BLUE-Text overall evaluation results and compared by quality of text 121
Table 16: Quality of text and Validity of BLUE-Text.....144
Table 17. BLUE-Text and other text understanding systems 155
Table 18: MedLEE extraction to satisfy the query for System A.....164

LIST OF FIGURES

Figure 1: The complete PATIENT STATE Information-Format. 18
Figure 2: Information format of the LSP language processor 19
Figure 3: MedLEE Frames representing Radiology Findings and Modifiers....22
Figure 4: Sample MedLEE output 23
Figure 5: MPlus BNs applied to "temporal subdural hemorrhage". 25
Figure 6: A: Class hierarchy representing a phrase; B: Syntactic constituents of a phrase and its decomposition process 27
Figure 7: Dependency diagram showing relations between words. 30
Figure 8: Output knowledge frame for the sample sentence 31
Figure 9: Fragment of the Conceptual Knowledge (domain ontology) 33
Figure 10: MEDSYNDIKATE system architecture 35
Figure 11: A: the Input and B: the output of the semantic interpreter 36
Figure 12: Schematic depiction of the BLUE-Text processes and ontologies 61
Figure 13: Layers of frameworks constructing the Semantic Web technology platform 62
Figure 14: Definition of Quantifiable Component (number) in BLUE-Text 64
Figure 15: Lexemes and an instance of Negation (“Reject”) in English 65
Figure 16: UMLS-SN and MTH represented semantically. 67
Figure 17: BLUE-Text Semantic Model; A high level model to interpret clinical text 72
Figure 18: Extending semantics of Locus by UMLS SN and Negation 73
Figure 19: Temporal entities such as the Observation and the Encounter....75
Figure 20: A high level representation of Time Ontology....76
Figure 21: UMLS Semantic Network Represented in OWL 77
Figure 22: Definition of the ‘Substance Administration’ using UMLS-SN concepts 80
Figure 23: parse graph for the phrase “Large Blister in Toes and the Abdomen”....84
Figure 24: Parse Tree (A) and Dependency Diagram (B) equivalents of the Parse Graph in the Figure 23 86
Figure 25: Mappings of Parse Graph to the Terminological and Semantic knowledge 88
Figure 26: The Conceptual Graph, Evidence Spaces and Tokens of a phrase 90
Figure 27: The Formal RDF output corresponding to the Conceptual Graph in Figure 26....92

Figure 28: the contingency tables to calculate validity rates 94
Figure 29: Evaluation items for the text-understanding algorithm. 100
Figure 30: The SPARQL query to retrieve relevant information for evaluation 101
Figure 31: The evaluation process 102
Figure 32: Quality of text and extraction rates based on the quality of text 129
Figure 33: BLUE-Text extraction of different categories of concepts 133
Figure 34: BLUE-Text Performance for the subclasses of the Observation concept 135
Figure 35: BLUE-Text performance and subclasses of the Healthcare Procedure 136
Figure 36: Locus (Anatomical and Histological concepts) related to Clinical Observation 140
Figure 37: Extraction of Modifier, Locus and Negation of Clinical Observations 142
Figure 38. Quality of Text, Accuracy and Error 143
Figure 39: Quality of Text and Recall, Precision, Accuracy, and Error Rates 145
Figure 40. Layers of models in RDF output and the queries they support 152
Figure 41. Structure of the MedLEE and BLUE-Text Output.....158
Figure 42. BLUE-Text output in a generic RDF/OWL editor software 162
Figure 43. SPARQLquery to extract keywords and their associations from the text 165
Figure 44. SPARQL query to extract possible evidence for poisoning or toxicity 168
Figure 45. The SPARQL query to satisfy requirements of task C.....170
Figure 46. SPARQL query to demonstrate cross-vocabulary encoding and retrieval 171
Figure 47. BLUE-Text output as a formal OWL ontology.....173
Figure 48: Extension of BLUE-Text output to detect Dehydration.....176
Figure 49. Extensibility of BLUE-Text output to extract Food Poisoning.....178
Figure 50. Formal definition and proof for inclusion criteria using BLUE-Text output 181

LIST OF APPENDICES

Appendix A: Definition of Terms.....206
Appendix B: Sample MedLEE Output 211
Appendix C: Sample BLUE-Text RDF output (without imports).....213
Appendix D BLUE-Text RDF Output (N3 Format).....222

PREFACE

Although many techniques have been introduced for processing of unconstrained text in clinical settings, natural language processing (NLP)(Allen 1995; Carpenter 2007) is not yet conceived as an integral component of electronic health record (EHR) systems and its utilization and adoption does not match its potentials(Institute of Medicine 2003). Current methods of NLP are generally specialized for limited use in certain domains (e.g., tumor detection in chest radiography reports) and are not easily and efficiently extensible to new domains(Friedman and Hripcsak 1998; Friedman and Hripcsak 1999; Friedman 2005). Most importantly they produce incomplete results that limit their use to the boundaries of a pre-defined system, and prevent their adaptation to novel use cases, even in the same domain (Friedman, Hripcsak et al. 1995; Friedman and Hripcsak 1998; Friedman 2005). The incomplete and inexpressive output of current NLP systems is compounded by other representational issues (implicitness, ambiguity, granularity, etc.) that prevent extensive adoption of current NLP solutions in environments other than their original, and hinders interoperability with other information systems(Jain, Knirsch et al. 1996; Jain and Friedman 1997). It is not an easy task with the current state of the art to retrieve relevant information from an existing NLP output, and digest, interpret, and utilize it along with other information that may already exist in the system to answer new user queries(Lyman, Sager et al. 1991).

The CTU algorithm introduced throughout this dissertation aims at bridging some of the gaps between the current and desired state of the art for the processing of unconstrained text in clinical environments. The Biomedical Language Understanding and Extraction system (code named BLUE-Text) aims at construction of a dynamically flexible, customizable, consistent, formal, and explicit representation of unconstrained clinical text. BLUE-Text uses a formal information

and knowledge representation framework to represent outputs of the NLP processing and shares its domain and semantic knowledgebases to provide with a self-descriptive output that is immediately 'understandable' for computer programs for automated processing. We present the results of our formal evaluation of the systems reliability and discuss how it may enable automated contextualization and integration of unconstrained text with other heterogeneous health data.

Outline of this dissertation

This dissertation describes motivation, conceptualization and design of a clinical text understanding system developed at the University of Texas School of Health Information Sciences to address requirements of an extensive information integration system for public health surveillance and translational clinical research. Our objective was to bridge the gap that exists between the state of the art natural language processing systems for biomedical and clinical text, and practical needs for information sharing and multidisciplinary reuse of structurally and semantically disparate information, integrated from multiple heterogeneous sources.

Chapter 1 overviews the background, and outlines the domain problems, the environment, and the context of this dissertation. In this section basic principals of clinical text understanding and its significance is discussed in light of the current problems of translational clinical research. Existing frameworks for clinical text understanding are introduced and major challenges of design, conceptualization, and implementation of robust clinical text understanding systems are discussed. We highlight some criteria for optimal output representation that can be used to compare efficacy of existing tools and the system described in this dissertation.

Chapter 2 reviews the prior art and describes design and conceptualization of at least one exemplary system from each of the different classes of medical language understanding systems. A comparative discussion of the prose and cons, and design implications of each systems is provided. This chapter concludes with a gap analysis that set the stage for further research and development in this area and rationalizes and motivates this work.

Chapter 3 formulates the problem from authors perspective, provides the motivation, rationale and criteria that informed the conceptualization of the BLUE-Text system and the methods used to implement it. This chapter continues with an in depth discussion of the system design, and its components. At the end, a brief review of the challenges facing the evaluation of text understanding systems is provided and followed by detailed explanation of evaluation methods used to assess validity and reliability of the system.

Chapter 4 presents the results of a comprehensive and methodological evaluation as described in the chapter 3. This chapter is presented in 3 sections: a) Reliability measures for BLUE-Text as a traditional concept extraction and encoding system, b) Reliability measures for BLUE-Text as a novel text understanding system, and c) Comparison of the reliability of the BLUE-Text and consistency of its performance based on the quality of the clinical text.

Chapter 5 is devoted to the in-depth analysis of the BLUE-Text design and conceptualization. The discussions are focused on the design rationale and outcomes of the evaluation in light of the desiderata put forward in the chapter 1 for the next generation text understanding systems, the gap analysis provided in chapter 2, and the motivations introduced in the chapter 3. Advantages and known limitations of the system are discussed, and its implications on different areas of clinical and translational research are explained.

Chapter 6 concludes the dissertation, recapitulates its main points, and highlights the contributions and the significance of the BLUE-Text design to the field of biomedical language understanding as well as medical ontology research and knowledge engineering. Plans for the improvement of the system to address its known shortcomings are discussed, and future directions for research and development in the field are highlighted.

Each chapter ends with a summary of its content recapitulating the main points and concepts introduced.

CHAPTER 1: BACKGROUND

Clinical text understanding (CTU) lies in the intersection of natural language processing (NLP), artificial intelligence (Wikepedia-Free Online Encyclopedia), and computational linguistics, and deals with the conversion of patient health data spoken or recorded as unconstrained text, into formal representations readily interpretable (understandable) by computer programs. This is of interest to health informatics because important information in electronic health records (notes taken by nurses and physicians, chief complaints, discharge summaries, etc) are frequently represented as unconstrained text, and are used extensively by human experts to guide clinical practice, decision making, and to document delivery of care and health status. However unconstrained text is largely unusable by information systems for queries and computations. Furthermore, recent initiatives advocating for translational research call for generation of technologies that can integrate unstructured clinical data with structured data and provide a unified interface for queries, search and information retrieval (Center for Health Research 2006; Mirhaji, Zhu et al. 2009). It is also critical to be able to contextualize and repurpose clinical information from electronic health records systems and research databases for multidisciplinary

research in a collaborative and distributed environment envisioned by CTSA program. That is, technologies for the natural language processing (NLP) of clinical texts should be evaluated not only in terms of their validity and reliability in their intended environment, but also in light of their interoperability, and ability to support information sharing, integration and contextualization in a network of loosely coupled information systems(Ricci 2002). This vision adds a new layer of information representation requirements that needs to be accounted for when conceptualizing implementation or acquisition of clinical text processing tools and technologies for translational clinical research and practice.

Frequently, NLP techniques are used to identify, extract and annotate important concepts (e.g. diseases, procedures, medications, etc.) from the clinical text (***extraction***), and encode them into a corresponding terminology system (***encoding***)(Friedman, Alderson et al. 1994). NLP technologies have also been extensively used to retrieve parts of clinical text (***retrieval***)(Friedman and Hripcsak 1999), or to identify documents that meet search criteria (***classification***). The system described by (Heinze, Morsch et al. 2008) is an example of an NLP system for classification.

However we use the term "text understanding" to refer to a category of NLP technologies that extend these capabilities by a formal and explicit representation of the meaning of the text and the semantic relationships between concepts extracted from them (Hahn and Schnattinge 1997; Hahn and Romacker 1997).

Current Approaches to Natural Language Processing:

NLP and text understanding techniques are generally comprised of two main processes: 1) Syntactic Analysis that decomposes a given clinical content into its lexical constituents using

some parsing framework, and 2) Interpretation and Output Construction that maps lexical constituents of the text into a desirable data structure that may enable computations and retrieval. Syntactic knowledge, semantic knowledge, and domain knowledge are frequently cited as important components supporting these processes in most NLP algorithms.

The syntactic knowledge is used to determine the structure, constructors, and components of a textual representation rooted from grammar and syntax of the underlying language of the text (sentence, phrase, noun, verb, adjective etc). Most systems also incorporate a controlled vocabulary such as SNOMED-CT or UMLS as source of terminological knowledge to enable standards based encoding.

Semantic knowledge defines meaning of syntactic components by mapping them to unique concepts and sensible relationships between them *(Observation: Fracture; Descriptor: Comminuted, Location: Tibia*). Domain knowledge is frequently used to define, constrain or sanction generalizable relationships between concepts extracted from text in a domain of discourse (*Fracture hasLocation all (Skeletal_Structure Or Bone_Part)*). More importantly domain knowledge enables inferences and reasoning by computers (*Comminuted Fracture of Tibia Implies Severe Trauma or Injury in Lower Extremities*).

Databases, custom data structures and information formats, Bayesian Networks, object hierarchies, conceptual graphs, frames and ontology languages such as Web Ontology Language (OWL) are among knowledge representation frameworks used to capture and represent semantic knowledge within NLP systems, each with varying degree of expressivity, richness, extensibility, and support of computer reasoning, and implications on automation, interoperability, portability

and dynamic extensibility of the underlying NLP method. Many techniques have been introduced for NLP in clinical settings(Friedman and Hripcsak 1998):

Keyword-based Systems. Text is processed using some superficial features of text such as keywords and Regex patterns. Keyword-based systems are generally simple to implement and contain no syntactic, semantic, or domain knowledge. Relationships between terms and concepts cannot be established, although some methods use heuristics based on positional indexing of keywords as a measure of being related(Sager, Lyman et al. 1994).

Syntactic, Heuristic Systems. Syntactic knowledge and grammar of the underlying language is included. This method is generally used to identify simple noun phrases so that the relationships between words may be established. This method requires a lexicon containing word categories and a grammatical knowledge that can be used to detect and parse sentences into a target structure (Carpenter 2007). The underlying assumption in this method is that the syntactic knowledge base (the grammar and the lexicon) can be extended to account for the differences between text from different domains (sublanguages), and that the commonalities between sublanguages are stable and significant enough to make them reusable and portable between domains. Generic components to syntactically parse English text are available through open source community and widely adopted by many medical NLP systems.

Statistical Systems (e.g., Bayesian methods) are data-driven approaches that use historical data, lookup tables or a pre-annotated body of text (corpora) to train an algorithm to extract or encode concepts in the text, or establish a relationship. Some semantic knowledge may be presented through the structure and relationships between nodes of a Bayesian network(Harris 1968; Schütze 1999 ; Olszewski 2003). Statistical methods are relatively easy to implement and

resilient to incomplete or irregular textual information. They are generally sensitive and specific in specialized domains or limited tasks (such as detection of smoking history or diabetes status). But changes in the domain or use case may change its accuracy and may require retraining the system (Schütze 1999 ; Carpenter 2004).

Syntactic and Semantic (Hybrid) Systems rely on syntactic parsing to instantiate a concept, frame or template class as defined by the semantic knowledge. The syntactic relationships and domain knowledge are used to establish relationships among terms and to identify their semantic properties. Hybrid systems are knowledge intensive and complex, as they frequently require rich and comprehensive syntactic, semantic, and domain knowledge and generalizable rules to establish relationships among terms based on their semantic and syntactic properties (Sager, Friedman et al. 1986; Friedman and Hripcsak 1998; Carpenter 2007).

Challenges of Clinical Text Understanding

There are several difficulties associated with processing clinical text and presenting a complete and reliable output that can meet requirements of clinical applications (Sager, Friedman et al. 1986; Friedman and Hripcsak 1998; Carpenter 2007).

A. Complexities of Natural Language:

Clinical text is complex since natural language itself is the most expressive, rich and complex form of information representation. Human experts use, and intuitively understand complex combinations of qualifiers that further describe clinical observations with essentially important details such as degree information (*mild, severe*), relative change (*worse or better*), absolute or relative time (*12/06/1990, past 5 days, yesterday*), temporal relationship (*after, during, before*),

spatial coordinates (*up, left, front*), qualitative descriptors (*size, shape, look, texture, color, odor, etc*), quantifiers (*2mm, 150mg*).

There also exists modifier information that affects the meaning of the primary terms that must be accounted for. For example assertions about certainty, possibility, belief, doubt or negation (*evidence consistent with, no sign of*) may dramatically change the implications and significance of the a term (*Myocardial Infarction*). An optimal output should be able to capture all such contextual information and make it available for clinical applications.

There are many different ways of saying the same thing in natural language expressions. Friedman et al. reported over 850 different phrases associated with certainty information in their experience(Friedman and Hripcsak 1998). It is important for NLP systems to be able to detect these variations and reduce them in their output into a uniform and consistent representation. Controlled vocabularies are extensively used for this purpose by providing relevant synonymy, hypernymy, hyponymy, and meronymy (Voytovich 1999) information that can be used to reduce variability and to produce a consistent and standards based representation, but not all clinically relevant terms are represented in current vocabulary systems (e.g. modifiers such as *possible, negation,* or colors and shapes)(Hersh 2005). An optimal output representation should support standard vocabularies, as well as a consistent representation of concepts that are not supported by current vocabularies.

On the other hand, the very same natural language utterance may have different meanings in different contexts, in different times, and by different interpreters. *Lexical ambiguity* (word sense ambiguity) happens when a single word can assume different meanings based on some other contextual information. For example the term “Infarction” may mean “myocardial infarction” if

it appears in a discharge summary from a cardiology department but "pulmonary infarction" in a pulmonary department. *Structural ambiguity* compounds the lexical ambiguity when combination of terms may yield several valid interpretations. For example "*Large Tumor Excision*" may mean: (1) a tumor excision that was extensive, (2) a large tumor that was excised, or (3) the process of excising large tumors.

Natural language expressions frequently present complex iterative dependencies and nested relationships. For example in phrase "*no evidence of pleural effusion*", *evidence of* modifies *pleural effusion* and *no* modifies *evidence of.* In another example, a nested representation such as '*right mid lobar emboli*' should be unwound to '*Emboli in the middle lobe of the right lung*'. If the NLP output retains the nesting, access to the information will become difficult and in some cases not practical (Friedman and Hripcsak 1998). An optimal output should disentangle nested information systematically and make them accessible without losing important contextual information in the process.

Output may be inconsistent or the resulting knowledgebase may be invalid if contextual or modifier information in the natural language is not considered. For example, if relative body locations are ignored, the statement *'no infiltrate in left lower lobe but slight infiltrate in right lobe was noted'* will result in conflicting information in the output (*no infiltrate* vs. *infiltrate*). An adequate semantic and syntactic representation can avoid such inconsistencies (Friedman and Hripcsak 1998). In fact, an expressive output representation can enable computers to detect logically inconsistent information in an EMR (patient reported with *no allergy* in chart but *allergic to Penicillin* in admission notes, or *male* in one section and *female* in another).

Electronic health records contain unconstrained clinical text that are highly variable in use of terms and documentation practices, are irregular and sometimes incomplete and telegraphic, or have minimal commitment to the grammatical and syntactic structure of the language. In some cases and in an attempt to preserve patients own verbiages, utterances from more than one language may appear in the same piece of text (for example English and Spanish narrations in triage notes are frequent). This hinders performance of natural language processing technologies that rely heavily on the syntax of a particular language and grammatical structure of the text.

B. Output representation

Output of most NLP systems is specialized to support tasks within a certain domain or clinical information system (e.g., tumor detection in CXR reports, patient diabetes or smoking status). It is difficult, if not impossible to obtain output that can be readily contextualized and reused for a novel use case (Friedman C 1995; Friedman and Hripcsak 1999). Limited reusability (and interoperability) of the current systems can be due to:

Output representation Framework: Frequently, the output generated by NLP systems is a reflection of its intended use in a certain application or domain. This may hinder availability and make it difficult for secondary applications to access the information. Output of NLP systems should be represented using an extensible, consistent and easily understandable framework that enables access, and retrieval of information by other systems without requiring proprietary tools or intensive reprogramming. Use of standards based data structures and information representation frameworks such as XML and RDF may facilitate (although not sufficiently) information sharing and interoperability across systems by enabling standard-based information retrieval protocols such as XQuery or SPARQL.

Implicit, ambiguous and vague output representation: Output of NLP system cannot be made readily useful and 'understandable' by other systems if information is not explicitly and clearly defined in a way that allows computer systems to interpret them identically and precisely. For example, 'Respiratory distress' in one system may mean 'any sign or symptom' related to 'abnormal functional status' of the 'respiratory system' whereas another system may interpret it as 'at least two members of the set [dyspnea, shortness of breath, cyanotic entities, …] without fever'. Clearly, although overlapping, these are two different concepts and should not be treated as equivalent. Ambiguity arises when more than one concept can be entailed from one representation ('MI' may mean either 'Myocardial Infarction' or 'Mitral Insufficiency' in the same domain, or 'Mobility Impairment' and 'Mental Illness' in across different domains). Vagueness happens when there are no clear facts indicating whether or not the concept applies (unbearable pain: exactly, how much pain is bearable?). A clear term in one system may present with ambiguity and vagueness in another, unless formal definitions and semantic properties are available that can be used by all systems to disambiguate terms and interprete them identically and in a proper context, and automatically.

A sharable and reusable output is one that not only makes information readily accessible but also defines all its content explicitly (unambigously and precisely) for secondary users.

Level of detail and granularity: Different systems and use cases may need information at different levels of detail and granularity. For example, a decision support system may not need to know about left or right sided "Foreign body in the *Eye"* to invoke the appropriate treatment protocol, but a discharge summary should include such detail for practical reasons. An expressive framework is required to represent all relevant detail captured by the NLP system, regardless of the primary use cases and applications. The framework should be extensible to

accommodate new information with all associated detail without creating inconsistencies or incompatibility with previous information. Ideally the output representation should allow systems to access information with desirable granularity and on demand.

Encoding to standard vocabularies: Controlled vocabularies and terminology systems have long been proposed to establish an agreed point of reference between multiple systems when referring to the same medical concept (Humphreys, Lindberg et al. 1998; The College of American Pathologists 2003). This reduces heterogeneity and helps to explicate and disambiguate concepts. Mature and comprehensive medical vocabularies exist today (SNOMED-CT, UMLS, etc.) and can greatly enhance the reusability and interoperability if encoding of extracted concepts is consistently represented by NLP output. However, many of the essentially important concepts (including but not limited to those related to uncertainty, inability, absence and negation) are not covered by current terminologies(Cimino 2006; Smith 2006). In this case, the NLP systems should provide with a formalized framework that makes it explicit and understandable for other systems how these concepts and their relationships are represented and accessible from the output (Burgun 2006).

Knowledge and Context Representation. The true meaning of a term in a text depends greatly on the other contextual and modifier information. For example, the term ‘Penicillin’ has different connotations in allergy, medication history, home medications, and prescriptions sections of a medical record.

Frequently NLP systems need to be able to draw inferences, and use reasoning based on domain knowledge, assumptions, defaults and rules, in order to deal with complex real world use-cases and requirements (Friedman and Hripcsak 1998). For example, the term *opacity* in an X-ray

report may denote ‘*neoplasm’* or some other benign conditions if used along with certain modifiers. An application searching for a possible *neoplasm* may need to recognize reports containing the finding *opacity* qualified by certain modifiers as relevant to the search criteria. In this example relevant modifiers may be ‘size: *large* or *gt 2 cm*, ‘shape: *ill-defined’*, ‘distribution: *cluster’*.

Frequently, this knowledge is omitted from the output representation based on the assumption that the systems utilizing the output share the same context and commit to the same heuristics, assumptions and presumptions. Absence of this information prevents from secondary use of NLP output or adoption of these systems in new environments as the output is not provable and cannot be validated. An optimal output representation should be self-descriptive in a way that enables information systems to access the relevant context, and the semantics of the information represented in the output, in a way that conclusions can be traced back to the associated evidence in the text and the logic and rules of inference in the knowledgebase.

Chapter Summary:

Clinical text understanding (CTU) lies in the intersection of natural language processing (NLP), artificial intelligence (Wikepedia-Free Online Encyclopedia), and computational linguistics, and deals with the conversion of patient health data spoken or recorded as unconstrained text, into formal representations readily interpretable (understandable) by computer programs.

NLP and text understanding techniques are generally comprised of two main processes: 1) Syntactic Analysis that decomposes a given clinical content into its lexical constituents using

some parsing framework, and 2) Interpretation and Output Construction that maps lexical constituents of the text into a desirable data structure that may enable computations and retrieval.

Syntactic knowledge, semantic knowledge, and domain knowledge are frequently cited as important components supporting these processes in most NLP algorithms. Databases, custom data structures and information formats, Bayesian Networks, object hierarchies, conceptual graphs, frames and ontology languages such as Web Ontology Language (OWL) are among knowledge representation frameworks used to capture and represent knowledge within NLP systems, each with varying degree of expressivity, richness, extensibility, and support of computer reasoning, and implications on automation, interoperability, portability and dynamic extensibility of the underlying NLP method.

There are several difficulties associated with processing clinical text and presenting a complete and reliable output that can meet requirements of clinical applications (Sager, Friedman et al. 1986; Friedman and Hripcsak 1998; Carpenter 2007). Clinical text is complex since natural language itself is the most expressive, rich and complex form of information representation. Electronic health records contain unconstrained clinical text that are highly variable in use of terms and documentation practices, are irregular and sometimes incomplete and telegraphic, or have minimal commitment to the grammatical and syntactic structure of the language. In some cases and in an attempt to preserve patients own verbiages, utterances from more than one language may appear in the same piece of text (for example English and Spanish narrations in triage notes are frequent).

Apart from complexity of the natural language and its representation within health information systems, the output of current NLP systems are not optimized for multidisciplinary reuse. Output

of most NLP systems is specialized to support tasks within a certain domain or clinical information system (e.g., tumor detection in CXR reports, patient diabetes or smoking status). Limited reusability (and interoperability) of the current systems can be due to the following factors:

Frameworks used to represent NLP output are not efficiently expressive

Output of NLP systems are often implicit, ambiguous and vague

Information in the NLP output are not presented with sufficient detail and granularity

Frequently NLP output does not support mappings to standard vocabulary systems

NLP output does not include the knowledge required to interpret information in an appropriate context.

An optimal output representation should be self-descriptive in a way that enables information systems to access the relevant context, and the semantics of the information represented in the output is explicitly and clearly represented with appropriate level of detail and granularity for retrieval by computer programs.

CHAPTER 2: PRIOR ART

Since 1960s the National Science Foundation and National Institute of Health have sponsored basic research in language and information as the basis for information systems of the future. Recent advances of the information sciences, computer sciences and the evolution of biomedical research and clinical and public health practice has created a new impetus for creation of robust text understanding algorithms as major components of future IT infrastructure that can support personalized healthcare, reduce medical errors, and facilitate collaborative and translational research.

LSP: The Linguistic String Project

The LSP system at New York University (Sager 1994) was one of the first significant steps towards conceptualization of clinical text understanding systems that extend beyond extraction and encoding of biomedical concepts, towards identifying their relationships according to the text. LSP builds on top of a parser algorithm and specialized programming language for computational interpretation of grammar of English language(Sager, Lyman et al. 1965; Sager 1981; Sager 1986). The LSP augments this core technology by a sublanguage grammar and an information structure that are specific to the clinical narrative.

The overarching hypothesis in this work is that "each medical discipline or sub-discipline expresses its content in relatively stereotyped sentence types based on its specialized word usage" and that these stereotypes are stable and share in large part, the same constructors and features. This implies that the grammar of the medical sublanguage and its word classification scheme remain stable over a range of clinical areas, that the lexicon associated with these classes require little modification (except additions when moving from one domain to another). The output of LSP is mapped into a database to enable querying.

The syntactic and semantic properties of clinical content (clinical sublanguage and its grammar) are captured through series of specialized data structures called Information Format (I-F). I-Fs are pre-coordinated templates for holding the words of a sentence (or sentence-part) that correspond to a pattern or stereotype of the sublanguage. Each I-F is about a topic and is composed of slots representing a syntactic component, corresponding text, and lexical class associated with that text. A slot can be expanded by (or hold as value) another I-F. Figure 1 depicts hierarchical expansion of the Patient State I-F.

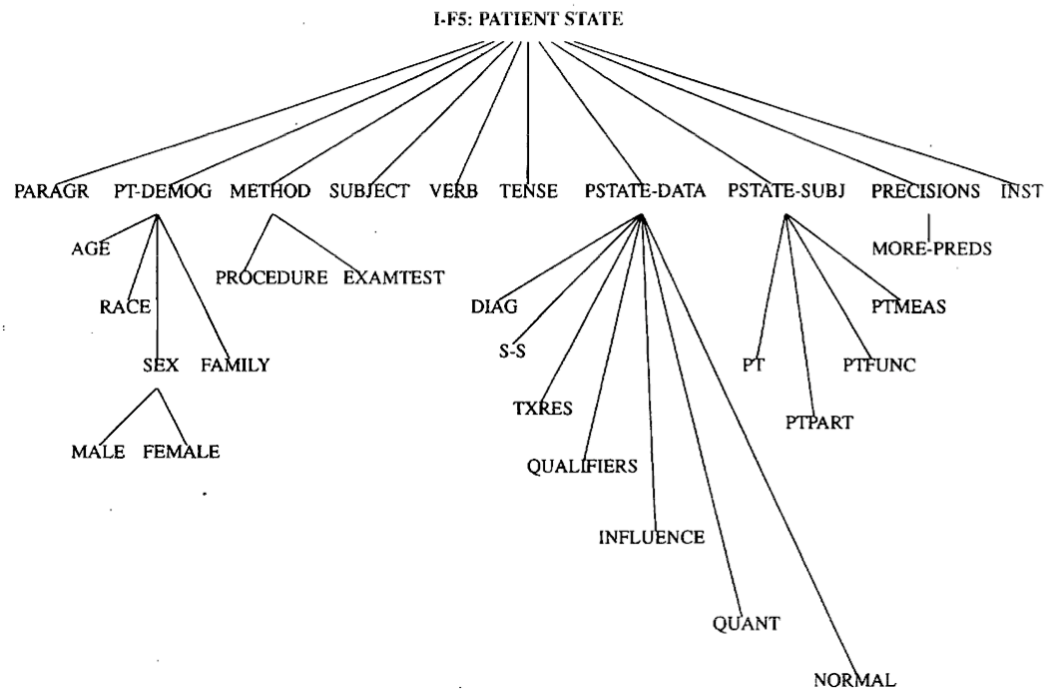


Figure 1: The complete PATIENT STATE Information-Format[α].

NEGATION, MODALITY (uncertainty) and TIME are modifiers that apply to all patterns and are defined as separate templates and inserted into the instantiated I-F to which they apply before loading to the relational database.

[α] PSTATE-DATA finding about the medical subject. Nonmedical subjects and verbs default to SUBJECT and VERB. METHOD contains the procedure or physical examination test that gives rise to the finding. holds additional medical modifiers and INST holds mentions of institutional personnel or departments (not proper names).

There is almost a 1:1 correspondence between the medical lexical classes (Fig. 3) and the similarly named slots of the I-F. That is syntactic procedures are responsible for parsing and segmentation of text into appropriate slots and their expansions to form complete units as outlined by the I-F (template). Similar process exists to correctly assign negation and uncertainty markers, and to identify time expressions for further processing. Figure 2 shows a clinical sentence analyzed into I-F occurrences of the Patient State type (the slots of the template are in capital letters and the words as extracted from the text are in bold italics, followed by the medical lexical classes of the word(s) in the slot. Specialized I-Fs are implemented that correspond to LABORATORY, TREATMENT, RESPONSE, DIAGnosis, HISTORY, physical EXAMination, etc.

```
*SID=GLCBA 002B.1.01
* THIS 29 YEAR OLD GIRL WHO IS KNOWN TO BE ASTHMATIC FOR 15 YEARS PRESENTED
* WITH A 4 DAY HISTORY OF STEADILY INCREASING EXERTIONAL WHEEZY DYSPNOEA
* WITH A COUGH PRODUCTIVE OF GREEN SPUTUM .

(CONNECTIVE (REL-CLAUSE (CONN = 'who'))
        (CONNECTIVE (RELATION (CONN = 'with' (H-CONN)))
                (I-F5: PATIENT STATE
                        (PSTATE-SUBJ (PT = this 29 (QNUMBER) year (NUNIT NTIME1)
                                        old (H-AGE) girl (H-PT)))
                        (VERB = presented (H-PTVERB) with)
                        (TIME (TM-PERIOD = a history (H-TMPER))
                                (Q-N    (NUM = 4 (QNUMBER))
                                        (UNIT = day (NUNIT NTIME1))))
                        (TENSE = [PAST])
                        (PSTATE-DATA (S-S = of exertional (H-PTFUNC) wheezy (H-INDIC)
                                        dyspnoea (H-INDIC))
                                (QUANT = steadily
                                        increasing (H-CHANGE [(MORE)])))))
                (I-F5: PATIENT STATE
                        (PSTATE-DATA (S-S = a cough (H-INDIC) productive
                                        of ('OF') green sputum (H-INDIC))))
        (CONNECTIVE (EMBEDDED (CONN = '[EMBEDDED-OBJ]'))
                (I-F00: SENTENTIAL OPERATOR
                        (VERB = is (VBE) known)
                        (TENSE = [PRESENT]))
                (I-F5: PATIENT STATE
                        (PSTATE-SUBJ (PT = girl (H-PT)))
                        (VERB = be (VBE))
                        (PSTATE-DATA (DIAG = asthmatic (H-DIAG)))
                        (TIME (TPREP1 = for ('FOR'))
                                (Q-N    (NUM = 15 (QNUMBER))
                                        (UNIT = years (NTIME1)))))))
```

Figure 2: Information format of the LSP language processor

The LSP system can be best explained as an information-formatting program composed of five modules that operate in sequence and sentence by sentence throughout the document:

Parse the sentence into its grammatical components using a grammar that embodies syntactic structures and constraints.

Filter out alternative syntactic analyses that are not semantically correct based on established patterns of medical word-class combination (medical co-occurrence patterns or “selection lists”).

Transformation module makes every conjunctional sub-statement complete to reduce syntactic variation (e.g., by expanding “pain in epigastrium and right lower quadrant” to “pain in epigastrium and pain in right lower quadrant”). Regularization treats the connective structure, turning the whole into Polish notation. The formatting module places sentence words into the appropriate slots of the I-Fs and prepares the output for mapping into the current database structure.

Sager et al evaluated the system through queries for the presence of 13 asthma-health-care quality assurance criteria in a database generated from 59 discharge summaries, and reported 95.7% precision rate for the 28 training set and 98.6% for the 31 test set. Recall was calculated using counts of major omissions only and was 93.9% for the training set and 92.5% for the test set.

MedLEE: A Medical Language Extraction and Encoding System

MedLEE (Friedman, Hripcsak et al. 1995) is a clinical text extraction and encoding system created by Carol Friedman in collaboration with the Department of Biomedical Informatics at Columbia University, the Radiology Department at Columbia University, and the Department of Computer Science at Queens College of CUNY. The system was originally designed for automatic parsing of radiology reports but has been expanded since then into a more generalized

medical language processing system used in other domains such as cardiology, respiratory disease, and other domains.

Recognizing some of the limitations of the LSP project MedLEE targets to provide a language processing system with the following distinctions: consistent encoding to controlled vocabularies, an easy to maintain and extend syntactic processing algorithm for clinical content represented in English language, a semantically oriented output representation that facilitates automated information retrieval.

The grounding hypothesis of the MedLEE design is that clinical content can be modeled semantically and prior to the implementation of the NLP system in such a way that it can capture majority of the expressions and information incorporated in clinical documents and such semantic models can augment parsing and processing of clinical text in a way that its output is more consistent, and reusable..

MedLEE replaces the steps 1-3 of the LSP system with a parser based on an extended context free grammar that creates a parse tree from each input sentence. The extended grammar includes translation rules to map the grammatical structures from the parse tree into target forms. It also includes constraints that specify well-formedness restrictions for the grammatical structures. The semantic grammar used to process and transform the text contains approximately 350 grammar rules.

In order to map the clinical information contained in the parse tree into a structured form (template), a formal model was designed to represent the clinically salient information and patterns identified in clinical content. The fundamental design of this model is based on the I-Fs developed by the LSP project. A frame based knowledge representation language is used to

represent the forms. Forms embody systems semantic knowledge about clinical concepts and their relationships as expressed in the clinical text.

Figure 3 illustrates a simplified version of the model for Radiology Report findings and Modifiers using the linear notation for Conceptual Graphs[β].

```
[Rad Finding Structure]-
     (Central Finding)->[Rad Finding:{*}]
     (Bodyloc Mod)->[Bodyloc:{*}]
     (Finding Mod)->[Modifier:{*}]

[Modifier]-
     (Certainty Mod)->[Certainty:{*}]
     (Degree Mod)->[Degree:{*}]
     (Change Mod)->[Change:{*}]
     (Status Mod)->[Status:{*}]
     (Quantity Mod)->[Quantity:{*}]
     (Descriptor Mod)->[Descriptor:{*}]
```

Figure 3: MedLEE Frames representing Radiology Findings and Modifiers

MedLEE provides with a regularization method similar to LSP (step 4) to account for multiword patterns and composite concepts to reduce variability of expressions in text.

Earlier versions of the MedLEE system mapped (encoded) a target term in the MedLEE output to a Medical Entities Dictionary (MED) code(Cimino 2000). But more recent revisions of the system use a more robust technique to map to UMLS CUI instead (Friedman, Shagina et al. 2004). Figure 4 is a sample MedLEE output representing one structured output extracted out of the frame templates described above. Alternatively XML and other data structures can be extracted from the system.

[β] In this notation "*" denotes cardinality = (0 or more) and can alternatively be replaced by 0, 1, >N or N. Conceptual Graphs are similar to Frames in that a [Class] (e.g., [Modifier]) is similar to a Frame and a (Relation) (e.g., (Bodyloc Mod)) is similar to a slot. Object of a relation can be another [Class].

```
problem:infarct
           bodyloc>> knee
           region>> right
                code>> UMLS:C0230431_structure of right knee
                code>> UMLS:C1279571_entire right knee
           certainty>> high certainty
           sectname>> report past history item
           status>> history
           code>> UMLS:C0021308_infarction

problem:vaso occlusive crisis
           certainty>> high certainty
           quantity>> [multiple,[idref,25]]
           sectname>> report past history item
           status>> need
           status>> past history
           code>> UMLS:C0750151_vaso occlusive crisis
```

Figure 4: Sample MedLEE output

MedLEE reports a recall of 0.83 (95% CI 0.79–0.87) for coding terms with a corresponding UMLS code and a recall of 0.84 (95% CI 0.81– 0.88) for extracting all terms. Extraction recall of the experts reported to range from 0.69 (95% CI 0.58–0.74) to 0.91 (0.95% CI 0.88– 0.95). The precision of the system is reported 89% (95% CI 0.87– 0.91), where the precision of the experts ranged from 0.61 (95% CI 0.51–0.71) to 0.89 (0.85–0.93).

MPLUS: A Probabilistic Medical Language Understanding System

MPlus is the latest from a family of statistical NLP tools developed at LDS Hospital in Salt Lake City, Utah (Christensen, Haug et al. 2002). Conceptualization of its predecessors SPRUS (Ranum 1989) and SymText (Koehler 1998) derive from common characteristics of radiology reports and has been used in the domain of radiology reports, admission diagnoses (Haug et al., 1997), radiology review (Friedman, Alderson et al. 1994) and syndromic detection (Olszewski 2003; Chapman, Dowling et al. 2004).

MPlus aims to take advantage of phrasal structure of the radiology reports to discover semantic patterns in text, as it tries to infer those patterns from lexical and contextual cues when

necessary. Furthermore the statistical processing of clinical text is believed to make the method resilient to the telegraphic and grammatically irregular text.

M+ represents the basic semantic knowledge of a medical domain in form of Bayesian Networks (BN). BNs provide an implicit framework for knowledge representation; as they have a character as both semantic networks and frames or slot-filler representations (Minsky 1975). BN are similar to a directed acyclic graph with nodes representing terms or concepts and edges representing relationships between them. Each node in a BN (e.g., "disease severity") is treated as a frame with an associated list of possible slot-filler values (e.g., ["severe", "moderate", "mild"]). The true value of a node is a probability that is assigned or inferred for each value by a probabilistic inference engine. A training process uses a set of select cases to calculate the Bayesian joint probabilities of all nodes. This is used to calculate the probability of each possible value of a node based on the probabilities of its neighboring nodes and by traversing through their edges.

The semantic properties of clinical content is represented as a network of nodes, with word-level and lower concept-level nodes providing input to higher concept-level BNs (Figure 5). Each BN instance is a template containing word and concept-level value assignments, and the interpretive concepts inferred from those assignments. The templates themselves are nested in a symbolic expression that facilitates composing multiple BN instances with arbitrary complexity.

A bottom-up parser with context free grammar (CFG) is used for a syntactic parse process. A new BN is instantiated for each phrase recognized by the parser and BN instances attached to sub-phrases within larger grammatical patterns are unified to create a set of completed BN

instances, as illustrated in figure 5. Each phrase recognized by the parser is assigned a probability, based on a weighted sum of the joint probabilities of its associated BN instances.

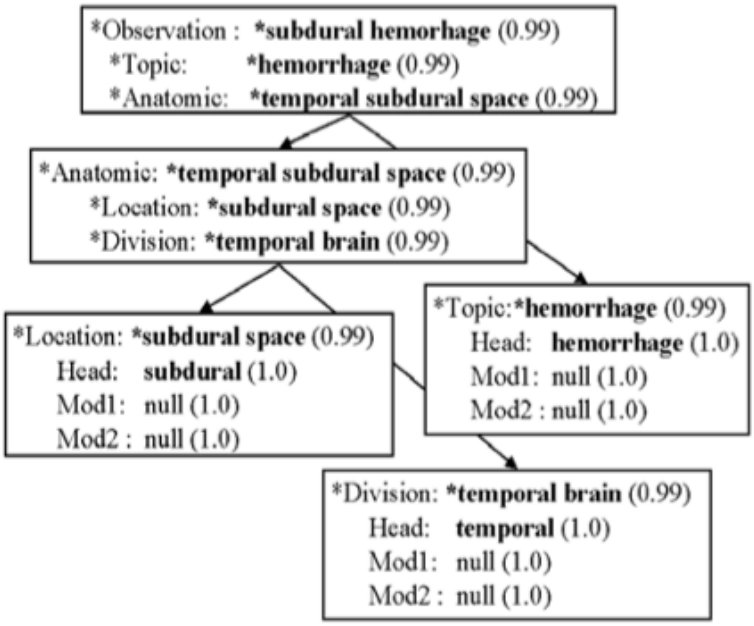


Figure 5: MPlus BNs applied to "temporal subdural hemorrhage".

A first-order language called the M+ Abstract Semantic Language (ASL) is implemented within M+ that treats BNs as a Class with the same name (e.g. "chest anatomy" and "chest radiology findings" class). The final interpretation of a phrase is an expression in the ASL, containing predicates to state the relations between BN instances, and the phrase they describe. For instance, the following ASL expression could interpret a phrase such as "opacity in the inferior segment of the left upper lobe, adjacent to the heart":

```
(and (head-of #phrase1 #find1)
    (located-at #find1 #anat1)
    (qualified-by #anat1 #anat2)
```

(adjacent-to #anat1 #anat3))

ASL expressions are based on an abstract semantic grammar (ASG) to formulate semantic rules for the interpretation of the BN instances and specific post parse inferences and augmentations such as unification and regularization. The ASG describes patterns of semantic relations among the BNs, and supports analysis and inference based on those patterns. It also permits rule-based control over the creation, instantiation, and use of the BNs, including defining pathways for information sharing among BNs using virtual evidence (Pearl 1988).

Training M+ in a domain involves gathering a representative corpus of training sentences for that domain. The training process begins with an initial set of interpreted "seed" phrases. The parser then applies to phrases similar to this set, and so semi-automatically traverses recursively into semantically contiguous areas within the space of corpus phrases.

M+ was evaluated for the extraction of American College of Radiology (ACR) utilization review codes from 600 Head CT-Scan reports (Fiszman, 2002) and based on eleven broad disease concepts. M+ reports a recall of 87% (CI, 78% to 95%), precision of 85% (CI, 77% to 94%) and a specificity of 98% (CI, 97% to 99).

Ong and Wang: Object-Oriented Approach to Medical Text Understanding

Kenneth Ong and Qiu He Wang from the Department of Electrical Engineering, National University of Singapore, argue that the major task of semantic analysis (understanding) of clinical content is to provide an appropriate mapping between the syntactic constituents of a parsed sentence or clause and the semantic representation of the sentence as a whole. They propose an object-oriented approach to semantic analysis of the medical text, composed of syntactic decomposition and semantic merging phases (Ong and Wang 1995).

A: The first component is a syntactic decomposer that generates an object tree from the syntactic constituents of a sentence (Figure 6-A) according to the phrase structure grammar (Chomsky, 1957). The decomposition starts from the whole sentence, and then its sub-components. Each syntactic constituent corresponds to an object and the decomposition process is driven by manipulation within the objects (Figure 6-B). An object creates sub-objects for its syntactic components, then passes control to them.

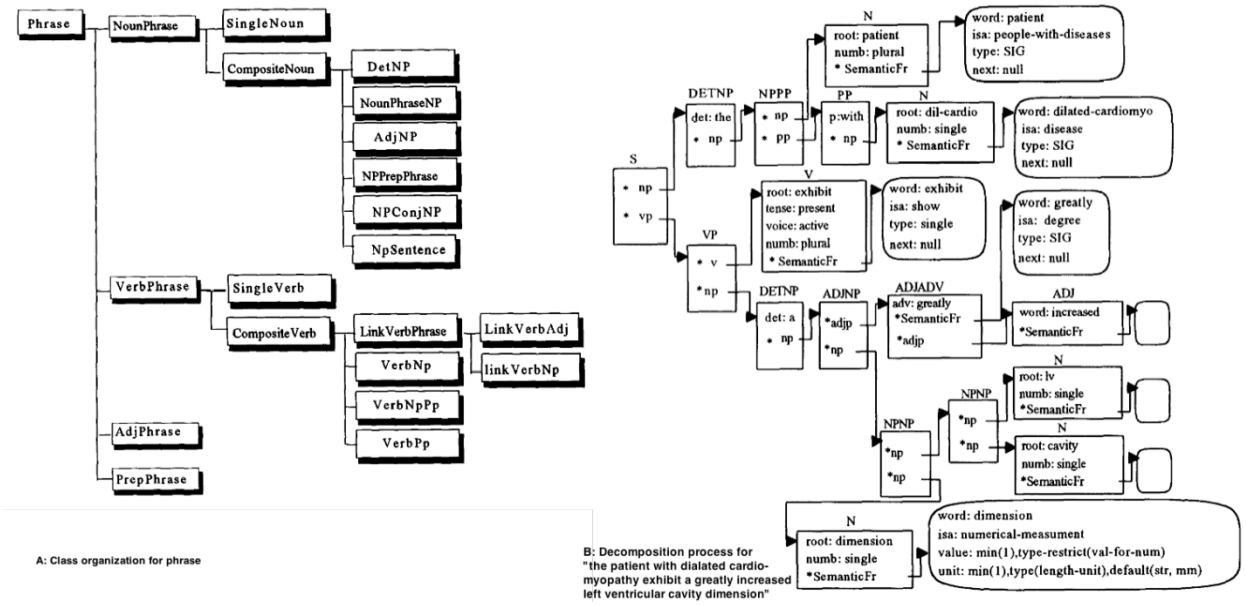


Figure 6: A: Class hierarchy representing a phrase; B: Syntactic constituents of a phrase and its decomposition process

A specialized frame language is designed for this project to represent the semantic knowledge for a single lexical item, a phrase, and a sentence. The semantic representation of a sentence is implemented by mapping from syntactic constituents to their corresponding semantic frame (SF) and merging the semantic frames of its constituents. The semantic merging is also able to unify and regularize sentences with different syntactic structures but the same meaning, by unifying them into the same internal representation.

As objects are created according to the syntactic constituents, corresponding semantic frames are attached to them. However, unlike the MPlus system the relationship between the syntactic constituents and semantic frames are not 1:1, because a word can have many senses. The semantic frames (SFs) for individual lexical items (e.g., a single word) are loaded from a semantic knowledgebase and are attached to their objects. The semantic knowledgebase encodes semantic properties associated with lexical items and the merge constraints between them.

A machine-readable version of the Longman Dictionary of Contemporary English (LDOCE) is being used as the base dictionary for domain independent words. The semantic knowledgebase also contains a domain dependent dictionary containing domain concepts and a generalized semantic grammar that instead of creating inference rules dealing with each word individually, expresses semantic merging rules that act on the knowledgebase.

B: The second component is a Semantic Merger that traverses the object tree by first attempting to merge the semantic frames for words, forming a semantic frame for a phrase and attaching it to the object of that phrase, forming a partial semantic description for a phrase, and then recursively merging the semantic frames for phrases until a semantic frame for the sentence is produced. Semantic frames of merged objects are deleted. The final result is a semantic frame net that expresses the meaning of a whole sentence.

Evaluation: The ECG reports used to make the following 11 diagnoses in are used to test the system's processing capabilities:

WPP = ventricular pre-excitation

LBBB = lefr bundle branch block

RBBB = right bundle branch block

LAntHem = left anterior hemiblock

LPosHem = left posterior hemiblock

RVH = right ventricular hyperrrophy

LAH = left atria1 hypertrophy

RAH-right aria1 hypertrophy

LVH = left ventricular hypertrophy

RBBB with L Ant Hem = right bundle branch block with left anterior hemiblock

RBBB with L Pos Hem = right bundle branch block with left posterior hemiblock

The sentences that the system can interpret account for 85% of the whole.

Taira et al, 2001: Automatic Structuring of Radiology Free-Text Reports

Taira et al describe design of a statistical NLP method under development that captures important aspects of a radiology free-text document (e.g., the existence, properties, location, and diagnostic interpretation of findings) as a formal information model that can be interpreted by a computer program (Taira, Soderland et al. 2001). An Extensible Markup Language (i.e., XML) is being used to provide the syntactic standard for representing and distributing these structured reports within a clinical environment.

The system is equipped with a hand-crafted knowledgebase (lexicon) that captures syntactic and semantic knowledge necessary to support the NLP process in the domain of Thoracic Radiology

and Neuro-Radiology. The lexicon contains domain specific terms, symbols and their semantic properties. A lexical analyzer takes a single sentence as input and associates each word or phrase in the sentence with a corresponding lexical item from the lexicon. This labeling process also identifies and annotates punctuation, dates, numeric measurements, special symbols, proper nouns, prefixes and suffixes. Our lexicons were developed manually.

The annotated sentence is used by a parser algorithm that generates a dependency diagram for the sentence, connecting words in a sentence to each other with unnamed arcs (Figure 7). An arc from word A to word B does not have any semantic entailment at this point other than "word A is associated with word B". The parser algorithm relies on two sets of probability distributions based on the "affinity" between two words and the "valence" preferences of each word in the sentence. The dependency diagram that globally maximizes these probabilities among all possible interpretations will be selected for future processes.

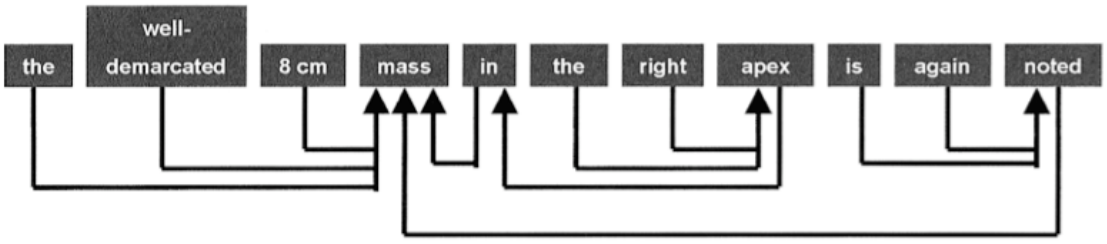
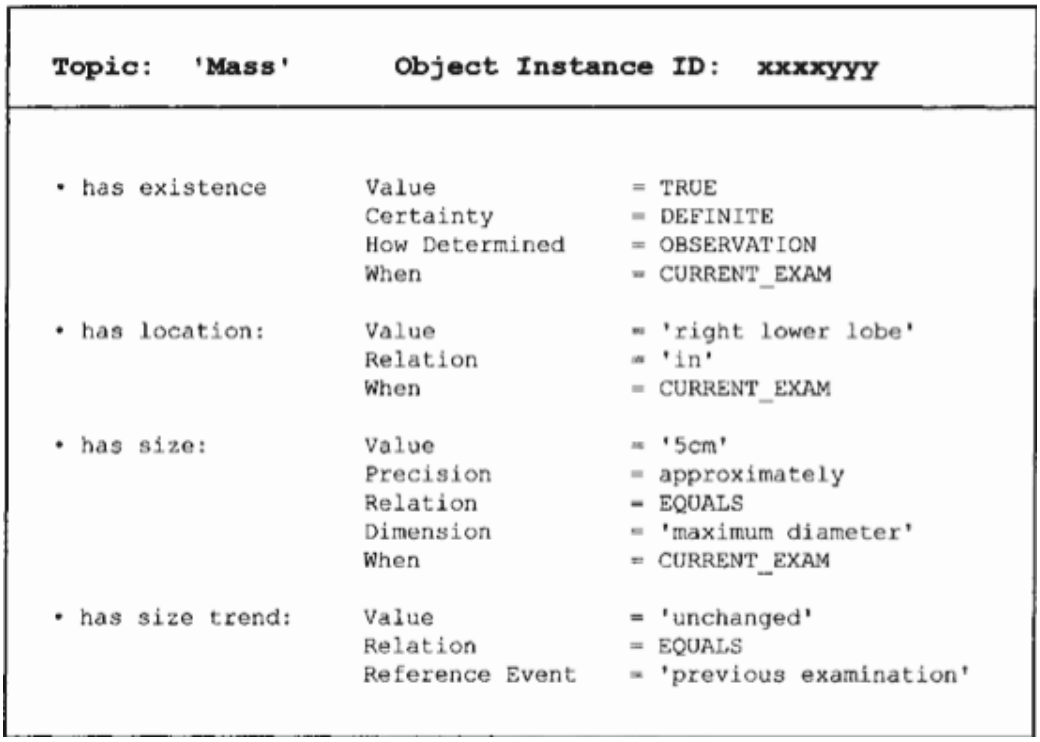


Figure 7: Dependency diagram showing relations between words.

Word affinity and valence probabilities were estimated by collecting a large sample of documents from the domain of thoracic radiology and manually creating a dependency diagram for each sentence as the ideal output of a parser algorithm, along with a valence table for each world in the sentence.



Topic: 'Mass' Object Instance ID: xxxxyyy

• has existence	Value	= TRUE
	Certainty	= DEFINITE
	How Determined	= OBSERVATION
	When	= CURRENT_EXAM
• has location:	Value	= 'right lower lobe'
	Relation	= 'in'
	When	= CURRENT_EXAM
• has size:	Value	= '5cm'
	Precision	= approximately
	Relation	= EQUALS
	Dimension	= 'maximum diameter'
	When	= CURRENT_EXAM
• has size trend:	Value	= 'unchanged'
	Relation	= EQUALS
	Reference Event	= 'previous examination'

Figure 8: Output knowledge frame for the sample sentence[ε]

A semantic interpreter maps the arcs of the dependency diagram to logical relationships indicating conceptual dependency between pairs of words that make medical sense. The logical relations also serve to normalize alternate ways of expressing the same concept in free text. The semantic interpreter algorithm uses a body of symbolic rules that are learned automatically from a set of hand-tagged training examples. Human experts first generate a target logical relations diagram in a training set using a specialized user interface. Then the system attempts to build generalizable rules by comparing the parse diagram from the training sentence with its target logical relations.

The semantic interpreter can be optionally augmented by method that constructs a maximum entropy classifier for each logical relation(Spyns 1996; Taira, Soderland et al. 2001). A frame

[ε] Frame represents "A mass is seen in the right lower lobe that measures 5 cm in maximum diameter and is unchanged from the previous examination". Terms from the lexicon are shown in capital letters, and values as expressed in the input text are enclosed in single quotation marks.

constructor algorithm uses all logical relations from a sentence to populate appropriate frames. Frames are hand-crafted and represent knowledge about a specific topic (e.g., “mass”), together with descriptions of select properties. There are three classes of topics: (a) abnormal findings, (b) anatomy, and (c) medical procedures and their corresponding frames (Figure 8).

The abnormal findings frame has 11 types of properties: existence, location, quantity, size, severity, trend, normalcy, external architecture, interpretation, association, and “other.” Each property may have its own sub-frame to represent relevant context modifiers such as time, evidence, certainty, degree, and dimension. The anatomy topic includes sub-frames for normalcy, subparts, direction, and distribution modifiers. The medical procedure topic class includes sub-frames for reason for procedure, technical description, and anatomic site.

Results of a preliminary evaluation document a recall of 87% and precision of 88% for the Parser algorithm (N= 4,314 possible dependency arcs) and a recall of 79% and precision of 87%. (N= 4,300 possible semantic relations).

Medsyndikate: Extraction of medical information from findings reports

SYNDIKATE (Hahn, Romacker et al. 2000; Hahn, Romacker et al. 2002) core technology and its customization to medical language understanding, MEDSYNDIKATE aims to extract conceptually deeper and inferentially richer forms of relational information from medical reports compared to the state-of-the art extraction systems. MEDSYNDIKATE presents a semantic approach to NLP in which the depth of text understanding is constrained only by the boundaries and scope of available knowledge sources, and the domain ontology rather than limitations inherent from knowledge representation framework used to represent semantic knowledge or restrictions due to pre-coordination of target templates. As invalid knowledge bases are likely to

emerge with purely sentence-oriented analyses, SYNDIKATE is also adapted to deal properly with various forms of anaphoric reference relations spanning several sentences (Hahn, Romacker et al. 1999).

The SYNDIKATE text understanding process at the basic sentence level takes three types of knowledge into account (Hahn, Romacker et al. 2002):

1: Grammatical knowledge or the Lexicon covers the lexical system and the syntax of the underlying natural language (German) for syntactic analysis and is represented in a fully lexicalized dependency grammar. Lexical entries contain declarative knowledge in terms of morpho-syntactic features (e.g., gender, case, mode, tense marking), word order constraints, and valence specifications. A Generic Lexicon contains domain-independent entries (such as move, with, or month), while domain-specific extensions (such as adenocarcinoma, gastric mucosa , etc.) are encapsulated in specialized lexicons.

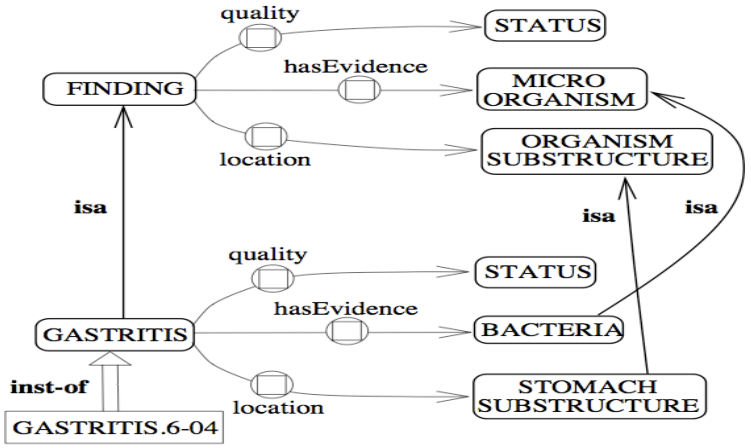


Figure 9: Fragment of the Conceptual Knowledge (domain ontology)

2: Conceptual (domain) knowledge which incorporates the general background knowledge, as well as specialized knowledge about the particular domain of the texts (anatomy and pathology,

for MEDSYNDIKATE) and is expressed in a KL-ONE-like representation language(Hahn, Romacker et al. 2002).

The knowledge representation language supports the definition of complex concept descriptions by means of conceptual roles and role filler constraints, and enables primitive taxonomic reasoning (following explicit links), as well as computation of subsumption relations between complex conceptual descriptions (Figure 9). The language also makes distinction between a concept class and instances that represent concrete real-world entities (Figure 9). Most lexical items in the Lexicon are directly associated with one or, in case of polysemy, several concept types.

3: Semantic schemata mediate between syntactic and conceptual (domain) knowledge and includes a set of schemata that captures minimal semantically interpretable subgraphs of the sentence dependency graph under incremental construction. Semantic knowledge enables the algorithm to determine the proper relations between instances of concept classes based on the subgraphs of sentence.

Figure 10 summarizes the overall architecture of SYNDIKATE (Hahn, Romacker et al. 2000). The general task of this system consists of mapping each incoming text, Ti, into a corresponding knowledge base, TKBi, which contains a formal representation of (portions of) Ti's content. Each knowledgebase can then be accessed, e.g. for inferentially supported question answering, but can also be used for information retrieval and even text summarization purposes. SYNDIKATE uses a discourse memory mechanism to prevent from possible inadequate knowledge representation issues that may emerge from sentence centric analysis of the text. An

array of open discourse entities are maintained in the memory as the text proceeds and are used to track and establish relationships between entities across sentences.

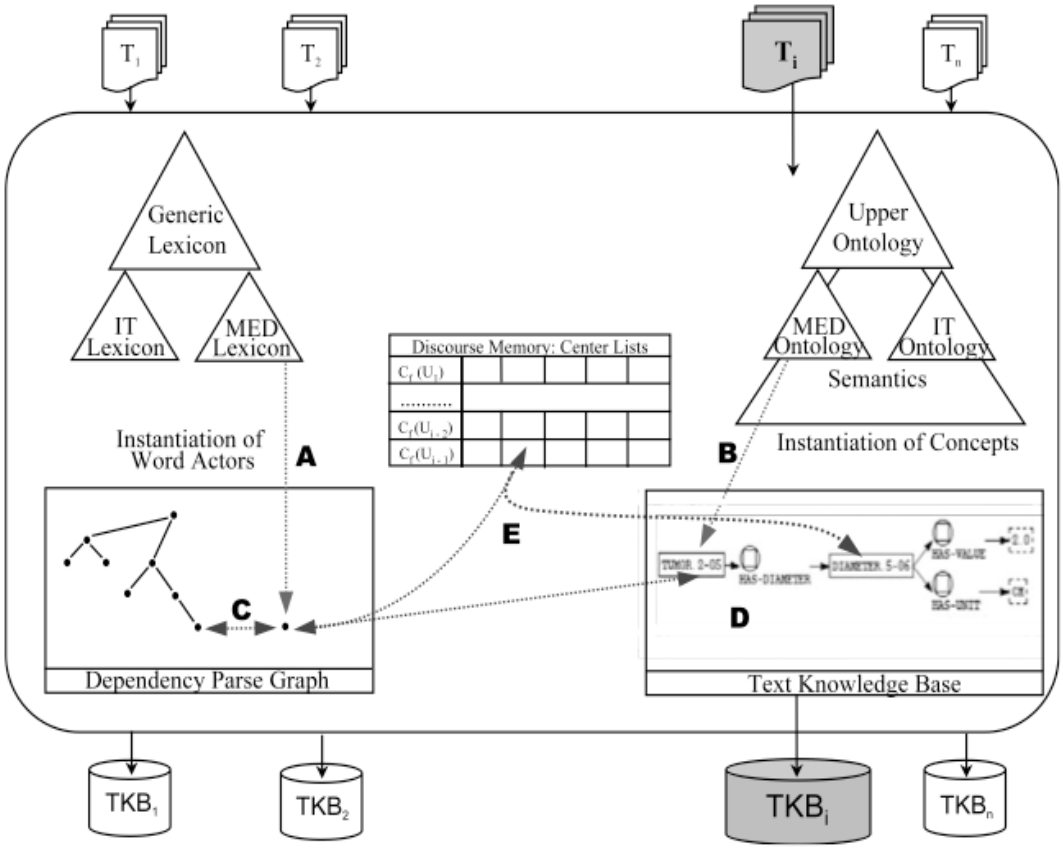


Figure 10: MEDSYNDIKATE system architecture

Whenever a lexical item is read from the textual input and it is identified as an entry in the lexicon, a specialized object called word actor is instantiated. Word actors combine lexicalized grammar knowledge and positional information from the text and interact with each other to produce a dependency trees labeled with functional relations between lexical terms. A semantic interpretation algorithm takes the dependency graph as input and applies rules incorporated in the semantic knowledge, and taxonomic and subsumption based classification based on the domain knowledge to produce a formal and consistent semantic graph. That is, the linguistic expressions extracted from syntactic analysis of the text are mapped to a semantic graph -that is a

canonical representation of the underlying domain knowledge- to form the final result of the text understanding process.

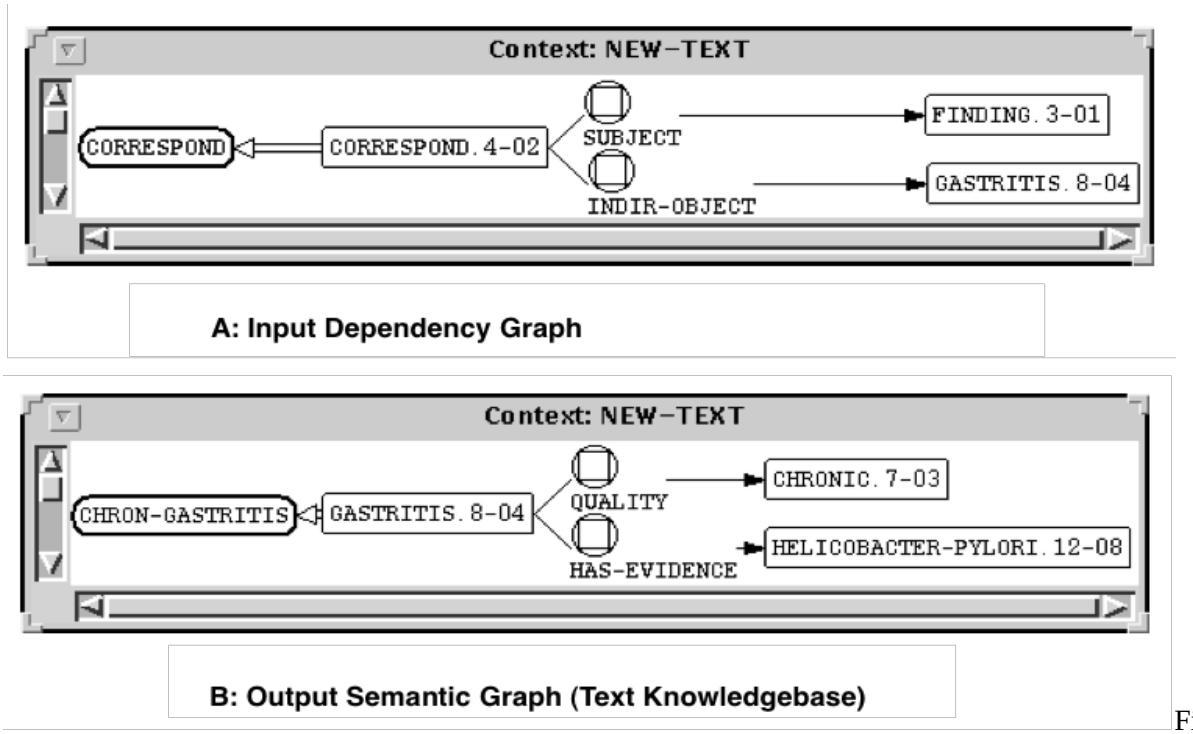


Figure 11: A: the Input and B: the output of the semantic interpreter

The semantic graph represents the semantics of the clinical content in terms of instances of various concepts from the domain ontology and relationships between them (Figure 11) and continuous processing of sentences one after another augments the text knowledge base incrementally.

Discussion:

Systems described above are exemplary natural language processing systems that aim to transform unconstrained clinical text into a structured representation that can be used by computer programs for querying and information retrieval. They all use various forms of knowledge representation to describe syntactic, semantic and domain knowledge that is necessary for an automated algorithm to successfully parse and interpret lexical constructs and

linguistic expressions and transform them into a desirable data structure. However there are important differences between conceptualization of these systems, richness and expressivity of the knowledgebases available to them, and richness, expressivity, and reusability of the output they generate. Table 1 outlines some of the similarities and differences between the systems outlined here. It is important to mention that not all systems introduced here claim or meet criterion for a true 'text understanding' system. That is, not all of these systems produce a formal and explicit output that support automated reasoning, contextualization and reuse by external systems. Most systems produce a data structure that makes querying and information retrieval possible, but the relationship between elements of the data structure and their meaning are only implied by the surface features of the data structure and its schemata. This limits reusability of the output for secondary purposes and may not support dynamic information sharing across a network of collaborative agents.

A: Syntactic Analysis

LSP system is an example of systems that rely extensively on syntactic parsing of clinical content using the surface features of linguistic expressions and a generic but extensible grammar. The clinical extension of LSP system is based on the assumption that there are a finite set of stereotype linguistic patterns that represents majority of sentences used in medicine and its disciplines. Hence the LSP system views clinical text as a set of linguistic patterns in terms of the sentence types, and specialized word usage that can be effectively parsed by an extended English language interpreter. However, this method of parsing relies on clinical content with sentences that are grammatically correct and complete, and whose structure, grammar and rules are known to the system. The method is fragile in the face of variability of natural language expressions, incomplete sentences, or irregularities in grammar. Extending the grammar of the language and

its patterns to a new sublanguage or to accommodate a variation in linguistic expressions are difficult and hard to maintain(Grishman and Kitteredge 1986; Hahn and Romacker 1997).

The MedLEE, MPLUS and the system described by “Ong and Wang” also use a syntactic parsing algorithm that is sensitive to the grammatical completeness and structure of the sentences but resilient in the face of variations and changes in linguistic expressions used in text. These methods represent the structure of a string according to some formal grammar that describe which of the possible sequences of symbols (strings) in a language constitute valid words or statements in that language, but it does not describe their semantics (i.e. what they mean). Context free grammars are basically based on the Phrase Structure Grammars, more specifically described through Chomsky Hierarchies (Chomsky 1956), and are most often used because parser for them can be efficiently implemented, although much more powerful and unrestricted grammars exist (Harris 1968; Grune and Jacobs 1990). Most modern parsers are at least partly statistical; that is, they rely on a corpus of training data, which has already been annotated by human experts. This approach provides the system with complementary information about the frequency with which various constructions occur in specific contexts (Probabilistic Context Free Grammars). Many different formal grammars exist for context free parsing of natural language, creating different parse trees from the same linguistic expression. Furthermore, ambiguous or complex text may produce multiple valid parse trees by the same grammar. That is, the same text can be represented by many different, but valid parse trees (ambiguity). None of the cited literature provides with a rational on selection of one parser over others, insight about consequences of using a different CFG for syntactic analysis, or how they deal with ambiguities during parsing. But it is conceivable that a parser has been selected based on a benchmarking over a pool of sample clinical text from their intended domain. This may result in unexpected

performance result if quality and syntactic features of the clinical text is different from the sample, or the method is being used in a different domain. Furthermore, syntactic parsers are generally sensitive to the grammatical and syntactic structure of the text. None of the cited literature indicates comparative performance statistics using data with qualitatively different syntax and grammatical structure.

Another popular strategy for syntactic analysis is dependency grammar parsing used by MedsynDikate system and Taira et al. **Dependency grammar** (DG) is a class of syntactic theories which builds relations between pairs of words rather than constructing constituents in a tree-like hierarchy and it lacks phrasal nodes typical of phrasal structure grammars. The structure in dependency diagrams is determined by the relation between a word (a head) and its dependents (direction and distance). For example, in an Subject-Verb-Object language like English, the verb would look left to form a subject link, and right to form an object link, whereas in an Subject-Object-Verb language like Persian, the verb would look left to form an object link, and a more distant left to form a subject link.

Dependency grammars are comparable to phrase structure grammar in terms of their sensitivity to grammar and syntactic structure of the text. That is, there is an implied assumption by their use that clinical content presented to the NLP system will have a fairly complete and syntactically and grammatically correct structure. Furthermore, they may also produce multiple ambiguous diagrams, and should be augmented by some statistical methods and training processes to reduce the size of the result-set produced.

In conclusion, one may argue that empirical evidence indicating a fit and rational to use certain syntactic analysis framework against others by medical and clinical language processing tools

are scarce and generally absent from the discussion of published literature. These frameworks are generally complex processes that depend on the grammatical and syntactic features of the underlying text and language to perform and are difficult to develop, extend and customize for specific characteristics of a new domain.

B. Knowledge Representation

All NLP systems need to have access to some form of syntactic, semantic and domain knowledge in order to successfully parse text, identify relevant terms and construct a meaningful output. However different systems use different combination of methods and frameworks to capture, represent and make them available to the NLP processes (Do Amaral and Satomura 1995).

The content (richness and scope) and the representation framework used to construct a knowledgebase are both important in evaluation of NLP tools.

Content of Knowledgebase: portability, generalizability and utility of most NLP tools are constrained by the coverage and completeness of their knowledgebase. Most tools are designed for a specific type of clinical content in a small domain (e.g., ECG reports, Head CT Scans, Chest X-Ray reports) and have highly specialized knowledgebases limited to the intended document types or domain and cannot be readily applied to other document types, regardless of the NLP processes used.

Representation framework: content of a knowledgebase becomes as accessible and useful to computer programs as their information representation framework allows. That is, the representation frameworks used to capture and represent information and knowledge constrains scalability, extensibility, availability and utility of that information and knowledge for computer

programs. Use of proprietary data structures that restrict interpretation and computation, retrieval, or extension of the knowledgebase may prevent from its reusability, portability and interoperability.

The content and representation of the knowledgebase compound each other in their influence on the overall utility of the knowledgebase. For example a representation framework may be expressive enough to support computer reasoning and efficient retrieval but the content of the knowledgebase may be incomplete and lack appropriate information to support it, or a complete and comprehensive knowledgebase may be represented through a representation framework that cannot support deductions and reasoning beyond what is intended for a specific NLP process. This can directly impact the richness and reusability of the output as well as portability and application of the NLP tool.

In this section content and representation of terminological, semantic and domain knowledge in the exemplary text understanding systems will be compared:

B1. Syntactic Knowledge: The syntactic knowledge is mainly used to provide a dictionary of valid terms in a domain (lexicon), morphological and syntactic rules of the underlying language (such as valence and inflexions), and a grammar that sanctions or constrains allowable combinations of terms in a domain. The lexicon may also contain relationships such as synonymy, hyponymy (i.e., narrower), hypernymy (i.e., broader), Polysemy (i.e., related terms), and meronymy (i.e., part of term) [see Apendix A] between terms (terminological knowledge) to be used for disambiguation and reducing the variability (normalization) of the output.

Content of Lexicon: scalability, generalizability and utility of most NLP tools are constrained by the coverage and completeness of their lexicon. Systems like LSP, MPLUS, Taira et al, and Ong

and Wang are designed for a specific type of clinical content in a small domain (e.g., Asthma, Head CT Scans, Thoracic Radiology, and ECG reports retrospectively) and have a highly specialized lexicon, constructed manually and limited to their intended document types and domains. As a result their lexicon and consecutively the system itself cannot be readily applied to other document types, unless the lexicon is extended to cover the new applications. This can be an intensive and rate limiting manual process and depends on the availability of specialized human expertise, and tools to support it. Systems like MedLEE and MedsynDikate that use standards based knowledge sources such as UMLS as source of terminological knowledge are better prepared for their NLP process to be ported and applied in a domain other than what originally intended.

Representation framework: All cited literature about conceptualization and design of biomedical language processing systems are silent about the structure and representation of their lexicon and terminological knowledgebase. It is conceivable that not being concerned with reusability, extension and sharing of this knowledge source with external systems or collaborators, each tool uses a proprietary data structure that restricts access and extension of the terminological knowledgebase. This may ultimately prohibit from the reusability of the output, and portability of the NLP method.

Ideally, a standards based and extensible knowledge representation framework specifically designed for representation of terminological knowledge sources such as taxonomies, thesauruses, and nomenclatures (e,g, W3C Simple Knowledge Organization System [see Apendix A]) could be used to represent lexicons and terminological knowledge within the NLP applications. This may provide a formal, rich and expressive framework that enables automated

contextualization and extension of the terminological knowledgebases used by NLP methods and its portability and reuse in new and novel use cases.

B2. Semantic Knowledge: Semantic knowledge defines meaning of lexical constituents of text and its syntactic components by mapping them to unique concepts and sensible relationships between them. In most systems semantic knowledge includes a set of explicit schemata that captures generalized semantically interpretable relationships between concepts, and semantic interpretation of template linguistic patterns observable or frequently used in the clinical content. That is, the semantic knowledge enables the algorithm to determine the proper relations between terms within the text, and transforming (mapping) them to desirable output formats. Generally, the more rich and complete the semantic knowledge, the more rich and complete the output of NLP method will be, as it will enable to algorithm to identify and extract more of identifiable patterns and relationships from the text.

LSP system uses a series of Information Formats to represent the semantic knowledge within the system. Custom data structures are pre-coordinated to represent the syntactic elements of text and their relationships with each other. However, the semantics of this relationship are only implicit and meaningful only to human being. The information formats in LSP system act similar to XML tags, that is, a syntactic convention to contain data without any semantic implication. As a consequence the Information Formats cannot support computer reasoning and retrieval and only mediate between the syntactic analysis and the database that is used to contain the output for querying.

In MedLEE and the systems introduce by Ong and Wang and partly by Taira et al, the semantic knowledge is represented through templates constructed using a formal frame language [see

Appendix]. As frames provide with some formal semantic that can be used for classification (based on inheritance and frame matching) and information retrieval, these systems provide a richer and more expressive representation framework to capture the semantic knowledge in the NLP system. However, because the semantics of a template frame, relationships between a frame and its slots, and slot values are not explicit and are implicitly defined within the boundaries of the NLP application itself, other applications cannot use them for computer reasoning, contextualization and repurposing of the NLP output. That is, the semantic knowledge is trapped within the NLP system and is not accessible to support reasoning, extension and reuse beyond what is anticipated and pre-coordinated by the NLP application itself.

MPLUS system uses Bayesian Networks and a custom made abstract grammar and language (ASG and ALG) that captures and represents the semantic knowledge and rules of transformation between syntactic components and the output. Although Bayesian Networks support probabilistic reasoning and can be conceptually seen as a knowledge representation languages with similar features to conceptual graphs and frames, but they are not generally known or used as robust information representation frameworks for information sharing and retrieval. Their construction, maintenance and extension require intensive knowledge acquisition and training processes and similar to other systems, semantic knowledge within these frameworks are generally not reusable, extensible or accessible by other systems.

In conclusion, systems like MedLEE that use frames and formal templates provide semantic properties of clinical content as much as it is enough to provide an information model representing linguistic expressions and lexical patterns found in clinical content, but do not provide a knowledgebase of biomedical and clinical domain that formally explicate semantics of the concepts for external systems that are interested in reusing, extending and contextualizing

their output. The boundaries of inference and deductions that they can support is defined by limited templates that contain only minimal factoids about particular, a priori chosen entities (diseases, findings, viruses, severity degrees etc.) defined by the intended document types and domain. These knowledge sources are considered to be entirely static. Accordingly, new templates must be supplied or existing ones must be updated manually when a new application domain or document type is introduced. Furthermore, if the templates are modified or an enhanced set of templates are introduced, the NLP process has to be repeated again and for the entire document collection as the system will not be able to compute the correspondence between the previous and current models automatically, as the emplates provide either no (e.g, Taira et al system) or severely constrained inference capabilities (Ong and Wang, and MedLEE) to reason about their fillers.

Comprehensive domain knowledge that defines domain concepts generically and independent of the NLP system along with the expressiveness and inferential power that comes with formal knowledge representation systems (e.g., Description Logic) can enable automated contextualization, extension and reusability of NLP tools.

B3. Domain knowledge: Domain knowledge is frequently used to provide the background knowledge about domain concepts, and to define, constrain or sanction generalized relationships between them and in a domain of discourse. More importantly, if represented with expressive knowledge representation frameworks such as Web Ontology Language (OWL), domain knowledge (Ontology) enables inferences and reasoning by computer programs that is critical for contextualization, extension and reuse of NLP output by external systems. However such ontologies with broad consensus and coverage are not currently available in mainstream biomedical and clinical informatics communities. Furthermore, knowledge sources such as

UMLS-KS from National Library of Medicine cannot be used directly as source of biomedical and clinical knowledge for NLP systems, because it is characterized by inconsistencies, circular definitions, insufficient specification depth and granularity mismatches, gaps, and more importantly lack of a solid knowledge representation framework that could support inferences and reasoning. However, enabling just-in-time use of such high-volume knowledge repositories within NLP systems as a source of domain knowledge and on-demand has been the center of attention for knowledge engineers and NLP research community.

MedsynDikate is the only system today that claims use of domain knowledge to further NLP process beyond superficial characteristics of lexical expressions and semantic templates available to the system. MedsynDikate also provides with a machine learning method to obtain domain knowledge from biomedical literature or consistently extract it from UMLS. However, using KL-one as the knowledge representation framework by MedsynDikate although supports a rich set of constructs for reasoning and classification tasks, but is not an appropriate for information sharing and collaboration in a distributed environment and cannot be easily extended and contextualized by third party systems.

C: Output Representation Framework

Frequently, the output generated by NLP systems is a reflection of its intended use in a certain application or domain and an indicator of how the overall NLP system has been conceptualized and implemented. The NLP output is only as rich, granular, and clearly and precisely interpretable as its information representation framework allows. That is, even with a comprehensive NLP processing and output availability and retrieval of information from NLP output may be hindered if the output representation framework does not readily support computation. That is, output of NLP systems should be represented using an extensible,

consistent and easily understandable framework that enables access, and retrieval of information by other systems without requiring proprietary tools or intensive reprogramming.

Use of custom data structures such as XML schemas, specialized array lists, object trees, information formats with data type descriptors, or relational database schemas (as being cited by all NLP tools in this article but MedsynDikate) are not an answer to the problem of interoperability and reuse, as they lack the sufficient semantics for an external interpreter to automatically and appropriately interpret the meaning of such schemata and structure. As a result intensive human knowledge and intervention, reprogramming and scripting will be required to consume outputs of an NLP tool for a secondary use.

These data structures are not able to assist computer programs to identify or appropriately deal with implicit, ambiguous and vague information within the output. Operating only on syntactic and schematic level, these information models are not extensible enough to accommodate change or new detail without creating inconsistency and incompatibility with previous information. That is, the entire collection needs to be reprocessed in the face of a minimal change in the information model itself or the granularity and level of details in existing elements. Furthermore and most importantly, none of these data structures have sufficient constructs with known and consistent semantics to support reasoning and classification, another “must have” if the NLP is to be reused and contextualized by external systems.

On the other hand use of standards based information representation frameworks with some formal semantics (such as Resource Definition Framework- RDF) may facilitate computer reasoning, contextualization, information sharing and interoperability across systems.

MedsynDikate is the only system that provides with a formal output representation that can support deep and knowledge level queries with inferences and reasoning. However, the representation formalism used in this system (KL-One) is not appropriate for information sharing, and collaborative extension, contextualization and reuse in a distributed environment.

D: Knowledge and Context Representation.

Frequently NLP systems draw inferences, and use reasoning based on some domain knowledge, assumptions, defaults and rules, in order to deal with complex real world use-cases and requirements (Friedman and Hripcsak 1998). For example, the term '*opacity'* in an X-ray report may be classified as '*neoplasm'* in an NLP system if used along with certain modifiers such as 'shape: *ill-defined'* and 'distribution: *cluster'*. This knowledge is frequently omitted from the output representation based on the assumption that the systems utilizing the output share the same context and commit to the same heuristics, assumptions and presumptions. Absence of this information prevents from secondary use of NLP output or adoption of these systems in new environments as the output is not provable and cannot be automatically validated or disambiguated. An optimal output representation should be self-descriptive in a way that enables information systems to access the relevant context, and the semantics of the information represented in the output, in a way that conclusions can be traced back to the associated evidence in the text and the logic and rules of inference in the knowledge base.

MedsynDikate is the only system in our list that constructs the NLP output as instantiation of a formal model that explicitly defines all logic, relationships and rules of inference in the system in a way that its conclusions are traceable. However, the representation formalism used in this system (KL-One) is not appropriate for information sharing, and collaborative extension, contextualization and reuse by external systems.

E: Encoding to Standard Vocabularies:

Mature and comprehensive medical vocabularies exist today (SNOMED-CT, UMLS, etc.) that can be used to disambiguate concepts, reduce the variability of the NLP output, and as a result greatly enhance its reusability and interoperability. However, there is a difference between using a controlled vocabulary as the lexicon or source of terminological knowledge and encoding NLP output to a controlled vocabulary. For example, MedLEE system uses UMLS as both Lexicon and Also uses a customized method of mapping frame concepts to a UMLS concept unique identifier (CUI). Whereas the MedsynDikate system uses UMLS as source of biomedical knowledge but does not encode its output to UMLS concept unique identifiers.

F: Evaluation

A reliable evaluation process should result in measures that may objectively and accurately predict the behavior of a text understanding system in a realistic clinical environment or when executing a specific task. However, established criteria for performance evaluation with generalizable guidelines that can be followed by designers and users of text understanding systems do not yet exist. The lack of standard evaluation methods (based on consensus or de-facto standards) with objective measures is detrimental because it may lead to deceptive results, false expectations, and increased clinical error. There are several reasons that contribute to the overall lack of standard evaluation methodologies for text understanding systems in medicine:

Text understanding systems are still in early stages of conceptualization, design and implementation with primary focus being the implementation rather than application and evaluation.

Proper evaluation of a text understanding system is a very difficult task to define and execute. Although all recall and precision rates measured for extraction and encoding also apply to text understanding systems as measured for other NLP systems, but measurement of 'understanding' is an ambiguous and controversial issue and should be defined and agreed upon first and before appropriate evaluation methods can be properly and accordingly conceptualized. One may define understanding in many different ways by focusing on various aspects of the technology such as: completeness of the knowledgebase, expressivity of knowledge representation framework and reasoning engine, consistency and tractability of resulting knowledgebase, recall and precision of the method in identifying relationships between concepts extracted from text, the ratio of extractions and encodings to inferences and deductions that system can support, or success and failure rate of the system in mapping surface linguistic expressions to some semantic schema.

There are no published guidelines for evaluating text-understanding systems in the biomedical literature. Although there is considerable literature and precedence on evaluation of NLP systems in general and in medical informatics (Anderson, Aydin et al. 1994; Grishman and Sundheim 1995; Friedman and Hripcsak 1998) articles that focus specifically on the evaluation of biomedical and clinical text understanding systems are absent.

With this introduction it is next to impossible to compare the performance of systems cited in this article as it relates to their ability to 'understand' text. LSP system, MPLUS, and Ong and Wang have not provided any detail about their evaluation philosophy, method, or measures used and have stopped short at presenting the recall or precision rate of their extraction and encoding performance (not directly related to understanding). MedLEE system is the only system with explicit description of the evaluation method, but again, has only focused on and reported only evaluation of the system's performance in extraction and encoding and not understanding. Taira

et el have reported preliminary results from an undisclosed evaluation methodology that focuses on accuracy and reliability of the system in identifying semantic relationships between concepts (rather than extraction or encoding). MedsynDikate has a rather different take on the evaluation of text understanding systems, by measuring success rate of parsing and mapping of linguistic expressions and morpho-syntactic features of the clinical text into some valid semantic representations in the knowledgebase.

Ch1apter Summary; A Gap Analysis:

With a review of the current state of the art it is evident that the conceptualization, design and implementation of current NLP systems constrains they application and use to a pre-coordinated document type, domain or task. It is not an easy and intuitive task to reuse output of current NLP systems in a novel context or to apply the method in a novel context without extensive customizations, modification and programming activities. The reasons behind the lack of portability of the methods and reusability of their output can be sought in the way these tools approach syntactic parsing of the clinical content, the content and the representation frameworks used to construct their terminological, semantic and domain knowledge, availability of domain knowledge that provides a formal definition for concepts incorporated in the output, conformance to standards based vocabularies, and an evaluation and reporting framework that objectively describes and measures performance criteria for the text understanding applications.

A: Syntactic Analysis:

Theoretical and experimental research in computational linguistics in the past few decades has produced many different frameworks for syntactic analysis of natural language. However,

empirical evidence on appropriateness of using one framework against others for parsing biomedical and clinical text does not exist. Current NLP literature stop short of explaining the rationale of using particular syntactic analysis methods and their impact on reliability and validity of the NLP output.

Most parsing frameworks share the assumption that “the syntactic and grammatical structure of the underlying language is complete and stable over time” (Sager 1986) . This assumption cannot be guaranteed in most clinical environments, and with unknown consequences on the performance of the NLP system. On the other hand, literature citing statistical (probabilistic) parsing methods and (theoretically) claiming resilience to incomplete or irregular linguistic expressions do not provide evaluation result indicating a consistent performance on qualitatively different clinical content.

B: Terminological Knowledge:

Handmade and customized lexicons that satisfy limited vocabulary needs of a small domain is prohibitive to extension and application of the tool to new context. Representation of other terminological relationships such as synonymy, hypo and hypernymy, etc is important for alignment, extension and application of the tool or reuse of the NLP output in novel scenarios. Ideally, the terminological knowledgebase should be a construct of a known standard-based vocabulary system or provide a consistent mapping to one. An ideal terminological knowledgebase should also support ad-hoc and multilingual translations and reuse of the NLP output.

C: Semantic Knowledge

Use of minimally modeled, pre-coordinated and static schema (frames) as template to represent important semantic relationships between components of text is limiting and prevents future extensions of the system to support different levels of granularity, details, and novel information models. An extensible and dynamic semantic model is a key to providing a rich and expressive output representation that can be reused ad-hoc and automatically by secondary systems.

D: Domain Knowledge and Context Representation

Availability and use of a rich knowledgebase that formally defines and provides background information about rules, logic and relationships between domain concepts is critical to ensure reusability, extension and contextualization of the NLP output but absent from most text understanding systems of today.

The domain knowledge should be made available to the secondary users of NLP output to support automatic contextualization and reuse. Although use of knowledge sources with broad consensus among clinical and biomedical research communities is desirable, such knowledgebases are not yet available for practical use by text understanding systems. Research is needed to engineer and evaluate machine learning techniques that can use existing sources of biomedical knowledge such as digital libraries or UMLS as a consistent and reliable source of domain knowledge for text understanding algorithms.

E: Knowledge Representation Framework

Knowledgebases of NLP algorithms (terminological, semantic and domain knowledge) are key to the interoperability, extension, and reuse of their output and portability of the method to new environments. However in most cases use of syntactic schemas or data structures with limited

support of computer reasoning and inferences to represent this information for the NLP system limits its utility and availability for secondary use by other computational algorithms aimed at contextualization and repurposing the NLP tool or its output.

Ideally an expressive and formal knowledge representation framework that supports reasoning and inferences should be used to represent all knowledge within the NLP system and make it available to external systems for reuse.

F: Output Representation

Output of most NLP systems (except MedsynDikate) is yet another textual representation but one that conforms to some predefined schema. For example, MedLEE system takes free text and transforms it into an XML output that is a purely syntactic representation. There is no link between elements of the XML output and the domain knowledge, semantic knowledge or terminologies that can further define them or support any form of disambiguation, computation, reasoning or interpretation.

Ideally output of NLP systems should maintain such linkages and incorporate meta-data and further information that can be used by secondary applications and systems to automatically interpret them. Furthermore, the NLP output should be represented using the same formal knowledge representation frameworks as the system knowledgebases in order to enable inferences and reasoning by automated computer programs.

The approach taken by MedsynDikate to present output of the NLP processes as instantiation of the same domain model used by the system is the most appropriate one, however the formalism used to represent the domain knowledge and the output is not appropriate for information sharing and collaborative reuse and contextualization of NLP output.

Information and knowledge representation frameworks such as RDF and OWL from the W3C Semantic Web technologies community of research are conceptualized to enable information sharing across boundaries of systems, applications and organizations, and in a distributed environment such as web. They aim to establish a framework for identical interpretation of informational resources by computers and human beings. This is a key prerequisite of a system with interoperability and reusability in mind, and seems to offer a more robust and appropriate framework to represent both system knowledge and system output for NLP applications in a way that NLP output becomes self descriptive for secondary applications and in light of the same body of knowledge that has been used by the system to construct the output.

Table 1. Comparative Summary of Cited Systems

Author	Primary Domain	Encoding	Domain Knowledge	Semantic Knowledge	Method	Recall	Precision
Taira, et al	Thoracic radiology	Manually developed lexicon	No	Implicit through Probability Distribution Tables and a Frame Language	Syntactic Parsing and Statistical Processing	87% (parser); 79% (semantic interpreter)	88% (parser); 87% (semantic interpreter)
MedsynDikate	Gastro-Intestinal diseases	Extractions from UMLS	KL-ONE Ontologies to represent Semantic and Domain Knowledge	LOOM ontologies to represent rules and semantic mapping schemata	Syntactic Parsing, Semantic Processing and Rules reasoning	93% (Genitives); 80% (Auxiliaries and modals); 85% (Prepositional phrases)	93% (Genitives); 84% (Auxiliaries and modals); 81% (Prepositional phrases)
Ong and Wang	ECG reprots	Manually developed lexicon	No	Semantic Frames (Frame Language?)	Syntactic Parsing and Object Oriented Processing	85%?	?
MPLUS	Radiology Reports (Head CT Scan)	Manually developed lexicon	No	Implicit through Bayesian Networks and ASL and ASG for formal representation of grammar and rules	Syntactic Parsing, Probabilistic (Statistical) processing augmented by Rules reasoning	98% (CI 97-99%)	85% (CI 77-94%)
MedLee	Radiology	UMLS	No	Frame Language	Syntactic Parsing, Semantic Processing and Rules Reasoning	70% - 85% (after training)	87%
LSP	Asthma Control Quality Assurance	LSP English medical dictionary	No	Implied through Information-Formats	Syntactic Parsing and Rules Reasoning	82.1 %	82.5%

CHAPTER 3: A MINIMAL SYNTACTIC, SEMANTIC APPROACH TO BIOMEDICAL LANGUAGE UNDERSRTANDING AND EXTRACTION (BLUE-TEXT).

Motivation and History

The Center for Biosecurity and Public Health Informatics Research has been interested in availability and utilization of robust clinical text understanding systems because critical clinical information represented as unconstrained text in electronic health records is known to be useful for timely and accurate case finding and reporting of incidents with public health importance(Mirhaji, Zhang et al. 2003; Mirhaji, Lillibridge et al. 2004; Mirhaji, Richesson et al. 2004; Mirhaji, Richesson et al. 2004; Mirhaji, Zhang et al. 2005). The center was interested in being able to use as much of clinical content available, without regards to its schema and structure, to build classification engines and data mining platforms for public health surveillance and preparedness.

Furthermore, recent initiatives advocating for translational research call for generation of technologies that can integrate unstructured clinical data with structured data while contextualizing and repurposing it to support multidisciplinary research in collaborative environments as envisioned by the CTSA program (Mirhaji, Zhu et al. 2009). With this vision the center was interested in NLP technologies, not only in terms of their validity and reliability in parsing clinical content, but also in light of reusability of their output, their portability, extensibility, and ability to support ad-hoc and just-in-time integration and contextualization of clinical content.

On the other hand, the center was receiving health records and clinical content from multiple health information systems that contained unconstrained clinical text with high variability in use

of terms and documentation practices across data providers, and without commitment to grammatical or syntactic structure of the language (e.g. Triage notes, physician and nurse notes, chief complaints, etc). This was a limiting factor for the performance of available NLP technologies that rely heavily on the syntax and grammatical structure of the text.

As it was apparent from early on that most conventional and available methods were not going to satisfy our requirements, the center outlined the following criteria for the ideal text-understanding tool in this context and set the stage for conceptualization and implementation of a prototype design that could embody it.

Criterion A: Syntactic Resilience: The system performance (recall and precision rates) should be consistently good as the grammatical structure and quality of the text changes

Criterion B: Consistent Vocabulary Encoding: The output of the system should consistently encode each and every biomedical and clinical concept to a corresponding concept identifier from a standards based vocabulary system.

Criterion C: Formal Output Representation: The output of the NLP process should be self-descriptive and interpretable for a computational algorithm (formalness) and should be represented in an expressive knowledge representation framework that enables reasoning and inferences.

An early study of the field for the Criterion A revealed that enough empirical evidence does not exist to motivate selection of any of the traditional syntactic analysis frameworks to satisfy this criterion. While statistical (probabilistic) techniques alone or in combination with other frameworks such as CFG or DG are believed to be robust in dealing with incomplete or

grammatically aberrant text, creation and maintenance of these methods are generally complex and resource intensive tasks as they require a large body of annotated corpora for training and optimization. Furthermore, a generalizable technology that can be reused for biomedical and clinical text processing in different environments is not available today. A new theory of syntactic analysis for parsing clinical text was to be developed to overcome these barriers.

Criterion B with an eye on portability and extensibility requirements leads to selection of Concept Unique Identifiers (CUI) from UMLS Methatesaurus as the basis for encoding biomedical and clinical concepts in the output, in order to ensure versatility and authoritative mapping between interchangeable vocabularies.

Criterion C has many implications: being self-descriptive requires that most, if not all, system knowledge pertinent to interpretation and disambiguation of the output should be readily available and interpretable for external systems. That is, terminological, syntactic and domain knowledge should necessarily be referenced in the NLP output, and represented using a standards based, non proprietary, and formal information representation language that is interpretable by external systems. Furthermore, the system knowledge and the system output should be represented through a unified and expressive information and knowledge representation framework in order to satisfy this criterion appropriately.

Overview of the Method

The BLUE-Text employs a broad range of technological frameworks and standards from the Semantic Web community. BLUE-Text uses the Resource Description Framework (RDF) to represent all system knowledgebases, intermediate data, and the output. The Web Ontology Language (OWL) is used to construct and represent all knowledgebases as formal ontologies that

incorporate semantic, syntactic and domain knowledge utilized by the system. Our method can also be categorized as an ontology driven method as it extensively uses ontologies to inform system behavior. A combination of Description Logic (DL) and rules reasoning is used by the semantic application for classification and inferences.

The minimal syntactic, semantic method comprises of the following main components:

1- A set of ontologies representing syntactic, terminological, semantic and domain knowledge used for all system processes including syntactic analysis, semantic indexing, reasoning and classification, and output representation.

2- A text-parsing algorithm for minimal syntactic analysis of clinical content.

3- An semantic interpreter algorithm that uses the above ontologies for ontology mapping, semantic indexing, and output representation (Figure 12).

Figure 12 illustrates the overall process and the relationship between these components. After a preliminary text preparation phase, the Minimal Syntactic Parser performs a 2 step text parsing and syntactic analysis. The results of the syntactic analysis forms a parse graph that is comprised of tokens of text mapped to the Syntactic Knowledge. The semantic interpreter uses the parse graph and maps it to concepts from semantic and domain knowledge (ontology mapping) and indexes its tokens by a semantic indexer algorithm. The interpreter uses the semantic and domain knowledge, and the parse graph to construct another RDF graph called “conceptual graph” that closely represents the content of the input text. The conceptual graph is an intermediary and generic output that builds on the parse graph, and maps it to the appropriate concepts from the semantic and domain knowledge. All nodes in the conceptual graph have both a positional and a

semantic index that will be used by the output constructor to extract different formats of outputs defined by user, task, and/or queries. The semantic interpreter uses this information to improve conceptual graph through a normalization and disambiguation process.

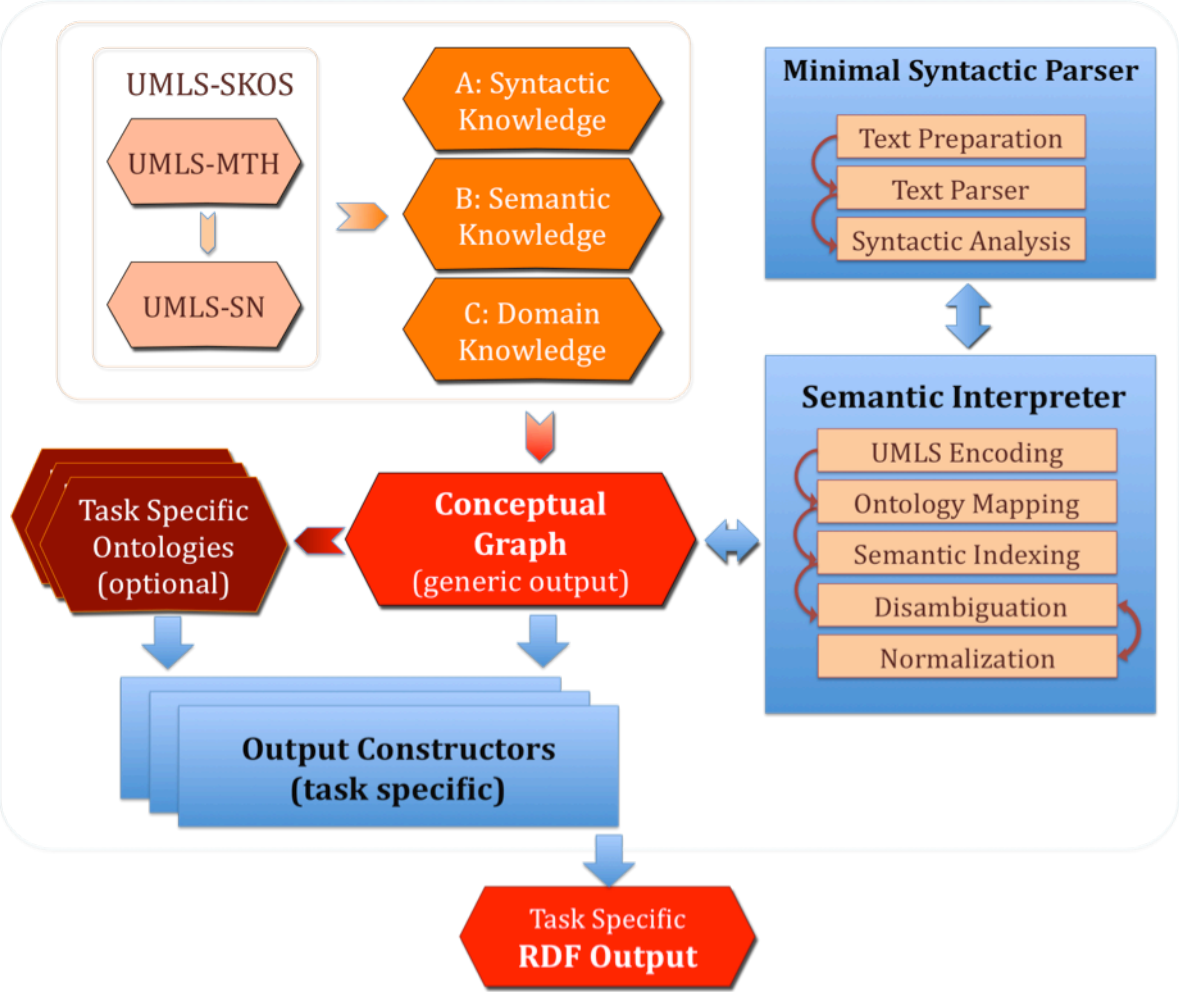


Figure 12: Schematic depiction of the BLUE-Text processes and ontologies

Figure 12 illustrates a serialized process where functions and methods appear to be called one after another, whereas in reality it is a parallelized process with an online reasoner that provides information about the meaning and significance of the tokens as soon as a link between the token and its syntactic and the semantic elements are established. This optimizes the performance of the semantic interpreter and the parser.

Why the Semantic Web?

The Semantic Web is an extension of current Internet technology in which the information is given well-defined meaning by making its underlying structure explicit. This makes the information both human and machine understandable and computable. This framework allows data to be shared and reused across boundaries of applications, enterprises, and communities. The Semantic Web technology is generally viewed as layers of technological frameworks as depicted in figure 13. Each layer extends and builds on the layer below and tends to be progressively more specialized and more complex. A lower layer doesn't depend on any of higher layers. Thus, the layers can be developed and made operational independently.

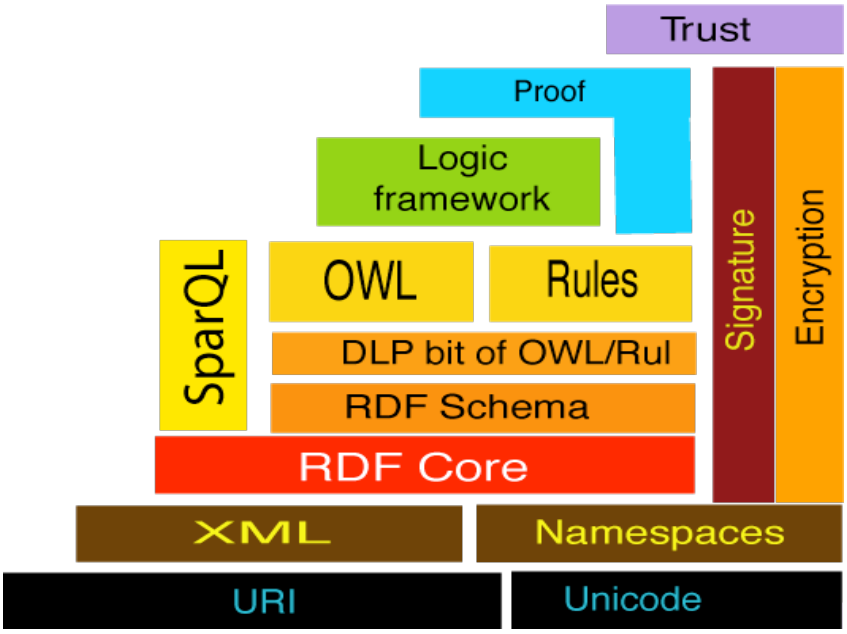


Figure 13: Layers of frameworks constructing the Semantic Web technology platform

Resource Description Framework (RDF): The RDF provides a general-purpose framework for representation of a web of information. An RDF statement comprises of three elements: subject, predicate and object, as in an English statement. Put together, all statements of an RDF document form an acyclic directed graph. Furthermore, each element in the statement (node) are

represented using a unique resource identifier (URI) that can be used to uniquely and globally identify any given resource in a web of distributed information.

OWL (Web Ontology Language): The Web Ontology Language (OWL) provides a rich and expressive language for defining structured ontologies. OWL is build upon the DAML + OIL (earlier generation of formal knowledge representation frameworks) and is an extension of the RDF Schema Language (RDFS) that provides modeling primitives for defining relationships between properties and resources. There are three major flavors of the OWL language: OWL-Lite, OWL-DL and OWL-Full. The BLUE-Text design uses OWL-DL as it supports maximum expressiveness while retaining computational completeness and decidability (Horst 2004).

System Knowledgebases (Ontologies)

BLUE-Text uses several ontologies (or ontology driven services) for syntactic parsing, semantic interpretation, mapping and reasoning services. As the concept of the Semantic Web advocates ontology reuse as much as possible, system knowledge bases are designed with encapsulation and modularity in mind so that each piece of knowledge can be maintained on its own, and shared with external systems ad-hoc, without imposing unnecessary ontological commitment to unwanted semantics and constraints. For example the terminological knowledge in the BLUE-Text comprises of several smaller, self-contained ontologies, each providing lexicon for a certain aspect of the clinical text understanding, such as negation and uncertainty, or units of measurement. A specialized terminology mapping method and service is designed to enable reuse and expose a given external OWL ontology as source of terminological or domain knowledge to the system, when a mature and reliable ontology was available from the web (e.g., Ontology of Units of Measurement from NASA Jet Propulsion Labs).

A: Syntactic Knowledge (Lexicon and Terminological Knowledge)

Terminological knowledge and lexicon of BLUE-Text is provided by a set of ontologies and ontology driven services. The BLUE-Text terminology service is able to import a given external ontology in an ad-hoc basis and use it as a source of terminological knowledge for syntactic parsing. The semantic knowledge may be extended accordingly in order to enable use of the new external ontology.

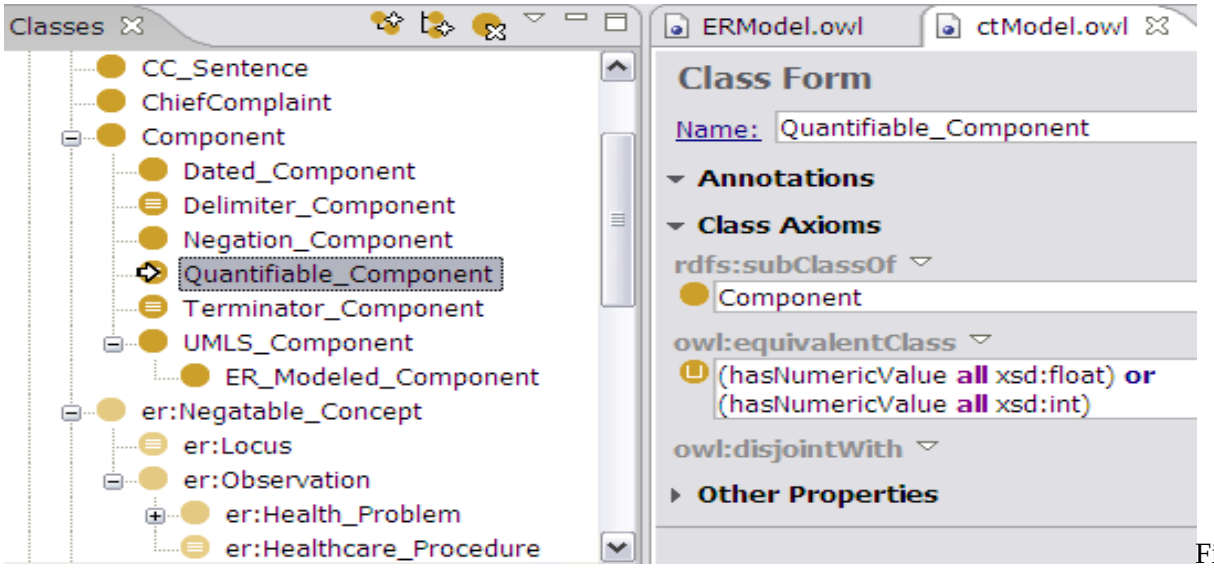


Figure 14: Definition of Quantifiable Component (number) in BLUE-Text

A.1. Syntactic Cues and Lexical Knowledge

A rather small ontology provides basic syntactic constructs used by the Minimal Syntactic Parser to identify a sentence, and its pieces in order to parse it to a minimum number of legitimate tokens. As the Minimal Syntactic Parser is language independent and has no grammatical commitment to a certain language, this model establishes a basis to identifying certain linguistic expressions that can be used by parser to identify differences in data types (e.g., Date, Time, Number etc), and some syntactic cues that may be reliably used for segmentation of a sentence (e.g., delimiters such as “,” or “.”). Figure 14 illustrates how tokens representing quantities are

defined in the BLUE-Text syntactic knowledgebase. This definition can be modified or replaced with another valid OWL expression in order to change the expectation of the Minimal Syntactic Parser from how numbers may be represented in a body of text.

A.2. The Lexicon

The Minimal Syntactic Parser uses an OWL ontology that represents a lexicon for the generic and mainly non-clinical aspects of the clinical content. The model represents each lexeme in terms of a unique resource identifier (URI) that can be referred to by many morphologically different symbols. Each lexeme is modeled as an instance of at least one semantic class in the Lexicon (e.g., "*ctm:Reject* [*reject, rejecting, rejected, rejects,*…]). Each class may have further semantics as inferred by its definition within the ontology. For example *ctm:Reject* may be a subclass of *ctm:Active_Negation*, whereas the *ctm:Unable* is an instance of both *ctm:Subjective_Negation* and *ctm:Passive_Negation* (Figure 15).

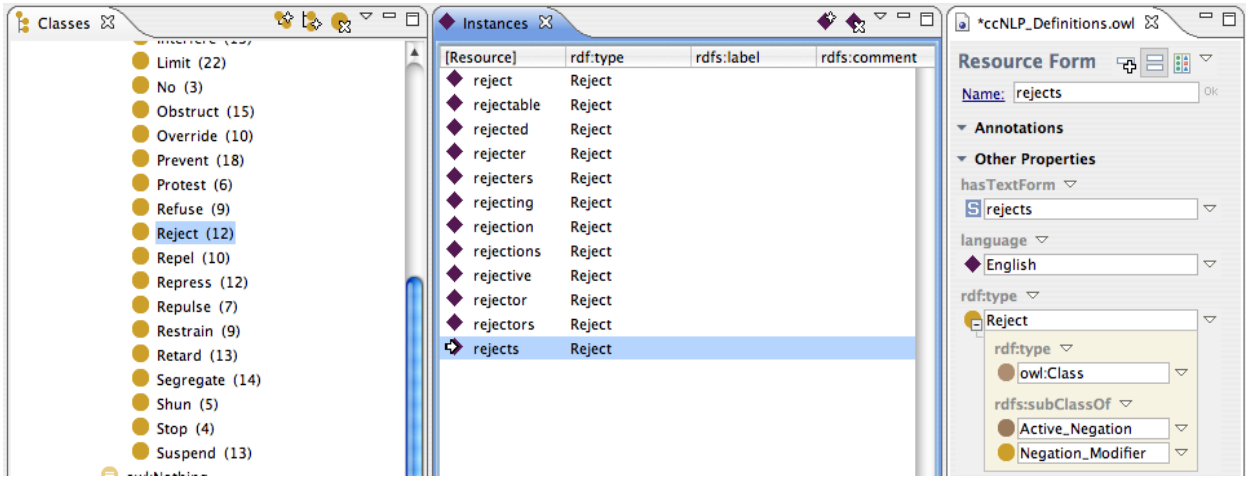


Figure 15: Lexemes and an instance of Negation ("Reject") in English

To support a somewhat language independent behavior, a lexeme can be used to represent a multilingual representation of a single lexical expression (using "hasTextForm property"), or multiple lexemes can be added to the Lexicon each belonging to a distinct language domain.

Such modeling techniques can be used to control identifying the clinical content as a whole or its parts as belonging to a different language domain and employing appropriate parsing and mapping function automatically.

The Lexicon supports the Minimal Syntactic Parser to identify surface expressions from clinical text that have a non-biomedical semantics. For example, all categories of Negation Expression, Uncertainty, names (of known real world objects, individuals, organizations, places), units of measurement, chemical and particles, etc.

Series of specialized services are prepared to import external reusable knowledge sources that contain appropriate labeling and terminological definitions to populate the Lexicon automatically. For example, if it is desirable for the BLUE-Text to also identify specific doctor names mentioned in a clinical text, the Lexicon could be extended to contain a new LDAP_Person class. Then the organization LDAP information can be queried to populate the Lexicon with information regarding names of the healthcare personnel, each being represented as a lexeme, and as an instance of the LDAP_Person class. If the LDAP_Person class is mapped as a subclass of the infM:Doctor class in the semantic information model (usingh rdfs:subClassOf relationship), the appropriate link between a syntactic expression and its appropriate semantic will be established and will be automatically followed by the semantic interpreter algorithm.

A.3. The Terminological Knowledgebase

One design principle during conceptualization of the BLUE-Text was to avoid problems associated with construction and maintenance of large and specialized clinical and biomedical knowledgebases. We were interested in reusing existing bodies of knowledge available from other authoritative sources such as National Library of Medicine (NLM) rather than constructing

one from scratch and specifically for this project. To provide the BLUE-Text with a comprehensive terminological and clinical domain ontology, the UMLS Knowledge Sources (UMLS-KS) developed and maintained by NLM (2003) was used as the main source of both the terminological knowledge and domain knowledge. The UMLS-KS provides an integrated Metathesaurus (UMLS-MTH) of biomedical and clinical concepts from over 200 different standard vocabulary systems (such as SNOMED-CT, LOINC, NCI Thesaurus, ICD etc). To support this project we developed an adaptation of the UMLS-MTH for the Semantic Web (Figure 2) that could be used (queried) to extract terminological and domain knowledge required by our system for text parsing, classification and reasoning.

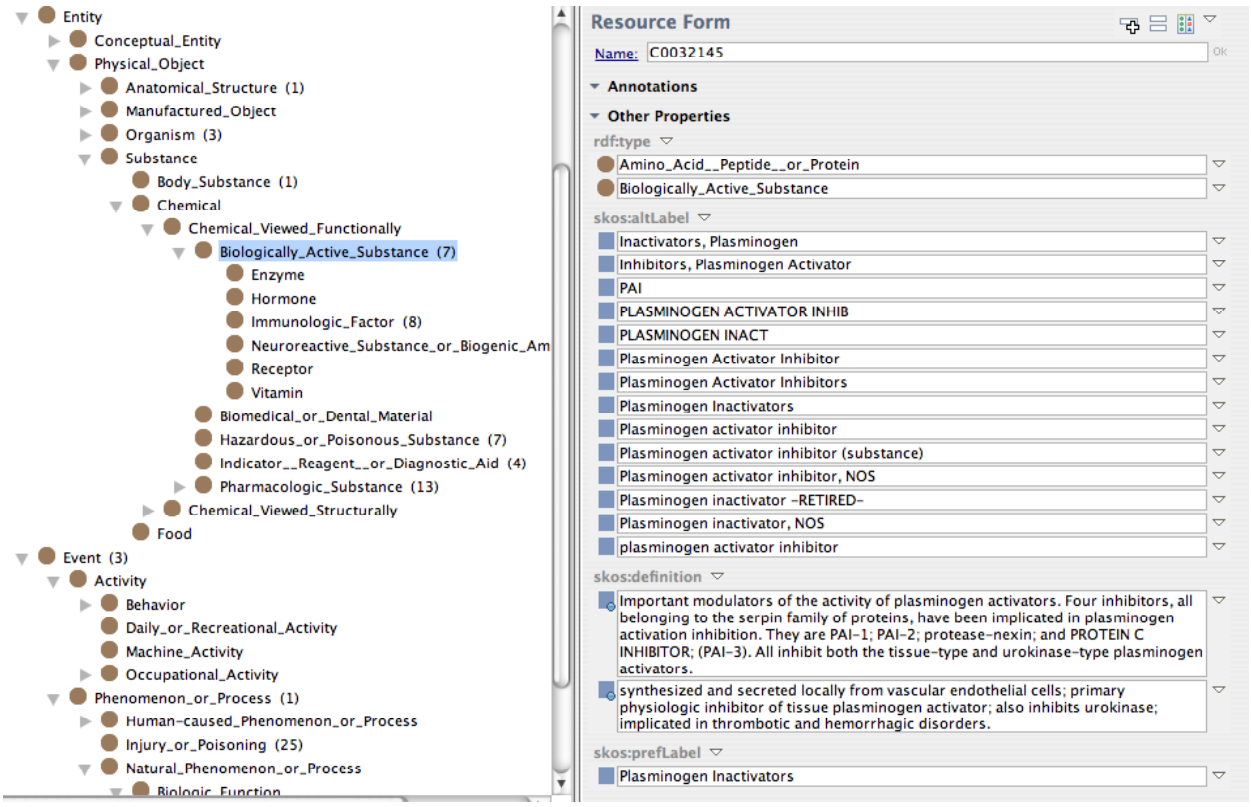


Figure 16: UMLS-SN and MTH represented semantically.

Each UMLS-MTH concept is provided with a unique concept identifier (CUI) that is used as a mapping point between concepts from multiple source vocabularies. Any textual representation

or ‘atomic term’ used by a source vocabulary to refer to a biomedical concept also has its own unique identifier (AUI). A CUI may be linked to multiple AUIs from the same or different source vocabularies. The UMLS-MTH also contains all relationships that a source vocabulary may have defined or describe between concepts or between terms. This qualifies the UMLS-MTH as a rich and expressive source of terminology for biomedical and clinical concepts. However the UMLS-KS as is cannot be readily used or queried by a semantic application, as the semantics of the relational schemata used to construct the UMLS-KS are implicit and not available for real time inferences for information retrieval and querying by semantic applications.

A semantic application needs to infer or retrieve relationships between domain concepts according to a knowledge source, retrieve terms used in clinical environments to refer to biomedical concepts, and identify the relative ‘codes’ associated with concepts according to different sources of vocabulary. In order to enable using the UMLS-KS to support the above requirements for a semantic application, the following transformational steps were inevitable:

- Relevant (probably all) biomedical vocabularies and taxonomies in the UMLS-KS needed to be explicated using formal information representation frameworks such as RDF. Simple Knowledge Organization System (SKOS) (Miles and Brickley Nov 2005) is an ongoing W3C standardization effort to support the use of knowledge organization systems (KOS) such as thesauri, dictionaries, nomenclatures, and taxonomies for the Semantic Web.
- Once UMLS source vocabularies were transformed into SKOS, a consistent method was to be devised for representing correspondence between concepts from multiple KOS, or for representing correspondence between concepts from OWL ontologies and KOS. The method should be able to precisely represent the semantics implied by the current UMLS-KS

relational schema, and it should represent terminological relationships between concepts and terms (synonymy, hyponymy, and hypernymy) or semantic relationships between concepts (such as part-whole relationship, parent-child, broader-narrower, and multiple different but related meanings) as asserted by the source vocabularies.

- A method of search and retrieval from a defined set of KOS needed to be crafted to retrieve relevant information based on a combination of concept names, synonyms, broader/narrower relations, codes, and coding schemes. These methods are traditionally implemented as Vocabulary Services within biomedical applications.

In order to support these features BLUE-Text implements the following:

- A SKOS model was developed (UMLS-SKOS) to represent the UMLS-MTH schemata, and the UMLS Semantic Network (UMLS-SN) (Figure 14) and all relationships extractable from combination the UMLS-KS. This translation enables the BLUE-Text to classify, infer or retrieve domain concepts based on UMLS-SN, and practically exposes the UMLS-SN as the domain ontology for text-understanding. We have extended the UMLS-SN inside the UMLS-SKOS model with properties to assert correspondence of concepts from any OWL ontology or SKOS concepts from other non UMLS source vocabularies, with UMLS-SKOS. A DL-reasoner infers correspondence between OWL concepts or SKOS concepts from multiple source vocabularies using transitive and functional attributes of the properties (e.g., *if concepts A and B both correspond to C, then A corresponds to B, hence all SKOS:Definition of A also applies to B and vise versa*).
- An algorithm to extract medical vocabularies from their source format and to translate to a SKOS based ontology useful for the Semantic Web application. The current implementation of the method can extract all source vocabularies included in the UMLS-KS and represent it

as SKOS ontology. We have also translated all 200 value sets from 15 vocabulary groups pertaining to the Public Health Information Network (PHIN) framework to a corresponding SKOS representation fully integrated and mapped to the UMLS SKOS ontology.

The algorithm in effect queries the current UMLS-KS relational schema as is, transforms the results on the fly to RDF and maps as individuals to the UMLS-SKOS concepts (populates the UMLS-SKOS ontology with a source vocabulary of choice).

- A Web Service[(Sun Microsystems 2004)] with methods to search and navigate the UMLS-SKOS in order to identify correspondence of terms, codes, and names used to describe domain concepts, based on the underlying sources of vocabulary (e.g., SNOMED-CT, LOINC, etc).

  Two versions of the service was conceivable (see A.4): a) a query interface to a pre-coordinated UMLS-SKOS ontology. The RDF transformation, and mapping processes to populate the UMLS-SKOS ontology have happened prior to the query. A simple query is enough to extract relevant information and return as RDF graph; or b) an online translational system that runs on real-time on an existing UMLS-KS relational database, transforms and maps to the UMLS-SKOS ontology and returns the results as an RDF graph. For practical reasons discussed below only the second method has been implemented for the prototype implementation (see below).

A.4. Terminology Service vs. Terminological Ontology

It is important to note here that to satisfy operational requirements of BLUE-Text the terminological knowledge could be supplied using two different methods (as described in the conceptualization of the Web Services associated with using UMLS-KS as source of vocabulary). These methods have different implications in terms of ease of implementation, maintenance, performance, and computational resources required.

The first method requires full RDF conversion of UMLS-KS (UMLS-SN and a customizable subset of the UMLS-MTH, filtered by source of the vocabulary), mapping into the UMLS-SKOS, and incorporation into a scalable knowledgebase with integrated reasoning facilities for fast querying. In this scenario the knowledgebase needs to be rebuilt as the UMLS-KS is updated (3-4 times per year), and expensive hardware infrastructure is required to ensure practically acceptable performance. However, relatively deeper knowledge-based queries, more complete inference and retrieval, with a better performance could be expected, especially in the case of complex queries.

The second method involves using a real-time processing method that transparently exposes the existing UMLS-KS relational schema and the associated database infrastructure ad-hoc and as a virtual semantic model. That is, terminology service requests would be mediated with a conversion algorithm that would transform and map results to a UMLS-SKOS based ontology before making them available to the semantic application. Although from the semantic application's perspective these methods both return a valid SKOS based ontology, the second scenario returns a relatively limited set of facts and supports classification based on a subset of relationships, as the current UMLS-KS schema does not have a built-in reasoning facility and does not explicate many of the meaningful relationships implied by the relational schema. Furthermore, multiple redundant query and transformation steps in the second scenario may hinder the overall performance. However, maintenance and update of the second scenario is easier and can be implemented with conventional infrastructure.

The current prototype of BLUE-Text has been constructed using the second method, although the conversion algorithms and all scripts necessary to implement and support the first scenario are also developed and evaluated.

B: The Semantic Knowledge:

An OWL ontology has been constructed to provide a generic and extensible information model for a prototypical clinical content. The model is conceptualized to serve as a high level schemata (information model) with minimal set of semantic constrains that sufficiently represent major patterns identifiable in a typical clinical text, and in the mean time enable ad-hoc extensions and mappings to more specialized ontologies by systems that intend to specialize it to meet particular requirements of a new use case or domain. This model replaces the need for a specific frame or template especially for each and every linguistic expression or pattern in the clinical content.

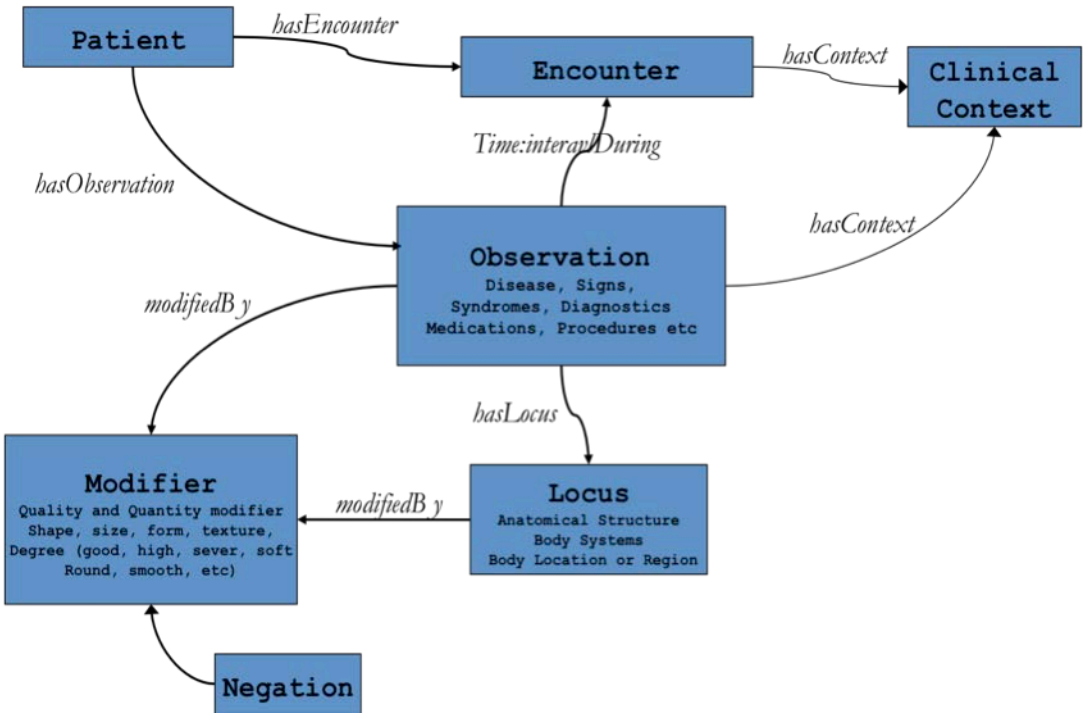


Figure 17: BLUE-Text Semantic Model; A high level model to interpret clinical text

This information model also provides mapping points for importing new semantic and syntactic knowledge, or extending it dynamically to meet requirements of a new type of document or domain (for example to add concepts pertaining to medications and prescriptions, in a model originally intended to capture vital signs and physical exam data).

Concepts such as Clinical Text and its different types such as Chief Complaint, relationships with Presenter (Patient, Nurse, EMS Personnel), Clinical Observation(Sign, Syndrome, Disease, Procedure), and their Locus (Body Site or Region, Body Part), Modifiers(QualitativeModifier and QuantitativeModifer), Clinical Contexts(Temporal_Context, Causation_Context, Process_Context, Allergy_Context, History_Context, etc) that can further explain implications of Clinical Observations are introduced in this model (Figure 17).

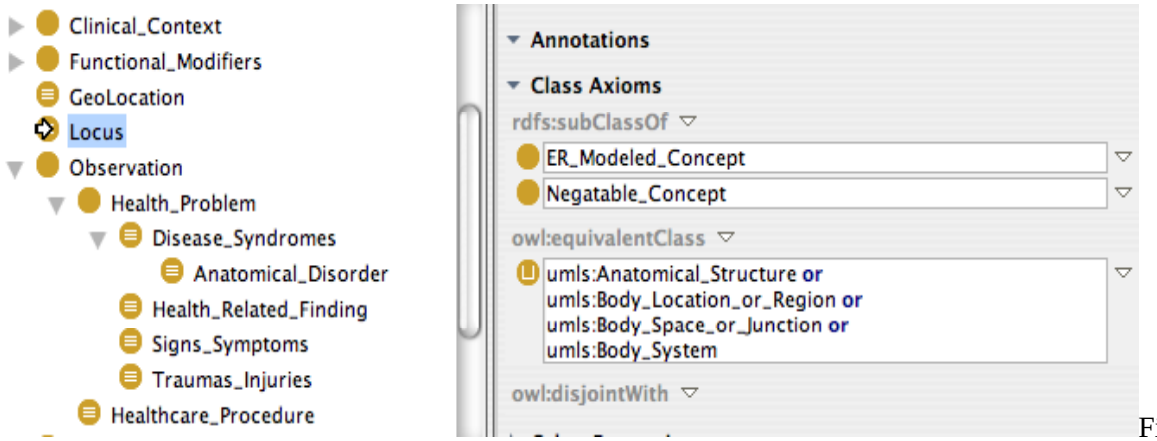


Figure 18: Extending semantics of Locus by UMLS SN and Negation

Most concepts in this model are meta-classes that need to be either extended or mapped to an existing ontology that further constrains their semantics and relationships with other concepts. That is the semantic model is only useful if it is extended and specialized by some other more task oriented ontology. For example, the concept of *Locus* in the BLUE-Text prototype implementation is extended by a logical expression that maps the *Locus* to the logical union of multiple concepts related to anatomical structures and body systems imported from the domain ontology. As the OWL representation of the UMLS-SN plays the role of the domain knowledge in the BLUE-Text, the *Locus* is further extended (defined) as the logical union of UMLS Semantic Types such as *Anatomical Structure, Body System, Body Location or Region* and others (Figure 18) . *Locus* is also extended further by another logical expression that maps it into a

"*Negatable Concept*" from another ontology (syntactic model) that models lexicon associated to negation and uncertainty (Figure 18).

This an example of how new and interchangeable knowledge can be incorporated into the system to enable proper handling (understanding) of statements that describe anatomical concepts. In this case, system sanctions negating relationship between *Observations and locus* (e.g, "*Rash on both hands, but not on the face*"), by making *Locus* a "*Negatable Concept*".

The semantic model provides a simple but extensible and generalizable model of the clinical information extractable from text. The semantic interpreter algorithm has awareness of only root concepts of the semantic model (Presenter, Locus, Observation, Functional Modifier, etc). That is, as long as the semantics of root concepts in this model remains intact, the model can be dynamically extended by new imports, all its imports can be changed dynamically without reprogramming BLUE-Text algorithm, in order to customize and port the method to new environments. An example of this dynamic extension in this case is presented here by using the UMLS-SKOS ontology to extend the semantic model and further constrain it in the prototype implementation of the BLUE-Text.

Extensions of the Semantic Knowledge

A: There are a number of heuristic rules to avoid inconsistencies and modeling problems rooted in UMLS-SN and UMLS-MTH (Hahn, Romacker et al. 2002; Kashyap and Borgida 2003). For example it is possible for the same UMLS-MTH concept to be an instance of *Qualitative_Concept* and *Organization* at the same time. The semantic interpreter uses the rules to update the conceptual graph and ensure its consistency for reasoning and retrieval tasks.

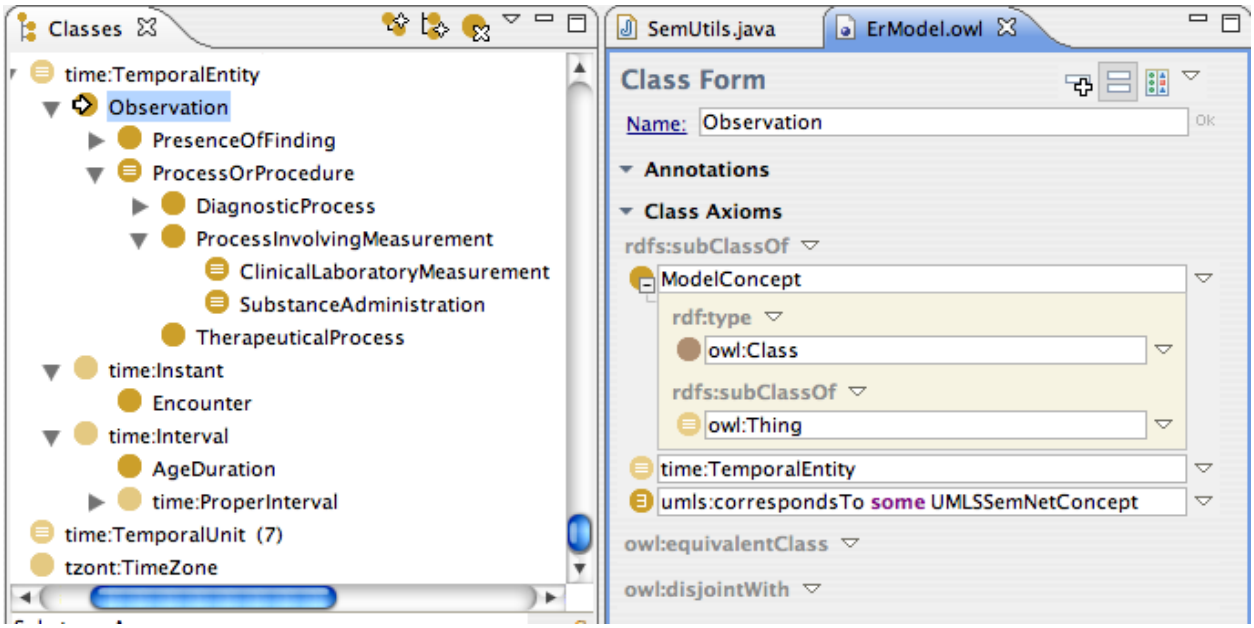


Figure 19: Temporal entities such as the Observation and the Encounter

B: Multiple OWL ontologies that collectively represent the Syntactic Knowledgebase extend the Semantic model to provide mapping between lexical expressions and their meaning (as demonstrated in the Figure 18). This enables interpreter to use the new syntactic knowledge in its correct context. For example, the infM:Presenter or its subclass infM:Doctor can be used to map to LDAP_Person concept from the Lexicon to enable BLUE-Text to identify names of doctors in a healthcare organization.

BLUE-Text uses this mapping and extension process as a training facility to acquire new concepts that might enable the algorithms to identify and properly handle (understand) non-medical concepts generally absent from biomedical knowledge bases. For example, valid indicators of Negation (e.g., “deny, stop, reject, not, unable”), units of measurement, people and organizations, etc are all modeled as dynamic components of the Lexicon and made available to the system through mapping them to an appropriate concept from the semantic model.

C: The semantic model and the information model that it provides, is further extended by the Time Ontology (Hobbs and Pan 2006) to support representation of time and temporal relationships and to enable temporal reasoning about clinical events. The *Encounter* and *Observation* are concepts conceptualized as the logical subclasses of the time:*TemporalEvent* from the Time Ontology.

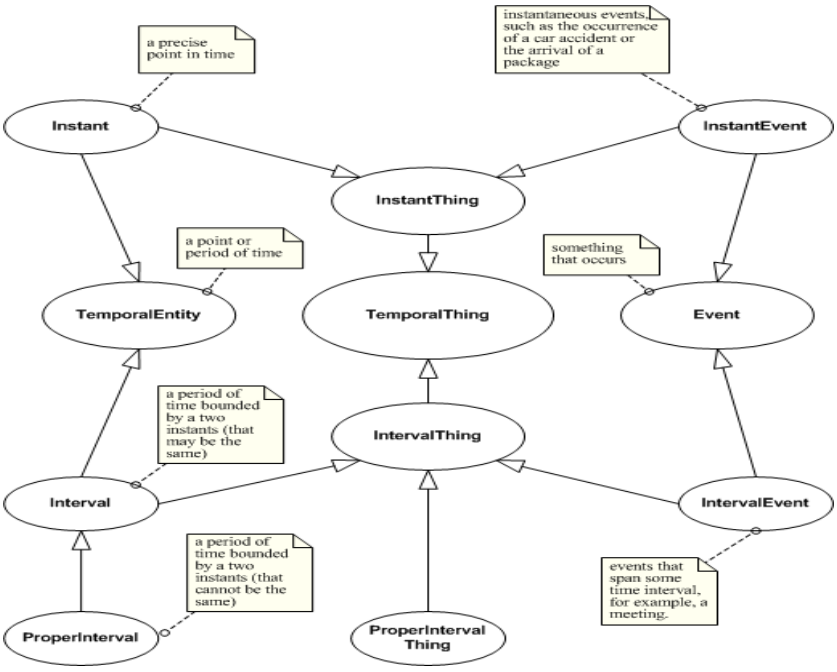


Figure 20: A high level representation of Time Ontology

This enables the semantic interpreter to identify and describe patient encounters and clinical observations with associated temporal information such as relative and absolute date-time information that could be extracted from text, and draw conclusions and inferences regarding the temporal relationships between clinical observations and patient encounters (before, after, during, past, present, etc…)

The Time Ontology also provides with some vocabularies and terminologies that are used by the Syntactic Parser as part of the lexicon accounting for temporal expressions such as ‘month’,

‘year’, and named concepts such as ‘Wednesday’, and ‘February’, and their underlying semantics.

C: The Domain Ontology (Domain Knowledge)

A text-understanding application intended to operate in a biomedical and clinical environment requires an ontology that formally describes domain concepts (e.g., *Diseases*) and semantic relationships between them (e.g., *All Infectious Disease are Caused by some Infectious Agent*). Although candidate ontologies have started to appear that claim comprehensive and formal description of biomedical concepts (GALEN(Rector, Rogers et al. 2003), FMA(Cook, Mejino et al. 2004) , NCI Thesaurus(Ceusters, Smith et al. 2005)), these ontologies are too large and complicated to be effectively used and maintained even by trained human experts(Seidenberg and Rector 2006).

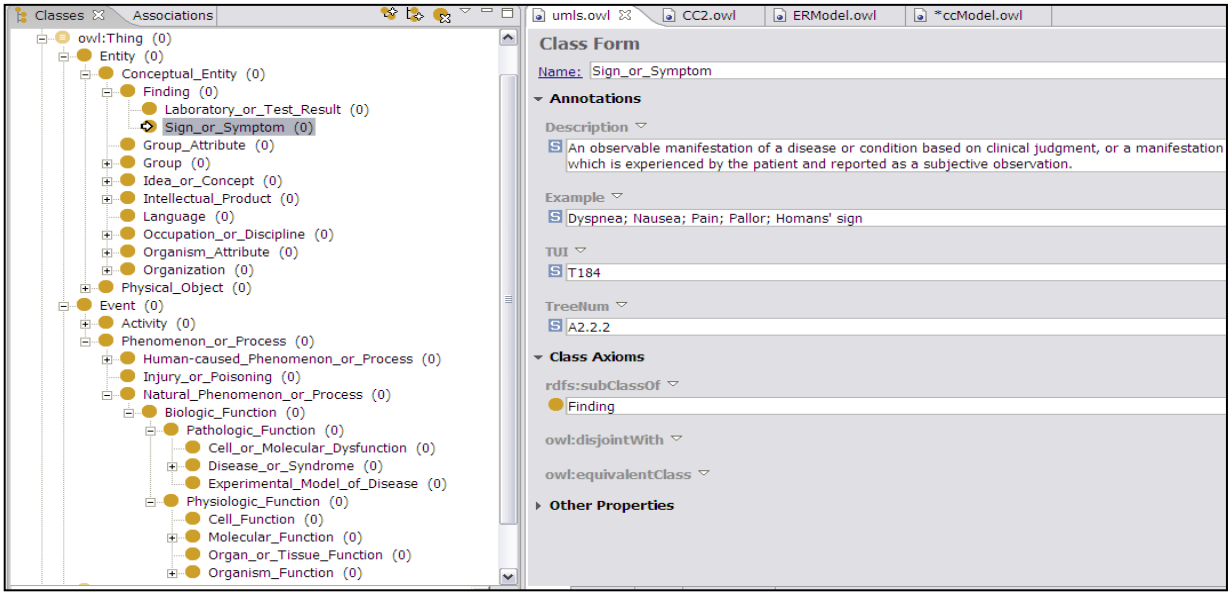


Figure 21: UMLS Semantic Network Represented in OWL

Rather than developing an extremely large biomedical ontology for BLUE-Text that describes every biomedical concept, we chose to adopt the SKOS conversion of the UMLS-KS (described above as UMLS-SKOS) as the source of biomedical domain knowledge with a method that

keeps the original UMLS-KS in place and only attempts to query and translate relevant aspects of the UMLS-KS to an SKOS representation on-demand (as explained above).

The UMLS-SN is part of the UMLS-KS (2003) that provides an abstract classification scheme for all biomedical concepts represented in UMLS-MTH. In the current version of the UMLS-SN there are 135 Semantic Types (nodes) that are networked through 54 Semantic Relationships (links). All UMLS-MTH concepts are assigned at least one Semantic Type with the most specific semantic in the UMLS-SN hierarchy. Semantic Types contextualize UMLS-MTH concepts with textual annotations that define their types, and place them in an 'isa' hierarchy (Figure 21).

The UMLS-SKOS is an OWL ontology that partially but consistently adopts the UMLS-SN for the Semantic Web applications (Figure 21). The model maps each Semantic Type into a corresponding owl:Class and each UMLS-Semantic Relationship into an owl:ObjectProperty. Concepts and Properties in this model have rdfs:subClassOf and rdfs:subPropertyOf relationships when there is an 'isa' relationship in the UMLS-KS. To avoid inaccuracies and inconsistencies associated with direct translation of the UMLS-SN to OWL, we did not translate some of the assertions of the UMLS-KS. As some of the semantic relationships within the UMLS-SN Semantic Types could not be consistently and reliably translated into OWL, our translation algorithm avoids representing them all together (e.g., rdfs:Domain, rdfs:Range, or universal or existential constraints on properties) (see (Kashyap and Borgida 2003) for a complete discussion of incompatibility of the UMLS-SN with the OWL semantics). This minimal adherence to the UMLS-SN semantics only when it applies consistently and without exception (OWL is a monotonic language), enables semantic applications to extend UMLS SKOS ontology without committing to an inaccurate and inconsistent translation.

In our conceptualization of the UMLS-SKOS each UMLS-MTH concept represents a resource with a unique resource identifier (URI) constructed using a NameSpace:CUI schema, where NameSpace can represent any unique URL such as 'umls=http://nih.nlm.gov/umls/'. All UMLS-MTH concepts are conceptualized to be instances of (rdf:type) the Concept representing its associated Semantic Type (Figure 16). For example the "*Plasminogen Inactivator*" with the CUI=C0032145, is a resource uniquely identified by the uri='umls:C0032145' in the UMLS-SKOS and has two semantic types of "*Amino Acid, Peptide, or Protein" and "Biologically Active Substance*". (Figure 16).

The semantics of each UMLS-SKOS resource (each UMLS-MTH concept) is defined by its source and through variety of means: by a textual definition or annotation; by its Semantic Type and its place in the hierarchy; by source defined relationships between concepts, and by terminological relationships between terms (hyponymy, hypernymy, synonymy, etc) defined by the UMLS-MTH. There are major groupings of Semantic Types incorporated in the UMLS-SN and therefore in the UMLS-SKOS for organisms, anatomical structures, biologic functions, chemicals, events, physical objects, and concepts or ideas. However, although the current network representation of the UMLS-SN cannot be directly utilized by the Semantic Web systems (Kashyap and Borgida 2003; Lussier and Patel 2004), the broad scope of our UMLS–SKOS translation allows for extensions that enable classification and reasoning in a ranges of applications related to the biomedical domains. For example, Figure 22 demonstrates how two UMLS Semantic Types (*Phenomenon_or_Process and Chemical_Viewed_Functionally*) have been used to express logical constraints that define the new concept of '*SubstanceAdministration'* inside the semantic to represent a new clinically meaningful pattern (an *Observation* that involves *administration* of at least one *chemical* with a known function,

along with some optional *dose*, *frequency* and *route* information. Remembering from previous section, an *observation* in this model is a *temporal entity*, that is, a *substance administration* will be sanctioned to have a relationship with a *temporal entity* such as an absolute or a relative time (e.g 2 hours ago).

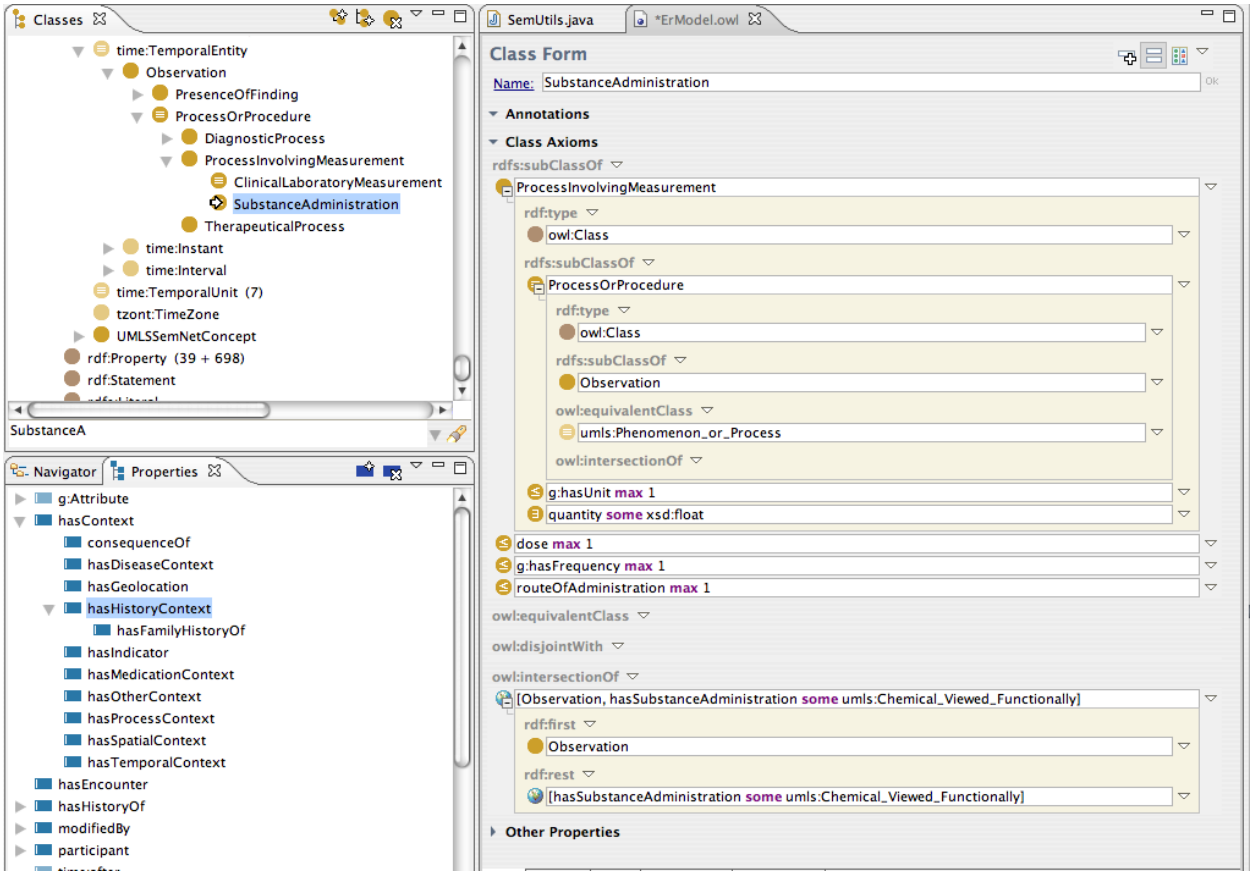


Figure 22: Definition of the 'Substance Administration' using UMLS-SN concepts

Syntactic Analysis

Text Preparation

The preparation process accounts for alterations in text that generally intend to improve its quality for future processes. Expansion of known abbreviations, acronyms and normalizing some known variations as defined by the syntactic model are part of the text preparation process.

Spell-Checking: The open source GNU Aspell (Atkinson 2008) spell-checking tool was modified to fit the requirements of spell checking of the biomedical and clinical content. In addition to the generic English language dictionary provided by the GNU Aspell, a collection of biomedical and clinical dictionaries containing lexical information for UMLS semantic types such as chemicals, disease, signs and symptoms, and anatomical concepts were compiled using information extracted from the UMLS-KS. The spell checker can mix and match different dictionaries on demand and based on the characteristics of different clinical contents and their requirements. As we have not fully studied the completeness of these dictionaries and reliability (sensitivity and specificity) of the different possible configurations of the spell checker in various clinical environments, the spell checker function was turned off during the evaluation. Use of the spell checker during text processing is optional and can be turned on or off as required.

Syntactic Analysis and Text Parsing

A custom developed parser algorithm computes an indexed array of all permutations of tokens extractable from the text based on the position of syntactic components defined by the syntactic knowledgebase. A token is any ordered combination of words extracted from text. Tokens are defined by their positional index (their distance from the beginning of the text) and their length (number of words they contain). Tokens can overlap, contain or trail each other (Figure 23).

The syntactic parser is in the heart of our minimal syntactic algorithm, and is conceptualized to segment text into tokens in a language independent manner. A design objective has been to reduce the negative impact of grammatical and structural aberrations of the clinical text in accuracy and reliability of the NLP output.

Most theoretical frameworks for syntactic parsing introduced in the previous section rely heavily on a language and syntax dependent parse algorithm, and their use connotes the assumption that the method will be used on grammatically conformant clinical text. Statistical methods are believed to be resilient to the completeness of sentence structure but require extensive human interaction to obtain and maintain reference corpora for training and maintenance.

A minimal syntactic, language independent parser

A new text parsing algorithm was developed based on the magic number 7 rule (Miller 1956) in order to enable segmentation of text independent of grammatical constrains of underlying language and without human interaction. We hypothesized that because human readers have only 7± 2 short-term memory slots available to them when reading text, the largest understandable token can only hold 7± 2 words, that is a token cannot be longer than 7± 2 words. Furthermore, we also hypothesized that all contextual information and relationships important to fully understand any given token in the text should be only maximum 7± 2 tokens apart from it. That is, any meaningful piece of text can only be maximum 7± 2 tokens away from the next related and meaningful piece of text, or otherwise human mind may have difficulty to fully understand it in the context of other related pieces of text. Another assumption in conceptualization of our parser algorithm was that some 'words' have mainly syntactic and grammatical bearings, and are used as cues to the reader to ensure an efficient reading experience (here called 'Stop Word'). Examples of these words may include but not limited to words such as ['so', 'the', 'as', 'that', 'which', etc). The hypothesis was that presence of these words could be ignored during segmentation and indexing process, to minimize the number of tokens without impacting the outcome.

This hypothesis was empirically validated before implementing the parser algorithm. An experiment was setup to evaluate validity of these assumptions. Human experts annotated 200 sentences randomly sampled from real clinical text for the existence, relationships and distances of related concepts in each sentence. Table 1 reports results of the preliminary analysis before and after excluding stop words from the analysis. As shown in Table 1, anatomical and modifier concepts (negation, size, degree and temporal modifiers) in a clinical text were almost invariably 1 to 6 words apart from their associated observations (disease or pathological process). When counting the stop words, the average distance between related tokens were 3.6 ± 1.8 words, whereas when excluding stop words, the average distance between related tokens were only 2.5 ± 0.7 word. This information identifies the token size of 3 words (after exclusion of the stop words) as optimal to capture the majority of relationships for a language independent tokenizer that creates and arranges tokens without using syntactic and grammatical rules. This experiment also identifies one exception for this empirical rule, being the temporal modifiers. Temporal relationships between tokens were an average of 3.6 ± 1.4 words apart. This is not only another evidence to the validity of the magic number 7 rule, but also to the validity of the primary assumptions that informed the conceptualization of a minimal syntactic and language independent parser, and a semantic interpreter algorithm.

Table 2: Distance of related words in clinical text.

Concept Type	N	Min	Max	Avg(SD)	Avg(SD) without Stop Words	Outliers: Distance (N)
Anatomical Location	199	1	8	3.45 (1.63)	2.49 (0.73)	≥10 (2)
Negation	20	1	2	1.20 (0.41)	1.20 (0.41)	
Degree Modifier	44	1	3	1.44 (0.72)	1.38 (0.65)	
Temporal Modifier	48	1	9	3.58 (1.84)	3.58 (1.37)	≥10 (1)
Size Modifier	5	1	3	1.80 (0.84)	1.60 (0.55)	

The parser first scans through the text to create larger segments of text (*Evidence Spaces*) based on syntactic cues found from the syntactic model (Figure 23). Evidence Space is a token, closest to a sentence or a phrase (remembering that the parser does not know about the concept of Sentence). A sentence may break into multiple or a single Evidence Space. Examples of such syntactic cues are delimiters such as "," and "." that normally signify a stop point (such as a sentence). Evidence Spaces are ordered, and are parsed individually by the tokenizer to create all permutations of legible tokens based on the above heuristics as it maintains the order of the Evidence Spaces according to the text (Orange arrows in Figure 23).

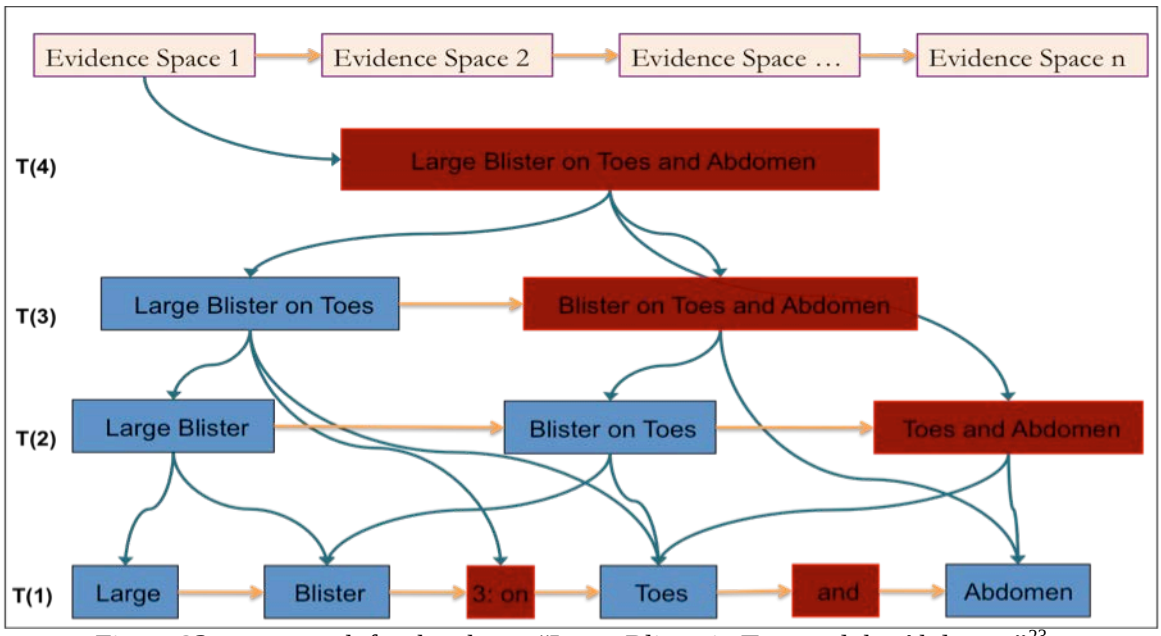


Figure 23: parse graph for the phrase "Large Blister in Toes and the Abdomen"[23].

To reduce the size of combinatorial space, an algorithm based on the regex expressions [see appendix] uses the lexicon provided by the syntactic model to identify and tag tokens with least possibility of representing a single unique concept (tokens containing *dates, time, numbers, separators, etc),* or those whose type is already identifiable by the links between the syntactic model and the semantic model *(named objects (People, Devices), units of measurement, negation*

[23] T is the token size in each row. "Large Blister" and "Toes and Abdomen" have the same size T(2).

marks etc) (tokens in red boxes from Figure 23). The parser algorithm simulates a 'reading' behavior that uses syntactic cues from the Lexicon and the rules to chunk the text into the least number of legible tokens for further processing, without any awareness of the grammar of the language.

The output of the syntactic parsing is a parse graph (Figure 23) that represents Evidence Spaces and their tokens. Evidence Spaces and their order represent the phrases and sentences, and their order in the text. Tokens of each Evidence Space are also ordered (Orange arrows in Figure 23). A large token may *contain* smaller tokens (Green arrows from Figure 23). The parse graph is a directed graph with a non-hierarchical structure (a network) that maintains an index of all tokens and their positional information from original text as well as their containment information. A parser can effectively query this parse graph to extract a Parse Tree consistent with the Phrase Structure Grammar, or a Dependency Diagram consistent with the Dependency Grammar. The figure 24 shows the corresponding output of a syntactic parser using a typical context free grammar or dependency grammar.

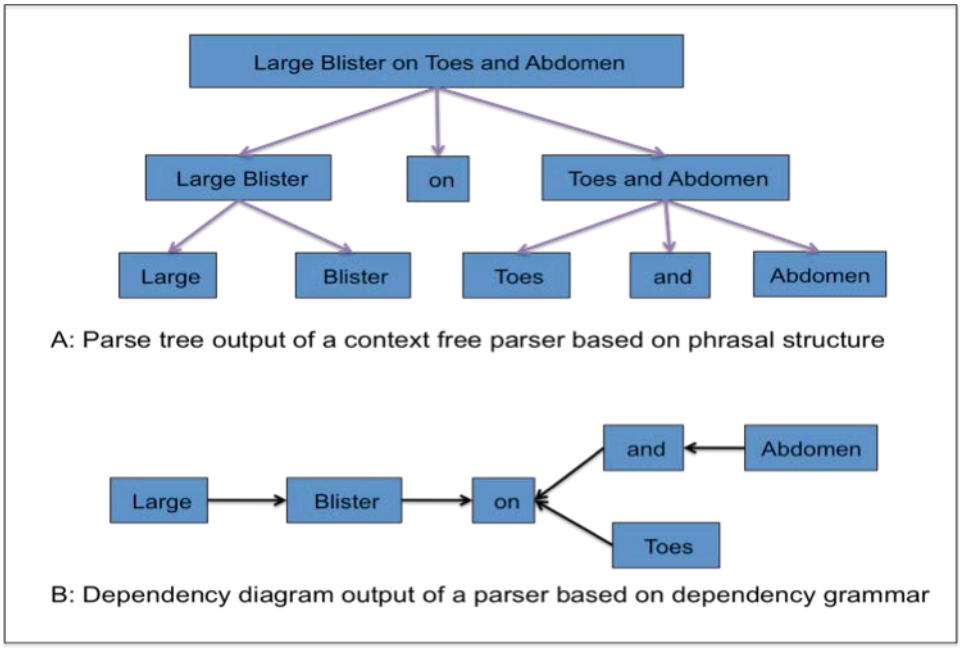


Figure 24: Parse Tree (A) and Dependency Diagram (B) equivalents of the Parse Graph in the Figure 23

Concept Mapping (Ontology Mapping):

The output of the syntactic analysis is an RDF document (the parse graph) that represents tokens extracted from the text and their positional relationships. Tokens are also described further by the syntactic knowledgebase if a link between them and a lexicon item has been identified (e.g., stop words, negation, numbers, date and time information, names, etc). An online reasoner and classification function infers and updates semantics of a token based on the syntactic knowledge base as soon as they are created (e.g., <:token1 lex:hasValue "John Doe"> implies that <:token1 rdf:type lex:OLAP_PhysicianName). As the Syntactic knowledgebase is also mapped to the Semantic Knowledgebase, semantic entailments of a token based on the Semantic knowledgebase is also inferred by the reasoner: <:token1 rdf:type infM:Physician> , <:token1 rdf:type infM:Presenter>, <:token1 rdf:type infM:Person>.

As the BLUE-Text lexicon only contains information regarding basics of the underlying language (such as stop words, days of week, negation marks, etc) and does not include

biomedical and clinical terms, none of the tokens representing biomedical and clinical concepts would be identified and associated with a counterpart from the Semantic Knowledge at this stage.

A filter function sifts through the remaining tokens from the parse graph and extracts legible tokens for mapping to the biomedical and clinical concepts (light green boxes from the Figure 23). A set of rules informed by the Syntactic Knowledgebase guide the filter function. For example tokens that contain a stop word (e.g., ‘and’, ‘with’, ‘whose’) as their first or last position or contain stop words such as ‘and, is, so, etc’ in the middle are not eligible for mapping to biomedical and clinical concepts. These rules enforce some de-facto criteria for terms used by clinical concepts and reduce the combinatorial size of tokens generated by a minimal syntactic algorithm even further. Only those tokens that meet these criteria will be eligible for the next step (ontology mapping) process as outlined below:

The MMTx linguistic analysis and concept mapping tool from NLM is used to map eligible tokens to the UMLS-MTH. While all eligible tokens will be processed by the MMTx, only tokens with a MMTX mapping score of 1000 (a perfect match with at least one UMLS-MTH concept). The CUI and Semantic Types associated with the token are returned as the results of this process. Figure 25 depicts the outcome of concept mapping. The MMTX algorithm adds the link between a given token and a corresponding CUI using the :correspondsToCUI property. This associates the token with the UMLS-SKOS resource defining the corresponding CUI and its Semantic Type(s). As mentioned in the section describing the system knowledgebases, the SKOS representation of the UMLS-SKOS model is being used as the source of both Terminological and Domain Knowledge for the BLUE-Text and is mapped directly to the Semantic model. As soon as a token is linked to a corresponding CUI the online reasoner infers class membership of the

token with a corresponding class in the Semantic model. The inferred knowledge is represented as dashed arrow from the token to the information model concept in the figure 25.

As a CUI may have multiple Semantic Types, or a given token may correspond to multiple CUIs, it is inevitable that a token may become an instance of more than one class in the Semantic model. This may lead to inconsistencies if the Semantic model and the domain knowledge provided by UMLS are not consistent. As mentioned in the section describing extensions of the semantic knowledgebase, several assertions and rules have been added to the Semantic Knowledgebase to avoid such inconsistencies and reduce unwanted and nonsensical inferences that may occur due to knowledge representation issues of the UMLS-KS.

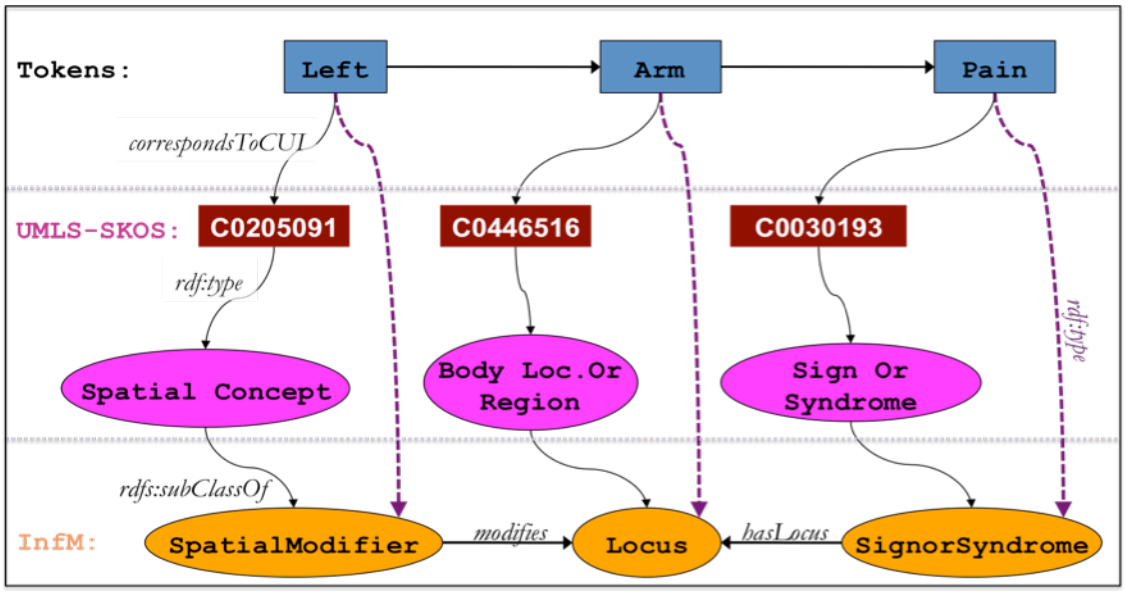


Figure 25: Mappings of Parse Graph to the Terminological and Semantic knowledge

In short, the Concept Mapping process takes the RDF graph generated by the syntactic analysis (Parse Graph) and tries to connect each token with some owl:Class from the Semantic model. That is, the Parse Graph is further extended by information regarding mapping of each token to a related concept from syntactic or semantic knowledgebase. Each token in the resulting RDF

graph is represented as an instance (rdf:type) of at least one concept (owl:Class) from the semantic model. Extensions and modifications to the OWL ontologies representing the Semantic model may affect the class membership and classification results. This can be used as a vehicle to customize and contextualize the behavior of the system for different use cases, without changing the algorithm.

After the mapping is complete and the inference engine reaches a stable state, a filter function discards from the Parse Graph all tokens that have failed to map to at least one class with an asserted or inferred mapping to the Semantic model. At this stage the process of extraction and encoding is complete in that the interaction of the tokenization, mapping and filtering functions have extracted all meaningful concepts identifiable using the combination of the system lexicon, the terminological and domain knowledge (UMLS-SKOS) and the Semantic model. However, the only relationship between the tokens extracted from the text is their positional relationship (order in which they appear in the text, and their containment information).

The Conceptual Graph

A semantic interpreter adds an index to all tokens based on their semantics extractable from the syntactic and semantic knowledgebase, and its linkage to the UMLS-SKOS model that serves as the domain knowledge for the BLUE-Text. The indexer uses heuristics associated with the magic number 7 (e.g., allowable distance for related concepts), few syntactic cues learned from Syntactic model (e.g., the role of ‘and’, ‘or’, ‘in, on, into, upon, of’ etc), and semantic relationships defined in the Semantic model and the Domain Knowledge to transform the parse graph into a conceptual graph in which tokens are related to each other based on a set of generic relationships other than their position in the text. Relationships between tokens in the conceptual graph are similar in utility to the edges in a dependency diagram, in that, they indicate

relationship between tokens without making an assumption about its nature and a specific meaning. However BLUE-Text conceptual graph uses a richer set of relationships that can be used to distinguish a syntactic relationship (e.g., containment, positional order) from a semantic relationship (e.g., negation, quantifier, locus, etc).

Figure 26 illustrates an example of the conceptual graph. Note that the tokens related to "Rash" and "Scar" both are related to the "Face" through a "precede" property but have no relationships with each other, and that the semantics of how this precedence should be interpreted, and what it may mean in any context is not represented.

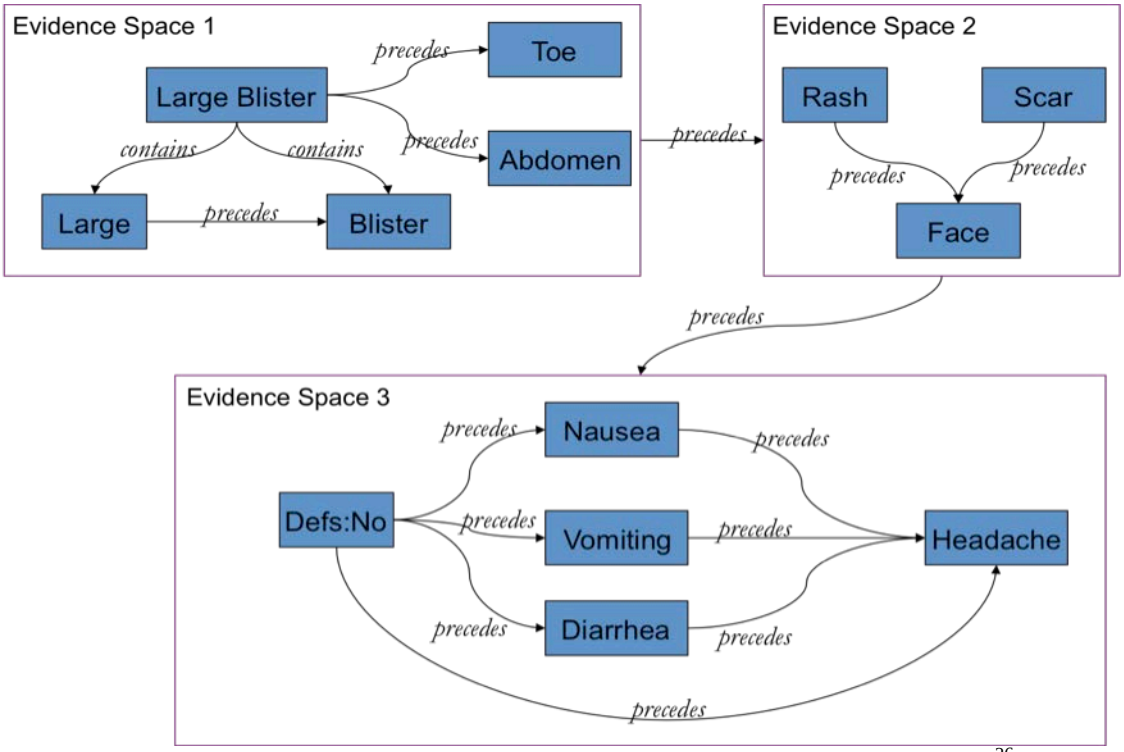


Figure 26: The Conceptual Graph, Evidence Spaces and Tokens of a phrase[26]

The conceptual graph is an intermediate output that represents tokens of clinical text mapped to concepts from ontologies with formal semantics and encoded with at least one UMLS-MTH CUI

[26] "Large Blister on Toes and Abdomen, Rash and Scar on the Face, no n/v/d or headache"

when possible, linked to each other and to their meaning in the ontologies available to the system. This enables any third party parser, classifier, or reasoner to be able to use the conceptual graph for further processing, querying and contextualization to construct outputs specific to their local needs, without having to commit to or agree with the conceptualizations modeled into the BLUE-Text Semantic model.

Output Constructor

To demonstrate how the conceptual graph can be contextualized for use in a clinical context the BLUE-Text algorithm has been extended with an output constructor service. The constructor is a mapping algorithm that uses the conceptual graph, the Semantic model, and a set of mapping rules to generate a formal output. The result is an OWL ontology mainly based on the Semantic model and its extensions, populated by instances derived from the conceptual graph. Figure 27 illustrates the output corresponding to the conceptual graph from Figure 26.

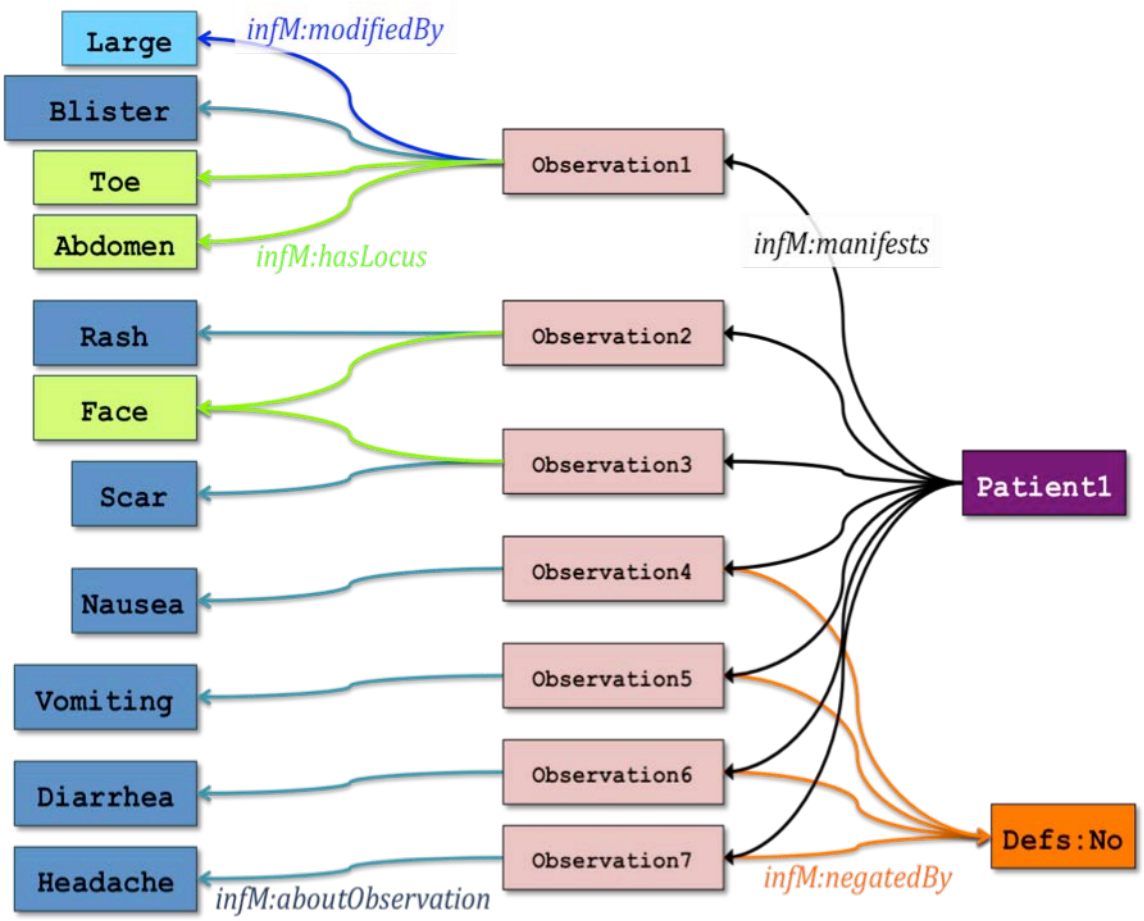


Figure 27: The Formal RDF output corresponding to the Conceptual Graph in Figure 26

The Semantic model used to formally define concepts present in the output and to interpret the relationships between tokens from text is also included in the output, and makes it a self explanatory and explicit document that can be interpreted buy other computer programs immediately. As seen in this figure, an instance of a *presenter* (Patient1) is associated to the instances of an Observation concept (using manifests property). The Observation and its subclasses define different classes of non-binary relationships between a presenter (patient) and a clinically relevant finding (e.g., Blister). A set of properties associate an instance of observation to the date-time, the UMLS-SKOS concepts defining that class of observations, anatomical location of the observation (e.g., Face), and its quantitative and qualitative (e.g., Large) modifiers.

Absence or negation of a clinical observation is also represented accordingly and using a specialized property (:negatedBy). The Syntactic model provides a distinct model and namespace for classification of different types of Negation in clinical text (i.e., Defs). This enables a reasoner to use the negatedBy property and the Syntactic model to distinguish between for example subjective negations (*patient denies having fever*) and objective negations (*patient does not have fever*) and avoid conflicting or inconsistent information in the knowledgebase (*Patient denies having a fever but EMS recorded 102F fever at the time*).

New constructors can be developed on demand to produce HL7-CDA messages (Dolin RH 2001), to map to other templates, frames, information models and databases, or ontologies as need, and to customize classification and retrieval of information for novel use cases (e.g., return candidates eligible to new research protocols). In all these use cases, the NLP algorithm itself is untouched and only the models used and the constructor script is modified to contextualize the output for reuse.

Evaluation

Introduction and Overview

An objective evaluation process is one that helps to reliably predict behavior of a text-understanding system in a clinical setting. However, established criteria for the performance evaluation of text-understanding systems do not yet exist, as the text-understanding systems are still in early stages of conceptualization and design, and have not been widely adopted yet. However, contingency tables as depicted in Figure 28 are frequently used to record and calculate reliability and validity measures for the natural language processing algorithms (Hamm 2002; Meystre and Haug 2006).

		Gold Standard	
		+	-
CTU	+	TP	FP
	-	FN	TN

Figure 28: the contingency tables to calculate validity rates

Once the contingency tables were completed, measures frequently used to describe the accuracy of Natural Language Processing systems are calculated as following:

Recall (sensitivity): R= TP/(TP+FN)

Precision (positive predictive value): P = TP/(TP+FP)

Weighted harmonic mean (F) of precision and recall is also calculated for all variables using the following formula: $F= ((\beta^2 + 1) P \times R) / ((\beta^2 \times P) + R)$

The F measure combines the precision and recall rates to create a single measure of performance for NLP algorithms. A β value of 1 gives equal weight to precision and recall (Balanced-F), and a value higher than 1 gives more weight to the recall.

We also used an Error measure that calculates the probability of any type of error (Type I or Type II error) in information retrieval:

Error: FP (Type I error) + FN (Type II error) / TP + FP (Type I error) + FN (Type II error)

It is important to note that in systems like BLUE-Text the TN measure will be extraordinarily high (due to large sources of vocabulary with hundreds of thousands, if not millions, of concepts), and use of the measures such as “Accuracy” and “Fallout” (as defined in below) will be misleading and may generate artificial results indicating a better than actual performance:

Accuracy: (TN+TP)/(TP+FN+FP+TN)

Fallout: (FP)/(FP+TN)

However, in case of evaluating reliability and validity of the method in dealing with Negations in the text where a single Boolean question is answered (is this fact negated?) these measures are applicable and meaningful.

The method described at [Newcombe RG. Improved confidence intervals for the difference between binomial proportions based on paired data. Stat Med 1998;17(22):2635-50] is often used to calculate the 95% confidence interval for the recall and precision measures.

Although the measures of reliability for extraction and encoding of concepts (e.g., recall and precision rates) also apply for evaluating the text understanding systems, measuring the

‘understanding’ is an ambiguous and controversial concept, and should be defined in clear and practical terms and agreed upon before proper evaluation methods could be conceptualized. One may define understanding in many different ways and by focusing on various aspects of the methodology such as: a) completeness of the knowledgebase produced; b) expressivity of the knowledge representation framework and the reasoning methodology; c) consistency and tractability of the knowledgebase; d) recall and precision of the method in identifying assertions about concepts extracted from the text; e) the ratio of extractions and encodings (assertions) to valid inferences and deductions that system can support; or f) success and failure rate of the system in mapping surface linguistic expressions from the text, to some semantic schemata.

To evaluate BLUE-Text reliability for both extraction and understanding tasks we decided to:

1- Calculate the recall and precision measures (± 95% confidence intervals) for extraction and encoding of important categories of concepts (clinical observations, anatomical concepts and modifiers). This would help compare performance of BLUE-Text with existing NLP system as it relates to the extraction and encoding of biomedical concepts from a clinical text.

2- Calculate the recall and precision measures (± 95% confidence intervals) for the representation of meaningful relationships between concepts extracted from the text. This can give us a measure of how successful the system was in extracting and explicating valid and computationally interpretable ‘facts’ from a given clinical text. The facts (statements about concepts) can be meaningful and accurate, only if they are about some meaningful and accurately extracted concepts, and if the relationship identified between them is supported by the gold standard. In another word, the subject, the object, and the predication between the subject and object should all be “True-Positive” for a fact to qualify as “True-Positive”.

Although other measures such as completeness, soundness, tractability, knowledge discovery, reasoning etc) are also valid measures for evaluating a text-understanding system, we argue that the recall and precision of the assertions generated by the system are the most fundamental performance indicators that can be controlled and objectively measured. All other measures primarily depend on the performance of the system in this level, and are abstractions taken from the assertions generated by the system. That is, other measures of performance should be interpreted, and can be understood only in light of the validity (Recall and Precision) of the assertions made by the system. Furthermore, designing an objective evaluation method for Knowledgebases is context and domain dependant and should be conceptualized in the scope of a well-defined problem and task, or the results may not be indicative of a predictable performance in other environments.

The evaluation method

To evaluate the reliability of the BLUE-Text we used 200 chief complaints and triage notes randomly selected from a pool of more than 70,000 encounters recorded by electronic health record systems of 8 different general hospitals from Houston Texas area. The method described at (Li and Fine 2004) was used to calculate the minimum sample size required to obtain an unconditional power no less than $1 - \beta = 0.9$, testing the $H_0 : Se = Se_0$ versus $H_1 : Se \neq Se_0$, with the significance level $\alpha = 0.05$, desired unconditional power $1 - \beta = 0.9$, and the desired sensitivity $Se_0 = 0.8$ in a one sample design.

Use of chief complaints and triage notes from different hospitals provides a test set with high degree of variability and irregularity often associated with unconstrained clinical text. Text shorter than 9 words were excluded from the study to make sure that enough context will be

available to evaluate the system for identifying and representing relationships between concepts. The average sentence length from all 200 entries was 14.8 words. Total number of sentences was difficult to identify, because some entries were represented as phrases without a complete sentence structure. In some other case multiple sentences were identifiable, but were not delimited appropriately. The maximum sentence length was 22 words (calculated only for complete sentences).

The gold standard was constructed through analysis and annotation of each of the 200 entries separately, by two independent subject matter experts (two internal medical specialists). The annotation process was mediated by specialized software that we specifically produced for this project to enforce the evaluation protocol and to ensure consistency and completeness of the annotations by subject matter experts. The software also eliminates possibility of bias introduced by interactions between the designers of the algorithm with subject matter experts annotating the test set, through enforcing a strict and consistent interaction and information collection model.

Each entry was rated for its quality by both annotators. An ordinal measure of quality (good, medium and bad) was used to indicate the quality of the text regarding its understandability as a whole, and without regards to the syntactic, or grammatical errors. A guideline was provided to make sure ratings of quality are consistent and meaningfully interpretable by raters. A good quality text according to the guideline was one that could be understood with only one time reading and did not include significant grammatical or structural aberrations. That is, the rater could answer all questions asked by software with high degree of certainty, and without having to read the text again. Medium quality text is one that is understandable with one time reading but suffers from notable structural or grammatical irregularities, or one that requires 2 times reading before answering all questions with high degree of certainty. Bad quality text is one that

suffers from significant lack of structure or grammatical errors, or requires 3 or more times reading to answer all questions, or answers to questions were not provided with high degree of certainty.

Annotators then separately examine each entry with respect to the following measures as illustrated in the Figure 29.

A. Extraction and encoding: To identify and list all concepts associated with

A.1 Clinical observations such as sign or symptom, disease or syndrome, trauma, poisoning or injury, health finding, pathological phenomenon, mental or behavioral dysfunction, therapeutic or preventive processes, diagnostics, clinical laboratory test or its result, acquired or congenital disorder, and anatomical deformity (item 1 Figure 29).

A.2 Concepts related to Anatomical relationships such as body sites and regions, anatomical structures, tissues and substances (item 2 Figure 29).

A. 3 Concept modifiers, qualifiers and descriptors such as qualitative and quantitative modifiers (item 3 Figure 29).

B. Representing the relationships between clinical observations and their associated body sites: The task involves explicating relationships between Observation and Locus concepts extracted in previous steps (item 4 Figure 29).

C. Representation of relationship between clinical Observations or body sites with qualitative and quantitative modifiers: The task involved explicating relationships between modifier concepts and the concept they described (item 5 Figure 29).

D. Negation: The task involved representation of the Negation. That is, if an observation, or a relationship between a body site and an observation were negated (item 6 Figure 29).

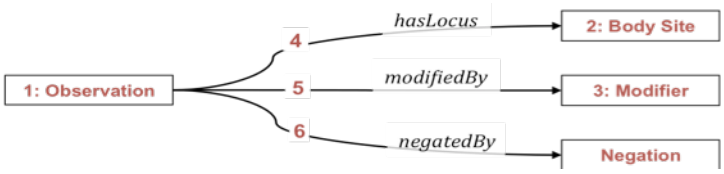


Figure 29: Evaluation items for the text-understanding algorithm.

It is important to notice that the items 1,2, and 3 are typically associated with the validity of the concept extraction and encoding, while items 4, 5, and 6 are associated with the validity of the text-understanding task, as they concern assertions about the concepts extracted from the text.

Positive Specific Agreement (PSA) measures were calculated to measure the inter-rater agreements. PSA is a measure of reliability of the gold standard and establishes a basis for comparing the human interpretation with the automated algorithm. It basically measures the rate in which human raters were in agreement about a positive conclusion (e.g., existence of a concept). A PSA equal to or above %75 is considered to be a reliable gold standard. A PSA below 75% indicates that more than often human experts could not agree with each other, which makes it difficult to interpret the significance of any agreement or disagreement between machine and gold standard..

Two-by-two contingency table

Rater A's judgment	Rater B's judgment		Total
	Positive	Negative	
Positive	a	b	$a+b$
Negative	c	d	$c+d$
Total	$a+c$	$b+d$	$a+b+c+d$

$$P_{pos} = \frac{2a}{2a+b+c}$$

$$P_{neg} = \frac{2d}{b+c+2d}$$

Table 3: Positive Specific Agreement and Negative Specific Agreement

Negative Specific Agreement (rate in which raters agree on negative information, such as lack of a concept) was applicable and calculated only for the Negation (item 6 Figure 29). After calculating the measures of reliability for the gold standard, a third subject matter expert (a physician) completed the annotation process by resolving disagreements of the two raters.

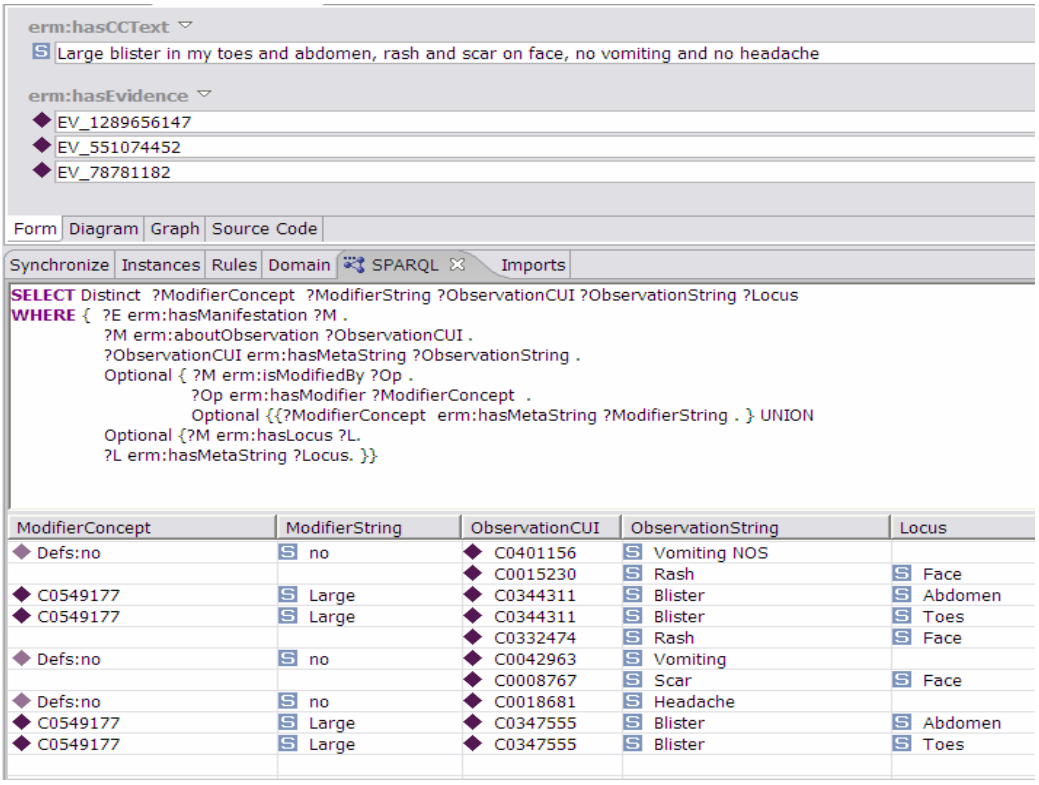


Figure 30: The SPARQL query to retrieve relevant information for evaluation

After running the algorithm on the test set, a single SPARQL query was executed on all outputs to retrieve information necessary for evaluation of the 6. The query is equivalent to an script that first extracts all clinical observations and their related UMLS codes, and then extracts the anatomical locations, modifiers and their negations for the extracted observation. Figure 30 demonstrates the example chief complaint used for this documentation, the SPARQL query and its results. As shown, each concept is accompanied by a human readable term, a CUI from

UMLS-KS. In this example, predicates such as :hasLocus, :negatedBy, or :modifiedBy are used to extract desired relationships for evaluation.

For each category outlined in the Figure 29 results were compared to the gold standard through calculation of the recall, precision and balanced-F measures. Figure 31 illustrates the overall evaluation method used to measure the performance metrics for the BLUE-Text. In the next section the results of the evaluation in the 6 areas mentioned above will be presented.

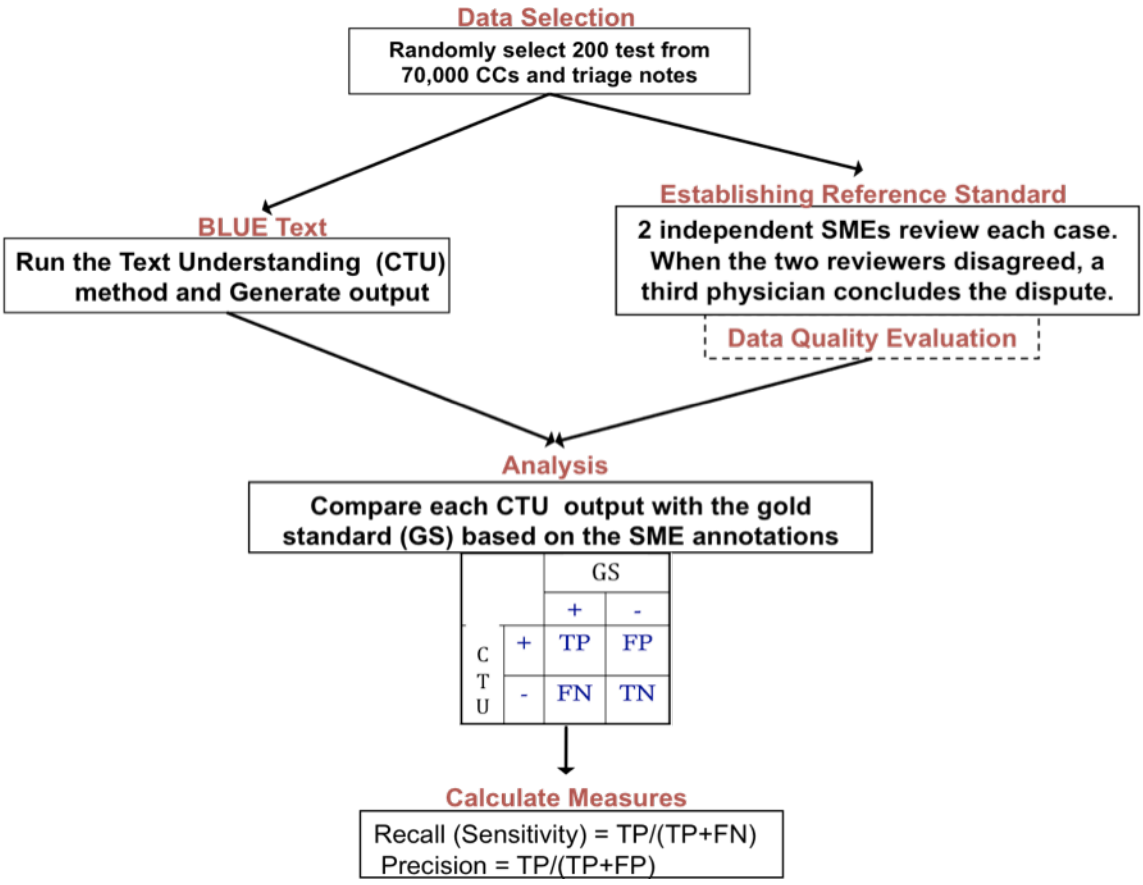


Figure 31: The evaluation process

Chapter Summary:

The Center for Biosecurity and Public Health Informatics Research has been interested in availability and utilization of robust clinical text understanding systems without regards to the

schema and structure of to build syndromic classification engines and data mining platforms for public health surveillance and preparedness. Recent CCTS program at the University of Texas Health Science Center at Houston calls for integrating unstructured clinical data with structured data and contextualization and repurposing of information to support multidisciplinary research. With this vision the center set the stage for conceptualization and implementation of a prototype text understanding system with resilience to grammatical and syntactic irregularity, consistent encoding of output to standard vocabularies, and a computationally interpretable and self descriptive output representation.

The prototype system design (here called BLUE-Text) employs a broad range of technological frameworks and standards from the Semantic Web community. BLUE-Text uses the Resource Description Framework (RDF) to represent all system knowledgebases, intermediate data, and the output. The Web Ontology Language (OWL) is used to construct and represent all knowledgebases as formal ontologies that incorporate semantic, syntactic and domain knowledge utilized by the system. A combination of Description Logic (DL) and rules reasoning is used by the semantic application for classification and inferences.

Figure 12 illustrates the process and the relationship between components of the BLUE-Text. In this design a set of ontologies representing syntactic, terminological, semantic and domain knowledge are used for all system processes including syntactic analysis, cocept mapping and semantic indexing, and output representation.

After a preliminary text preparation phase, the Minimal Syntactic Parser forms a parse graph that is comprised of tokens mapped to concepts from semantic and domain models, and indexed by a semantic indexer. An interpreter algorithm uses the semantic and domain knowledge, and the

parse graph, and constructs a conceptual graph that represents the text as a formal graph. Inferences based on the conceptual graph are used to normalize and reduce the variability of the output.

Measuring the reliability and validity of a 'text-understanding' system is an ambiguous and controversial concept, and should be defined in clear and practical terms and agreed upon before proper evaluation methods could be conceptualized. To evaluate BLUE-Text for both extraction and understanding tasks we calculated the recall and precision measures (± 95% confidence intervals) for extraction and encoding of important categories of concepts (clinical observations, anatomical concepts and modifiers) and the recall and precision measures (± 95% confidence intervals) for the representation of meaningful relationships between concepts extracted from the text (recall and precision of explicating meaningful assertions about concepts). This can give us a measure of how successful the system was in extracting valid and computationally interpretable 'facts' from a given clinical text.

A gold standard comprised of 200 chief complaints and triage notes, randomly selected from 8 different general hospitals and annotated (interpreted) by two internal medicine specialists, were used to calculate the above measures. Use of chief complaints and triage notes from different hospitals provides a test set with high degree of variability and irregularity often associated with unconstrained clinical text. The annotation process was mediated by specialized software that we produced to ensure consistency and completeness of the annotations by subject matter experts, and to enforce the evaluation protocol. Figure 31 illustrates the overall evaluation method used to measure the performance metrics for the BLUE-Text.

CHAPTER 4: RESULTS

Overview

In this chapter the results obtained from the comparison of the BLUE-Text output with the annotations made by the human subject matter experts (gold-standard) will be presented.

This chapter starts with an introduction to the test set (sample) used for this evaluation and will describe some of its characteristics, including the distribution of the Quality of Text variable. Then we will proceed with the following four sections:

Section (A) presents evaluation results indicating validity of the BLUE-Text algorithm for concept extraction and encoding. The recall and precision measures along with the ± 95% confidence intervals were calculated for extraction of the clinical observations (item 1 in Figure 29), body sites (item 2 in Figure 29) and modifier concepts (item 3 in Figure 29) and presented here.

Section (B) presents the recall and precision measures for extraction of meaningful and valid assertions and facts about the relationships between a) the clinical observations and body sites (item 4 in Figure 29); b) clinical observations and modifiers (item 5 in Figure 29); and c) negation of clinical observations (item 6 in Figure 29) along with the ± 95% confidence intervals.

Section (C) will follow by presenting performance statistics comparing the performance of the BLUE-Text with different qualities of clinical text. The Recall and precision measures along with their ± 95% confidence intervals for extraction of clinical observations (item 3 in Figure 29), and assertions about their anatomical sites (item 4 in Figure 29), modifiers (item 5 in Figure

29), and negation (item 6 in Figure 29) will be presented comparatively for categories of thte quality of text (good, medium and bad).

Section (D) presents results from evaluation of the validity of the BLUE-Text to retrieve information from clinical text using the UMLS-SN Semantic Types as the sole source of domain knowledge and without using concepts originated from the BLUE-Text semantic knowledge. This information would indicate how complete (useful, and appropriate) the UMLS-SN and the UMLS-MTH are, were to be used as sole source of domain knowledge for clinical text-understanding out-of-the-box and without extensions and corrections similar to ones made by the BLUE-Text semantic knowledge.

For all categories of results, the Balanced-F (a measure of overall accuracy) and Positive Specific Agreement (PSA, a measure of inter-rater agreement for the gold standard) are also presented when appropriate.

The Test Set (Sample)

The gold standard used for this evaluation consisted of a randomly selected set of entries into the chief complaints and triage notes sections of 8 different electronic health records systems. The only exclusion criteria used for the selection process was the text length less than 9 words. Definition of a word was any set of characters separated by any of the characters described as “Separator” in the Syntactic Knowledge (e.g., “space”, “comma”, forward or backward “slash” etc). For each entry (test case) a consensus was made by both SME regarding the quality of the text based on the guideline described in the Chapter 3 (Evaluation).

The Table 4 presents the distribution of the good, medium and bad quality text in the gold standard, and the frequency of concepts and facts extracted from them.

Table 4: Quality of text and the distribution of extracted concepts and facts

Variable	All N (%)	Good Quality N (%)	Medium Quality N (%)	Bad Quality N (%)
Test Cases	**200**	**36 (18%)**	**123 (61.8%)**	**41 (20.5%)**
All Extractions:	**1798(PSA=86.8%):**	**66 (3.7%):**	**1510 (83.9%):**	**222 (12.4%):**
Observation Concept	639 (35.5%)	23 (3.6%)	543 (85%)	73 (11.4%)
Modifier Fact	171 (9.5%)	3 (1.8%)	128 (74.8%)	40 (23.4%)
Locus Fact	537 (29.9%)	23 (4.3%)	458 (85.3%)	56 (10.4%)
Negation Fact	451 (25.1%)	17 (3.8%)	381 (84.5%)	53 (11.7%)

The majority of test cases in the gold-standard (62%) were of Medium Quality text as identified by subject matter experts, incorporating 84% of all concepts and facts extracted by the system.

Good quality text and bad quality text composed 18% and 21% of the gold standard respectively. However, test cases with Bad Quality included more than 3 times more concepts and facts to be extracted by the subject matter experts as relevant to this evaluation (222 (3.7%) vs. 66 (12.4%))

The overall Positive Specific Agreement (PSA) between the human experts used as raters in this evaluation was calculated as 86.8%. This is an indicator of a high overall quality of the gold-standard used for this evaluation. However, a more precise PSA is also calculate based on different categories of evaluation and reported in the following sections of this report.

A) Extraction and Encoding

Frequently, NLP techniques are used to identify, and extract important concepts (e.g. diseases, procedures, medications, etc.) from the clinical text (***extraction***), and encode them into a corresponding terminology system (***encoding***). Extraction and encoding are two distinct functionalities that can be evaluated separately. That is, not every extraction system supports encoding, and not every extracted concept can necessarily be encoded by a controlled vocabulary. The reliability and validity of the encoding may depend to the performance of

mapping function, completeness and richness of the vocabulary systems used for encoding and the coverage and appropriateness of the vocabulary to the domain and the clinical content.

The BLUE-Text uses a text extraction and encoding algorithm that identifies tokens of text that can be successfully mapped into the UMLS-MTH Concepts. That is, by design BLUE-Text only extracts concepts it can find a perfect map (scored 1000 by MMTx mapping algorithm) to at least one UMLS-MTH concept. All extractions with less than a perfect mapping score are disregarded from the BLUE-Text output, even if in some circumstances they may represent a valid concept. This behavior is rooted in the fact that BLUE-Text uses the UMLS as its source of vocabulary, and is bound by its completeness to identify segments of text that can be best interpreted by a concept from biomedical and clinical terminologies.

Hence the process of extraction and encoding in BLUE-Text are integrated into a single operation and can be evaluated as such. There is another extraction function supported by the Syntactic Knowledge that is responsible for extracting non-biomedical and syntactic elements of the clinical text. However, results of this extraction function cannot be mapped into biomedical and clinical vocabularies (as they are by definition non-medical concepts) and reporting the system performance on these items, are not relevant, important or interesting for the biomedical community.

Table 5 presents the results of the evaluation of BLUE-Text for extraction and encoding in the categories related to observations, body sites (Locus) and modifiers.

1. Clinical Observations

Clinical observation is a concept from the semantic model that serves a super class for all biomedical concepts that represent signs and symptoms (e.g., fever), health findings (e.g.,

drowsiness), disease or disorders (e.g., diabetes), trauma, injury or poisoning (e.g., burn), anatomical disorders (e.g., cleft palate), or any diagnostic (e.g, radiography), or therapeutic (e.g., kidney transplant) procedure or process including administration of medications (e.g., Digoxin). The BLUE-Text demonstrates recall rate of 78.6% (75.33% 81.88%) and precision rate of 92.9% (90.72% - 95.16%) for extraction and encoding of clinical observations (F=85.2%).

Table 5: BLUE-Text recall and precision rates for extraction and encoding

	N[ץ]	PSA	Recall (confidence int.)	Precision (confidence int.)	Balanced F
Observation Extraction	639	91.5%	78.6% (75.33% 81.88%)	92.9% (90.72% 95.16%)	85.2%
Locus Extraction	304	94.9%	94.18% (91.49% - 96.86%)	95.82% (93.50% 98.13%)	94.99%
Modifier Extraction	120	62.3%	92.86% (86.82% 98.89%)	56.52% (47.46% 65.58%)	70.27%

The semantic model also includes high level specializations of clinical observation that serves as a meta-class for other applications to specialize and extend the model for their needs. This includes a class. The *Health Finding* class represents all concepts with a clinical significance but not necessarily with a pathological connotations (e.g., tingling, dry skin, etc). Disorders represents all observations with a potential pathological connotation (syndromes, disease, signs and symptoms, injuries, etc). the Health Procedure class represents all therapeutic, preventive, or diagnostic procedures (activities with an objective). Table 6 presents a breakdown of the overall recall and precision rates for extraction of concepts from each of these categories.

The best performance is observed when applied to the Disorders (F= 91.6%) whereas the BLUE-Text performance is marginally good when applied to extract Procedures. A more detailed discussion of the performance results is provided in the chapter 5.

[ץ] N: total number of concepts or relationships extracted from the 200 test cases

Table 6: BLUE-Text recall and precision for Subclasses of Clinical Observation

	N	PSA	Recall (confidence int.)	Precision (confidence int.)	Balanced F
Health Findings	404	96.0%	81.1% (77.18% 84.99%)	94.6% (92.12% 97.00%)	87.3%
Disorders	136	95.8%	85.2% (79.19% 91.18%)	99.1% (97.46% 100%)	91.6%
Healthcare Procedures	70	72.3%	73.7% (62.25% 85.12%)	76.4% (65.14% 87.59%)	75.0%

2. Body Site (anatomical concepts) Extraction

Locus is a concept from the semantic model that serves as a super class for all anatomical concepts including but not limited to the body locations or regions (e.g. abdomen), anatomical structures (e.g., skull), body systems (e.g. respiratory tract), organs, tissues and body fluids (e.g. tear, blood).

The BLUE-Text demonstrates recall rate of 94.18% (91.49% - 96.86%) and precision rate of 95.82% (93.50% 98.13%) for extraction and encoding of the anatomical concepts (F=94.99%). This is nearly as good as human expert.

3. Modifier Concept (Qualitative and Quantitative) Extraction

The Modifier concept in the BLUE-Text semantic model is a super concept for all concepts that describe other concepts in terms of their quantities (qualitative modifiers such as *severe* in "*severe cough*") and quantities (quantitative modifiers such as *high* in "*high* blood glucose").

The BLUE-Text demonstrates recall rate of 92.86% (86.82% 98.89%) and precision rate of 56.52% (47.46% 65.58%) for extraction and encoding of modifier concepts (F=70.27%).

Although the balanced F measure is acceptable the low precision is an indication of problems rooted in the system knowledgebase.

B) Text Understanding

Text understanding refers to a category of NLP technologies that extend extraction and encoding capabilities by representation of the semantic relationships between concepts extracted from text (Hahn and Schnattinge 1997; Hahn and Romacker 1997). In another word a text-understanding system may be evaluated based on the extraction of the assertions about concepts from the text.

Table 7 presents the results of our formal evaluation of BLUE-Text for the extraction of assertions about clinical observations regarding a) anatomical sites when present; b) qualitative or quantitative modifiers when present; and c) negation of an observation when present. Table 7 reports standard measures of Specificity and Accuracy for Negation (see Chapter 3), rather than the Precision and F measures.

Table 7: BLUE-Text recall and precision rates explication of semantic relationships

	N	PSA	Recall (CI)	Precision (CI)	Balanced F
Body Site	537	86.0%	79.4% (75.36% - 83.41%)	67.4% (63.10% - 71.69%)	72.9%
Modifiers (Qualitative, Quantitative)	131	77.6%	78.7% (69.40% - 87.94%)	51.3% (42.17% - 60.44%)	62.1%
Negation	451	98.2% Kappa: 83.5	97.4% (92.48% -100%)	Specificity: 97.6% (92.8% -100%) Accuracy: 97.3% Fallout: 2.7%	

For an assertion, for example between an observation and its locus, to be listed as true positive in this context, the system needs to produce 3 true positive extractions: 1: a true positive Observation, 2: a true positive *Locus* and 3: a true positive semantic relationship between the

Locus and the Observation. An error in extraction of any of these concepts will result in an error in the assertions produced by the system.

4. Anatomical sites of Clinical Observations

Clinical Observations may have been associated with some *Locus* (anatomical sites, organs or tissue, etc) within the clinical text. The task of a text-understanding system in these cases is to extract the relationships between the observation and the body site as an explicit statement that is semantically in agreement with what is presented in the text.

The BLUE-Text demonstrates recall rate of 79.4% (75.36% - 83.41%) and precision rate of 67.4% (63.10% - 71.69%) for explicating relationships between Locus and Clinical Observations concepts (F=72.9%).

Table 8: BLUE-Text explication of assertions about categories of Locus

	N	PSA	Recall (CI)	Precision (CI)	Balanced F
Anatomical Concepts	494	87.7%	78.2% (73.86% 82.44%)	67.1% (62.55% 71.58%)	72.2%
Histological Concepts (Tissues)	34	95.8%	100.0% (95% 100%)	67.6% (51.92% 83.37%)	80.7%

Table 8 provides a break down on the performance of the BLUE-Text on explication of relationships between *Clinical Observations* and their *Locus*. The recall rate is meaningfully higher (the CIs don't overlap) for identifying relationships with tissues and body fluids, than with body parts and anatomical structures, while the precision rate remained the same. .

5. Quality and Quantity Modifiers of Clinical Observations

The meaning and context of a Clinical Observation may be further refined and described by *Modifier* concepts within the text. The task of a text-understanding system in this case is to explicate a semantically equivalent statement about the observations extracted from the text and their modifiers.

The BLUE-Text demonstrates recall rate of 78.7% % (69.40% 87.94%) and precision rate of 51.3% (42.17% - 60.44%) for explicating relationships between Modifier and Clinical Observation concepts (F= 62.1%).

6. Negation of Clinical Observations

Negation is basically an assertion about quality of a concept, that is, Negation is a *Modifier* and this is how semantic model conceptualizes Negation. However, BLUE-Text uses a custom lexicon from its Semantic knowledgebase to identify negations, mainly because the negation is not covered consistently by any biomedical terminology system, including UMLS-KS. Hence, the evaluation of Negations is not incorporated in the overall evaluation of the Modifier concepts (that are highly affected by the quality of the UMLS knowledge source) and is measured and reported separately. According to the BLUE-Text semantic model, *Locus* and *Clinical Observations* are both *Negatable Concepts.* Hence the model sanctions assertions such as 'patient has no rash', or 'patient has rash on the face but not on hands or abdomen'.

However, as mentioned in chapter 3, for this evaluation we use precision metrics for all evaluations but Negations. A specificity metric along with other measures such as Accuracy and Fallout are only evaluated for the Negation of Clinical Observations (item 6 of the Figure 29).

The BLUE-Text demonstrates recall rate of 97.4% (92.48% 100%) with specificity of 97.6% (92.82% 100%) for explicating negations of Clinical Observations concepts (Accuracy= 97.3%), a near perfect match with human subject matter experts (observed inter-rater agreement: 98.2%, κ=83.5%).

C) Consistency of extraction by quality of text

The performance of the extraction and encoding algorithm across qualitatively different content has been evaluated. As the BLUE-Text algorithm uses a minimal syntactic algorithm for parsing and syntactic analysis of text, the hypothesis is that the performance of the algorithm will not be affected by overall syntactic and grammatical quality of the text. That is, the recall and precision rates for the good, medium and bad quality text will be consistently the same.

A good quality text in this context is one that could be understood by the human expert with one time reading, the medium quality text is defined as the one that requires two times reading by human experts to be understood, and the bad quality text is one that requires more than 2 times reading to be understood, or if human expert could not answer questions about the meaning of the text with high degree of certainty.

7. Quality of text and Extraction of Clinical Observations

The Table 9 presents the recall and precision rates for extraction of the clinical observations from clinical content.

The statistical comparison of the recall and precision rates between the three categories of text does not show a significant difference (P>0.8). Both recall and precision rates for extraction of the clinical observation are consistent as the quality of the text changes.

Table 9: BLUE-Text extraction of clinical observations by the quality of text

		Good quality text	Medium quality text	Bad quality text	All Text
Clinical Observation	N	23	543	73	639
	PSA	93.0%	91.5%	88.9%	91.5%
	Recall (CI)	86.4% (72.02% - 100%)	77.9% (74.34% - 81.52%)	81.2% (71.9% - 90.4%)	78.6% (91.49% - 96.86%)
	Precision (CI)	95.0% (85.45% - 100%)	92.8% (90.35% - 95.24%)	93.3% (87.02% - 99.65%)	92.9% (93.50% - 98.13%)
	Balanced F	90.5%	84.7%	86.8%	85.2%

8. Quality of text and assertions about Locus of the Clinical Observation

Table 10 presents the recall and precision of the BLUE-Text in explicating the relationships identified between the Locus and Clinical Observations in qualitatively different text.

Table 10: Quality of text and extraction of the anatomical relationships

		Good quality text	Medium quality text	Bad quality text	All Text
Clinical Observations - Locus Assertions	N	23	458	56	537
	PSA	100.0%	86.8%	74.2%	86.0%
	Recall (CI)	89.5 (75.67% - 100%)	77.2% (72.67% - 81.76%)	92.5% (84.34% - 100%)	79.4% (75.36%, 83.41%)
	Precision (CI)	81.0 % (64.16% - 97.8%)	66.3% (61.59% - 71.09%)	69.8% (57.45% - 82.17%)	67.4% (63.10%, 71.69%)
	Balanced F	85.0%	71.3%	79.6%	72.9%

The statistical comparison of the recall and precision rates between the three categories of text does not show a significant difference (P>0.1). Both recall and precision rates for explication of

the relationships between Clinical Observation and Locus are consistently the same as the quality of the text changes.

9. Quality of text and negation of Clinical Observations

Table 11 presents the recall and precision of the BLUE-Text in explicating the negation of Clinical Observations in the clinical content by different quality of text.

It is important to note that none of the sentences in the test case were categorized as “good” quality text, when there was a negation in the text. That is all sentences with a negation, were categorized as medium quality or bad quality text by human experts without an exception. As a result the recall rate was not calculated for the system in this category, since neither the gold standard, nor the BLUE-Text identified a negated observation in this category. With the same token the specificity of BLUE-Text in identifying negated Observations in good quality text was 100%, due to the fact that the system agreed with the gold standard in that 100% of the observations were true negative in respect to negation.

Table 11: Quality of text and Negation of Clinical Observations

		Good quality text	Medium quality text	Bad quality text	All Text
Negated Clinical Observations	N	23	381	53	457
	Inter-rater Reliability	94.1%	98.4%	98.4%	98.1%
	Recall (CI)	N/A (0% true positive)	97.2% (91.8% - 100%)	100% (0% false negative)	97.4% (92.48% 100%)
	Specificity (CI)	100 % (100% true negative)	96.8% (91.7% - 100%)	100% (0% false positive)	97.6% (92.82% 100%)
	Accuracy	100%	96.85%	100%	97.3%

The statistical comparison of the recall and precision rates between the three categories of text does not show a significant difference. Both recall and precision rates for explication of the negated Clinical Observation are consistently the same as the quality of the text changes.

D) BLUE-Text and Information Retrieval using UMLS-SN

BLUE-Text uses the UMLS-KS as the source of vocabulary and the lexicon to identify the biomedical and clinical concepts and their qualitative and quantitative modifiers in the text. The UMLS-SN and the UMLS-MTH are also used as the part of the domain knowledge that extends the BLUE-Text semantic model.

Evaluations presented in the previous sections were made from the perspective of BLUE-Text semantic model and after reasoning, classifications and concept mapping processes infer class membership for all extractions based on the domain knowledge and semantic knowledge provided to the system.

However a different query to the BLUE-Text output, this time from the perspective of the UMLS-SN could retrieve another set of results, shedding some light on the appropriateness of the UMLS-SN as is as the semantic model for text understanding systems such as BLUE-Text. This information can be used to identify knowledge gaps and potential errors associable to the UMLS-SN and UMLS-MTH as the source of domain knowledge for the text-understanding systems.

In this section we present results from evaluation of the validity of the BLUE-Text to retrieve information from clinical text using the UMLS-SN Semantic Types as the sole source of domain knowledge and without using concepts originated from the BLUE-Text semantic knowledge. This information would indicate how complete (useful, and appropriate) the UMLS-SN and the

UMLS-MTH are, were to be used as source of knowledge for clinical text-understanding out-of-the-box and without extensions and corrections similar to ones made by the BLUE-Text semantic knowledge.

10. UMLS-Semantic Types and retrieval of the Clinical Observations

BLUE-Text generates a formal output linked with ontologies that could be used for information retrieval and ad-hoc contextualization. UMLS-SKOS includes a partial OWL representation of the UMLS-SN that can be used to retrieve concepts extracted from the clinical text based on the UMLS Semantic Types.

Table 12 presents the results of the evaluation for reliability and validity of the information retrieval from BLUE-Text output, using the UMLS-SN and the UMLS Semantic Types as the domain model to extract clinical observations.

Table 12: BLUE-Text retrieval of Clinical Observations by UMLS Semantic Types

	N	PSA	Recall (confidence int.)	Precision (confidence int.)	Balanced F
umls:Sign_or_Symptom	287	97.8%	82.9% (78.52% - 87.32%)	97.5% (95.51% - 99.47%)	89.6%
umls:Finding	124	91.5%	77.7% (69.97% - 85.39%)	87.9% (81.45% - 94.31%)	82.5%
umls:Disease_or_Syndrome	49	94.6%	81.6% (70.79% - 92.47%)	100% (100, 100%)	89.9%
umls:Injury_or_Poisoning	48	94.5%	85.1% (74.93% - 95.28%)	97.6% (92.84% - 100%)	90.9%
umls:Pathological_Phenomenon	41	98.8%	90.2% (81.16% - 99.33%)	100%	94.8%
umls:Therapeutic_or_ Preventive_Procedure	39	76.9%	66.7% (49.80% - 83.54%)	69% (52.13% - 85.80%)	67.8%
umls:Diagnostic_Procedure	14	63.2%	66.7% (39.99% - 93.34%)	80% (55.21% - 100%)	72.7%
umls:Health_Care_Activity	13	53.3%	90.9% (73.92% - 100%)	83.3% (62.25% - 100%)	86.9%
umls:Laboratory_Procedure	5	100.0%	100%	100%	100.0%

The weakest performance is observed in the “Therapeutic or Preventive Procedure” and “Diagnostic Procedure” categories, and is attributable to an incomplete lexicon used for representing shorthand and acronym form for the procedures. It is also important to note that the reliability of the gold standard regarding to the “Diagnostic Procedures” is also questionable. A rather wide confidence interval along with the less than reliable gold standard in this category is most probably due to a variability of representation of the diagnostic procedures in the clinical text and small sample size in this category (n=14). Rows with gray background in the table 12 present Semantic Types their evaluation results are not reliable due to small sample size or low gold standard reliability (Lower than 75% inter-rater agreement).

11. UMLS-Semantic Types and retrieval of Locus assertions

Table 13 presents the results of the evaluation for reliability and validity of the information retrieval from BLUE-Text output, using the UMLS-SN and the UMLS Semantic Types as the domain model to extract assertions about the anatomical associations of clinical observations.

Table 13: BLUE-Text retrieval of anatomical assertions by UMLS Semantic Types

	N	Gold Standard Reliability	Recall (CI)	Precision (CI)	Balanced F
umls:Body Part, Organ, or Organ Component	315	87.8%	90.3% (86.39% - 94.26%)	66.7% (61.28% - 72.06%)	76.7%
umls:Body Location or Region	298	87.1%	72.7% (66.94% - 78.60%)	68.8% (62.88% - 74.68%)	70.7%
umls:Tissue	34	95.8%	100%	67.6% (51.92% - 83.37%)	80.7%

There is an important and significant difference in recall rates observed within the three categories associated with the anatomical concepts extracted from the text (gray background), as the confidence intervals of the 3 categories show no overlap.

12. UMLS-Semantic Types and Modifiers of the Clinical Observation

Table 14 presents the results of the evaluation for reliability and validity of the information retrieval from BLUE-Text output, using the UMLS-SN and the UMLS Semantic Types as the domain model to extract assertions about the qualitative and quantitative modifiers of clinical observations.

Table 14: BLUE-Text retrieval of modifier assertions by UMLS Semantic Types

	N	Gold Standard Reliability	Recall (CI)	Precision (CI)	Balanced F
umls:Qualitative Concept	94	82.8%	75.9% (64.85% - 86.88%)	55% (44.10% - 65.90%)	63.8%
umls:Quantitative Concept	38	75.0%	83.3% (66.12% -100%)	42.9% (2646% - 59.25%)	56.6%

No significant or meaningful differences exist between performance in different semantic types.

Chapter Summary:

Table 15 presents a summary of the results obtained from the comparison of the BLUE-Text output with the gold-standard annotated by subject matter experts. The overall recall and precision measures were calculated for extraction of the clinical observations, body sites and modifier concepts, and then compared with the recall and precision in good, medium, and bad quality text.

The recall and precision measures for valid assertions (extraction of facts) about the clinical observations and body sites, clinical observations and modifiers, and the negation of clinical observations, in overall examination and with different quality of text are also presented in this table.

Table 15: BLUE-Text overall evaluation results and compared by quality of text

Variable	Overall						Good Quality Text						Medium Quality Text						Bad Quality Text					
	N	Reliability	Recall	Precision	F-Measure	Error	N	Reliability	Recall	Precision	F-Measure	Error	N	Reliability	Recall	Precision	F-Measure	Error	N	Reliability	Recall	Precision	F-Measure	Error
Observation	639	91.5%	78.6%	92.9%	85.2%	25.8%	23	93.0%	86.4%	95.0%	90.5%	17.4%	543	91.5%	77.9%	92.8%	84.7%	26.5%	73	88.9%	81.2%	93.3%	86.8%	23.3%
Health Findings	404	96.0%	81.1%	94.6%	87.3%	22.5%	14	92.3%	85.7%	100.0%	92.3%	14.3%	345	96.4%	80.5%	94.0%	86.7%	23.5%	45	94.0%	84.1%	97.4%	90.2%	17.8%
Disorders and Pathological Processes	136	95.8%	85.2%	99.1%	91.6%	15.4%	5	100.0%	100.0%	100.0%	100.0%	0.0%	107	96.1%	84.1%	100.0%	91.4%	15.9%	24	93.3%	87.0%	95.2%	90.9%	16.7%
Healthcare Procedures (Diagnostic or Therapeutic)	70	72.3%	73.7%	76.4%	75.0%	40.0%	2						63	72.1%	73.6%	79.6%	76.5%	38.1%	4					
Modifier	171	65.2%	68.2%	42.0%	52.0%	64.9%	3						128	62.7%	71.0%	38.3%	49.8%	68.0%	40	71.4%	73.9%	50.0%	59.6%	57.5%
umls:Qualitative Concept & umls:Quantitative Concept	131	74.6%	78.7%	51.3%	62.1%	55.0%	3						100	78.6%	74.1%	46.5%	57.1%	60.0%	28	75.0%	94.4%	63.0%	75.6%	39.3%
Locus	537	86.0%	79.4%	67.4%	72.9%	42.6%	23	100.0%	89.5%	81.0%	85.0%	26.1%	458	86.8%	77.2%	66.3%	71.3%	44.5%	56	74.2%	92.5%	69.8%	79.6%	33.9%
Anatolmical Concepts	494	87.7%	78.2%	67.1%	72.2%	43.5%	21	100.0%	88.2%	78.9%	83.3%	28.6%	417	88.5%	75.7%	66.0%	70.5%	45.6%	56	77.4%	92.5%	69.8%	79.6%	33.9%
Histological Concepts (Tissues)	34	95.8%	100.0%	67.6%	80.7%	32.4%	2						32	95.5%	100.0%	65.6%	79.2%	34.4%						
Negation	451	90.2%	97.4%	77.6%	86.4%	2.7%	17	0.0%				0.0%	381	92.1%	97.2%	76.1%	85.4%	3.1%	53	80.0%	100.0%	100.0%	100.0%	0.0%
Negation Specificity		97.6%						100.0%						96.8%						100.0%				
Negation:Negative specific agreement		99.0%						97.0%						99.1%						99.0%				
Negation:Observed agreement		98.2%						94.1%						98.4%						98.1%				
Negation: Kappa		83.5%						94.1%						82.0%						91.0%				
Negation: Accuracy		97.3%						100.0%						96.9%						100.0%				
Nagation: Fallout		2.7%						0.0%						3.2%						0.0%				

CHAPTER 5: DISCUSSION

Introduction

Understanding and Text-understanding; the Definition: Using "Understanding" as a term to describe a feature or behavior of a computational algorithm and subsequently trying to measure it may be problematic without having an unambiguous, and clear definition and an objective, reproducible and reliable metric to evaluate it. However, the term itself is an ambiguous one and loaded with connotations from cognitive science, human-computer interaction, artificial intelligence, psychology, behavioral science, and neuroscience. Meanwhile the literatures on the subject of text understanding do not provide a clear definition or rationale for the use of term or objective measures of its assessment and evaluation. In many cases the term 'text-understanding' is being used loosely and interchangeably with other terms such as language processing or text parsing.

For this work we would like to use a definition for the term "Understanding" rooted from semiology. Semantically the term understanding refers to the ability to identify the relationships between concepts, and between the concepts and entities they refer to. Syntactically the term understanding refers to the ability to explicate relationships between entities in a formal and computationally interpretable way. Pragmatically the term understanding relates to the ability to provide a form of output representation (syntactic) that enables or improves practical utilities in a clinical domain. With these definitions in mind we refer to the 'text-understanding' as the ability (of any given system) to retrieve the semantics of a given text, that is, to be able to retrieve facts regarding the objects and concepts present in the text (diseases, bacteria, anatomical entities) and relationships between them (skin infection caused by Pseudomonas aeruginosa in a diabetic patient), and to create a form of formal output representation (syntactic) that is computationally

interpretable for queries and retrieval, and can be contextualized and reused on demand (pragmatic).

With this definition then the task of an objective evaluation is to scrutinize the outputs of a text-understanding system from the following 3 perspectives:

Semantics: Whether or not the input and the output of the system have identical meaning. In another word, can one answer all questions from the input text, by using the output instead?

Syntactics: Does output representation is formally and structurally appropriate to enable effective information retrieval and information interpretation by computer programs? In another word, can a computer agent (in lieu of a human expert) utilize the output (in lieu of the input text) to answer all questions pertinent to the text?

Syntactic evaluation may also address ability for communication, information sharing, reuse and contextualization in a distributed and multidisciplinary environment, as these categories of features depend extensively on using automated processes for interpretation, transformation and exchange of pertinent information.

The theoretical significance of establishing a text understanding system in Semantic and Syntactic levels is that it enables construction of information systems that can interpret clinical text identical to human experts. That is, one can expect (theoretically) that it can provide similar answers to questions from text. However the practical applications and utility (impact, cost effectiveness) can be understood only by a pragmatic evaluation of such a system.

Pragmatics: Pragmatic is concerned with outcomes of a text understanding system from the perspective of a user or a community of users, and the question is “what are the efficiencies of

using text understanding system in a specific use case"? In other words what is the utility of using such a system in solving a given practical problem? Examples of such practical problems would be using text-understanding systems for Syndromic case finding in public health preparedness, or in identifying malignant breast tumors in chest X-Ray or mammography reports.

BLUE-Text Evaluation Paradigm:

The minimal syntactic, semantic process for clinical text understanding and extraction introduced in this documentation (BLUE-Text) sets out to provide a method of transforming unconstrained and free text information into a computer interpretable (formal) representation that can be used for automation, querying (search and retrieval), information integration, contextualization, reuse, and sharing in a distributed and multidisciplinary environment. Furthermore, the conceptualization of the BLUE-Text algorithm intends to provide a method of clinical text understanding that is self-descriptive (incorporates all concepts and associations necessary for a secondary computational algorithm to meaningfully interpret the output), is domain independent (generic) but extensible to specialized use cases on-demand, and is measurably resilient to the grammatical and structural irregularities frequently found in the clinical content. However BLUE-Text project has not aimed at improving or enabling a clinical or health related use-case in particular.

That is, the focus of this project so far, has been conceptualization, implementation and evaluation of a system that can produce appropriate output with the Semantics and Syntactic characteristics consistent with a typical text understanding system described above. Discussion of the pragmatic utilities and value propositions of such a system requires more extensive and task specific evaluation of the system in a well-defined and controlled clinical environment and

is out of the scope of this work. Therefore, the focus of our attention is to investigate and evaluate the BLUE-Text output from the standpoint of its semantic and syntactic validity.

The evaluation method and results presented in the chapters 3 and 4 of this document are aimed at providing evidence that can be used for assessing performance and validity of the BLUE-Text algorithm in meeting the semantic and syntactic characteristics of a valid and useful NLP output.

This chapter will review the results provided in the chapter 4 in the context of the BLUE-Text project objectives, that is implementing a system that can demonstrate Semantic and Syntactic characteristics of a text-understanding system. We will also use the desiderata and the gap analysis put forward in the Chapters 1 and 2 as the guideline for this evaluation.

This chapter starts with the discussion of the characteristics and quality of the gold-standard. This sets the stage for future discussions and helps better understand the upcoming analysis. We will then continues with the following 4 sections:

Section A evaluates BLUE-Text from the standpoint of the validity of the overall extraction algorithm. We will discuss the performance of the BLUE-Text in extracting the biomedical concepts from text, the performance of the BLUE-Text in extracting the facts and the assertions about biomedical concepts, and the resilience of the BLUE-Text algorithm to the quality of the clinical text. The design rationale and their implications on the observed performance will also be discussed in this section. Section A in essence focuses on evaluating the output of the BLUE-Text from a Semantic perspective, that is, we evaluate the agreement of the output of BLUE-Text with its input, using annotations and interpretations of human experts.

Section B will discuss the syntactic and representational characteristics of the formal output generated by the BLUE-Text in light of the desiderata for an optimal NLP output representation and requirements to satisfy Syntactic characteristics of an optimal text-understanding system output. A comparative analysis of the BLUE-Text and MedLEE output will be provided at the end to illustrate the differences and the contributions of the BLUE-Text algorithm. Section B intends to discuss the validity of the BLUE-Text output from a syntactic point of view with an eye on the potential pragmatic value propositions of producing a semantic (formal and explicit) and ontology driven output as opposed to relying on the information and data structures (XML, Templates, etc) without formal semantics.

The outcomes of this work have been greatly influenced by the fact that we have extensively used the UMLS-KS as the source of knowledge and terminology for clinical content. Hence, the Section C is devoted to an examination of the current state of the UMLS-KS as a candidate knowledge source for conceptualization and implementation of clinical text understanding systems. The results of the formal evaluation presented in the Chapter 4 (Section D) will be used to highlight some of the pitfalls and shortcomings of the UMLS-KS as well as potentials and value propositions for using UMLS –KS to extract and construct reliable knowledgebases for clinical text-understanding systems.

Section D will discuss the limitations, shortcomings and known issues with the conceptualization, design and current implementation of the BLUE-Text algorithm. This chapter will lay the ground to suggest solutions to overcome some of the shortcomings of the BLUE-Text algorithm and its future extensions and improvements as discussed in the upcoming chapter (Chapter 6). As the pragmatic evaluation of BLUE-Text has not been a focus of this work, we

will provide brief and introductory information and value propositions about the Pragmatic values of systems like BLUE-Text in Chapter 6.

A chapter summary at the end will recapitulate on the main points raised throughout the chapter and transition to the next chapter.

The Gold-Standard: Good, Medium, Bad Quality of Text

Outlining a objective and measureable criteria for the quality of clinical content has been a big challenge for this project. Some immediate ideas included calculating a score based on typographic and syntactic errors (e.g. misspellings and typos, grammatical errors), or following a best practice guideline (e.g., short sentences, proper use of hyphenation, etc). However, grammatically correct does not always mean meaningful or understandable. An utterance might be meaningless or ambiguous without having a grammatical or structural error (e.g. “The fracture in blood was observed by Aspirin”). Furthermore, not all syntactic or structural errors and irregularities have a similar effect on the quality of the text and its interpretation. For instance, a sentence could be unambiguously meaningful and understandable (“Pt a 26 yrs old Af/Am fem. w/ n/v/d, Hx of DM x 3 years.”) despite frequent syntactic or grammatical irregularities. Therefore a scoring system solely based on the grammatical and structural cues may not be appropriate for measuring the quality of the text.

For this project we used a rather subjective criterion that aimed at capturing problems with understandability of text including but not limited to those related to the syntax and grammar. Consensus was made by three human experts about the quality of all test cases in the gold-standard based on the following criteria. The criteria designates each entry into one of the Good Quality, Medium Quality and Bad Quality categories based on both readability (syntactic and

grammatical quality) and overall understandability of the text. A Good Quality text did not include significant grammatical or structural aberrations that hinded reading or understanding, hence it is understandable when reading only once, and the rater could answer questions with a high degree of certainty. A Medium Quality text could be understandable after reading once but suffered from structural and/or grammatical irregularities that hinder reading. It could also be any grammatically correct entry that required reading twice in order to answer questions with a high degree of certainty. Bad Quality text was designated as containing significant structural and/or grammatical problems. Also it might require three or more readings to answer all questions. Finally, text that the raters could not be used to answer questions with a high degree of certainty. This criteria, although subjective, was one that captured most qualitative problems with clinical content that might render it incomprehensible to both human readers and computational algorithms.

Figure 32 illustrates the distribution of the Good Quality, Medium Quality, and Bad Quality text in the randomly selected sample (gold-standard) as annotated by the subject matter experts and the overall extraction rate from each category of the quality of text.

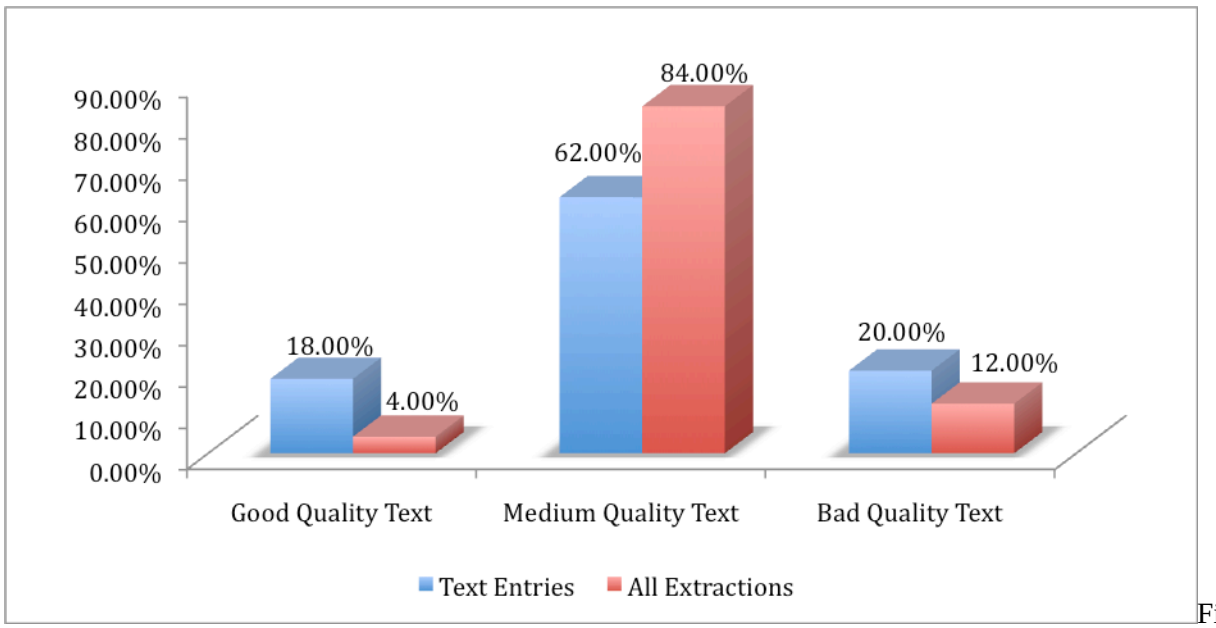


Figure 32: Quality of text and extraction rates based on the quality of text

It appears that although the frequency of Bad Quality text (20%) is close to the frequency of Good Quality text (18%) in the gold-standard, it incorporated significantly higher (3 times more) concepts or facts to be extracted (12% of all extraction from Bad Quality text vs. 4% of all extractions from Good Quality text). With a closer inspection of the extraction rates (Table 4) a similar pattern emerges consistently for different types of concepts or facts extracted (observation concepts: 11.4% vs. 3.6%; facts about the location; 10% vs. 4%; negation:12% vs. 4%; modifiers:23% vs. 2%).

One explanation for this observation is that the Bad Quality text found in the clinical content is frequently and significantly richer than the Good Quality text. That is, it encompasses important clinical information that is densely represented and should not be disregarded merely because of its poor representation. On the other hand, the richness and the compact information representation itself may have contributed to the complexity and irregularity of the clinical content (from a syntactic and structural point of view) and by extension being rated as Bad Quality text.

However we explain this observation, it will be a testimony to the importance of devising computational algorithms for clinical text understanding that are resilient to the readability and quality of the text, in a way that enables extraction of the important clinical information and making it available for queries and further processing. We will discuss the performance of our algorithm in successful extraction of concepts from such text in the next section.

Section A: Semantic Validity of BLUE-Text Output

In order to provide an understandable and consistent method of interpreting the evaluation results in a way that it explains the Semantic validity of the BLUE-Text output we present the problem as an Extraction problem throughout this section. The assumption is that the output is Semantically valid if the pertinent set of concepts and facts are extractable from it. Extraction of a concept in this context means that the algorithm returns a single symbol, and that symbol is consistently associated with a proper object or concept in a domain of discourse. For example, the symbol “HTN” is consistently extracted to indicate the concept of “High Blood Pressure”, or the symbol “Severe” is extracted to indicate a quality such as “alarmingly high”, every time such concepts appear in the text.

It is important for a text understanding system to not only extract concepts but also to identify relationships between them. However, identifying the relationships between concepts can also be conceptualized as an extraction task. Extraction of a fact in this context means that the algorithm identifies relationships between 2 symbols and extracts it as a single fact. For example, the algorithm may extract the relationship between HTN and Severe from previous example as a single fact such as “HTN modifiedby Severe”. A fact frequently is comprised of 3 concepts: A subject (e.g., HTN), a relationship (e.g., modifiedby) and an object (e.g., Severe). For a true

positive extraction of a fact, all three concepts should be extracted as true positive first, or the fact will be extracted as False Positive (type I error).

The advantages of presenting the validation problem as extraction problem are two fold. First we can use a simple and understandable method to evaluate the performance of BLUE-Text in regards to extraction of concepts as well as facts about (and relationships between) them. The second advantage is, that there are standard and proven metrics for evaluation and interpretation of Extraction problem that can be used to interpret and discuss the results of BLUE-Text evaluation.

In the upcoming sections we will use the following standard measures frequently used to evaluate information retrieval, to evaluate the performance and validity of the BLUE-Text regarding extraction of concepts or facts about concepts (i.e., assertions about relationships between concepts):

Recall (sensitivity): $R= TP / TP + FN$ (Type II error)

Precision (positive predictive value): $P = TP / TP + FP$ (Type I error)

Weighted harmonic mean (F) of precision and recall is also calculated for all variables using the following formula: $F= ((\beta^2 + 1) P \times R) / ((\beta^2 \times P) + R)$

The F measure combines the precision and recall rates to create a single measure of efficiency for NLP algorithms. A β value of 1 gives equal weight to precision and recall (Balanced-F), and a value higher than 1 gives more weight to the recall.

We used Error as a single measure to calculate the probability of any type of error (Type I or Type II error) in information retrieval:

Error: E= FP (Type I error) + FN (Type II error) / TP + FP (Type I error) + FN (Type II error)

It is important to note that in systems like BLUE-Text the True Negative rate will be extraordinarily high, due to large vocabulary with hundreds of thousands, if not millions, of concepts that will be True Negative for ever single case of True Positive finding. For example if a token is successfully extracted as a concept mapped to the concept of Diabetes, millions of True Negative should be recorded for the fact that the system has not mapped it to other concepts such as Infarction, Infection, etc. This makes use of measures such as “Accuracy” and “Fallout” misleading as it may generate artificial results indicating a better than actual performance (see Evaluation Section in Chapter 3). But in case of evaluating the validity of the method in dealing with Negations in the text, where a single Boolean question is answered (is this fact negated or not?) these measures are applicable and meaningful.

It is important to note that in the following sections and for the purposes of readability, we will use the term accuracy (lower case) to refer to the Balanced-F measure that replaces the standard measure of “Accuracy”. The term “Accuracy” is used to refer to the standard measure of Accuracy only when discussing the statistics and results pertaining to extraction of assertions about the Negation of the Observation concepts.

A1. Overall Validity of the BLUE-Text Extraction Algorithm

Table 5 presents the performance statistics of the BLUE text for extracting concepts classified as Observations, Modifiers or Body Sites (Locus). Figure 33 illustrates comparatively the

performance of the extraction algorithm used by BLUE-Text in extracting these different categories of concepts from clinical content.

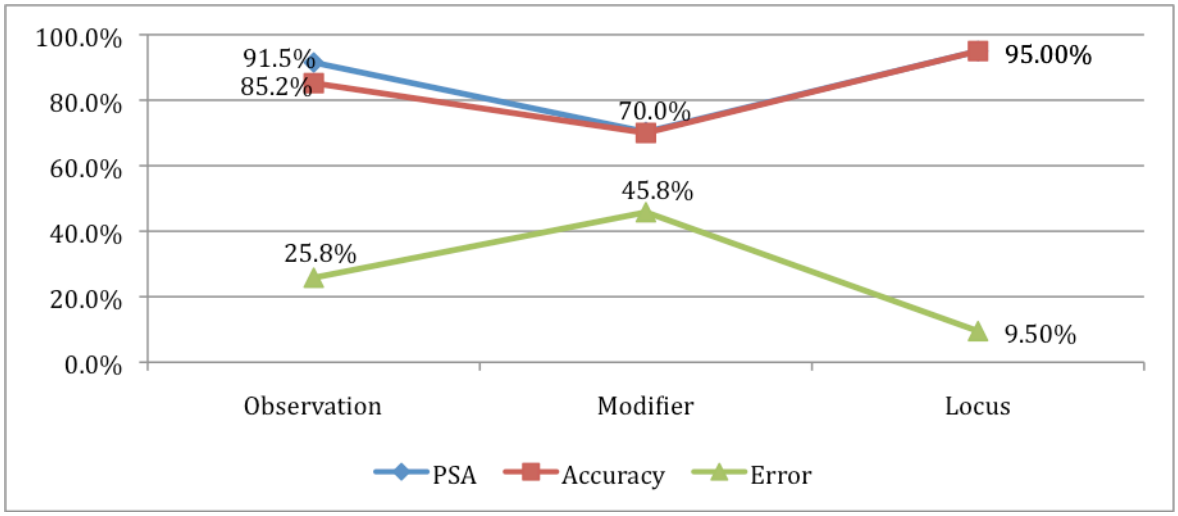


Figure 33: BLUE-Text extraction of different categories of concepts

It appears that the BLUE-Text extraction algorithm demonstrates an excellent (>85% accuracy) overall performance extracting concepts pertaining to Observations and Body sites (Locus), but an acceptable but lower than desirable performance for extraction of the Modifier concepts.

A2. Interpretation of the PSA, Accuracy and Error rates:

Figure 33 illustrates a strong correlation between the Positive Specific Agreement (PSA) and the accuracy (Balanced-F) of the algorithm (agreement between the algorithm and the gold-standard). That is, when there was a low agreement between the human raters about existence (positive agreement) of a concept in the text, the BLUE-Text algorithm also shows relatively low accuracy and high Error rate, proportional to the PSA.

This pattern (proportionally and directly correlated PSA and accuracy rates, or indirectly related PSA and Error rates) may indicate that the low accuracy and high error rates are in part attributable to variables external to the algorithm. Examples of such variables may be existence

of other contextual or background information that causes disagreement or ambiguity in interpretation of the text even for human experts.

A low accuracy (high Error) rate despite high PSA among human raters may be an strong indicator of an issue internal to the extraction algorithm itself, as the gold-standard appeared to be unambiguously interpreted by human experts but not by the computational algorithm.

A high accuracy (low Error) rate despite a low PSA rate may be an indication of a problem with the quality of the gold-standard used for evaluating the performance of the system. It may indicate that ratings and agreement between human experts can be explainable by chance and render the gold-standard inconclusive.

It is important to note that human experts in this evaluation demonstrate a PSA of 86.8% for all extractions. This is indicative of a high quality of the gold-standard used for this evaluation. However, we will provide the specific PSA for each category of the extraction and in times will use category specific PSA for interpretation of significance of our findings.

A3. Extraction of Observations:

This class of concepts represents health related events such as diseases or disorders, signs and symptoms, medications, or diagnostic and therapeutic events (See Chapter 3).

The BLUE-Text demonstrates an excellent performance in extracting all concepts classified as Observation (R: 79%; P: 93%;F: 85%; E:26%), comparable to most state of the art extraction system in presence (see Table 1). However a lower than expected Recall rate (79%) with Error rate of 26% demonstrates room for improvement and a reason for further investigation.

In a closer inspection of the BLUE-Text performance in extracting different subclasses of the Observation concept (Table 6) it is clear that the class of concepts representing “Healthcare Procedures” (therapeutic interventions, diagnostic tests, etc) alone can explain the lower than expected performance.

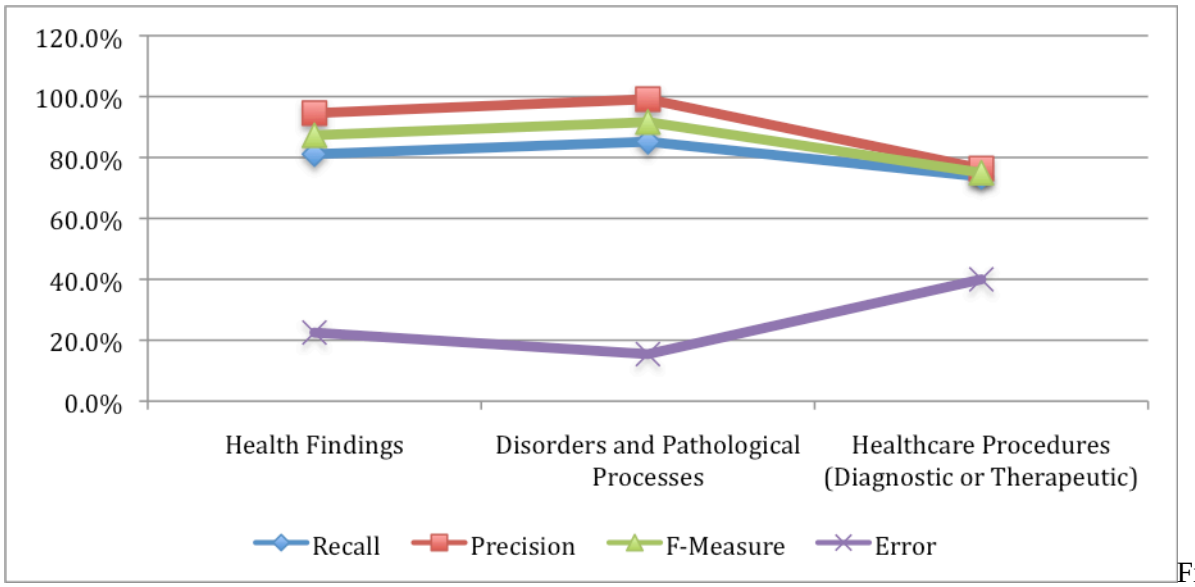


Figure 34: BLUE-Text Performance for the subclasses of the Observation concept

BLUE-Text demonstrates an acceptable but less than desirable performance in extracting concepts classified as “Healthcare Procedures” (R: 74%; P: 76%; F: 75%; E: 40%), but excellent performance in extracting concepts related to other categories of concepts such as Health Findings (R:81.1%; P: 94.6%; F: 87.3%; E: 22.5%) and Diseases or Disorders (R: 85.2%; P: 99.1%; F: 91.6%; E: 15.4%).

As the BLUE-Text Healthcare-Procedure is a class defined by using UMLS-SN Semantic Types, a drill down into performance of the BLUE-Text in extraction of those UMLS-SN concepts may shed more light on the contribution of each category of concepts in the observed performance. The Table 12 presents BLUE-Text performance in retrieval of Clinical Observations by UMLS Semantic Types, including Semantic Types that are used by BLUE-Text to define the

"Healthcare Procedure" concept. Figure 35 compares the accuracy and Error rates of the BLUE-Text in extraction of Semantic Types umls:Therapeutic_Preventive Procedure (R:66.7%; P: 69.0%; F: 67.8%; E= 48.7%), umls:Diagnostic_Procedure(R: 66.7%; P: 80.0%; F: 72.7%; E: 42.9%), umls:HealthCare_Activity (R: 90.9%; P: 83.3%; F: 86.9%; E: 23.1%) and umls:Laboratory_Procedure (R: 100.0%; P: 100.0%; F: 100.0%; E: 0.0%; (N=5)):

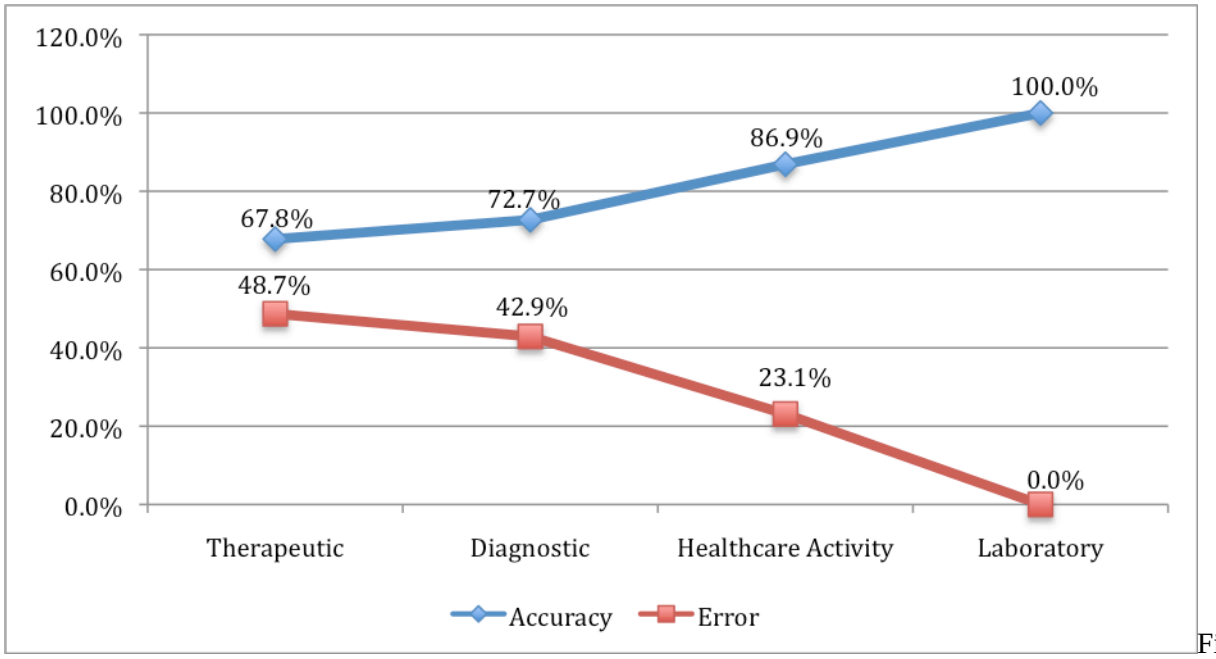


Figure 35: BLUE-Text performance and subclasses of the Healthcare Procedure

It is clear that the umls:Therapeutic_Preventive Procedure, umls:Diagnostic_Procedure with Error rate above 45% are responsible for the less than desirable performance of the BLUE-Text algorithm in extraction of the "Healthcare Procedure" concepts from clinical text.

An explanation to this discrepancy between performance of an algorithm between siblings of a class may be provided by the observation that abbreviations, acronyms, short hand forms or use of trade-names that are not supported by the UMLS or the BLUE-Text Lexicon were more frequent in these category of concepts in the gold-standard. For instance, we observed that many of the failures in extracting Therapeutic Procedures were attributable to the use of trade-marks

and ambiguous abbreviations rather than full-names for medications. However, these abbreviations were easy to interpret and contextualize for human experts. BLUE-Text performance can dramatically improve by including those abbreviations and trade-names into the Lexicon.

A4. Extraction of Body Site (Locus):

Locus is a concept from the Semantic model that serves as a super class for all anatomical concepts including but not limited to the body locations or regions (e.g. abdomen), anatomical structures (e.g., skull), body systems (e.g. respiratory tract), organs, tissues and body fluids (e.g. tear, blood).

The BLUE-Text demonstrates an excellent performance (Table 5, Figure 33) for extraction of the anatomical concepts (R: 94.2%; P: 95.8%; F: 95%; E: 9.5%) from clinical text.

A5. Extraction of Modifiers:

The Modifier is a Semantic model concept that intends to define a super concept for all qualitative (shape, color, texture, appearance, look, feel, etc) or quantitative (size, degree, level, amount, score, etc) descriptor or modifier concepts in the text. However, BLUE-Text does not include a separate Terminological knowledge for Modifier concept and relies solely upon UMLS-KS as source of that knowledge.

The BLUE-Text demonstrates an acceptable but less than desirable extraction rate for Modifier concepts (R: 92.86%; P: 56.52%; F: 70.27%; E: 45.83%). Although the overall accuracy is acceptable, the low precision and high Error rate is an indication of problems rooted in the system knowledgebase.

As mentioned in the case for low accuracy for Healthcare Procedures the extraction algorithm used by BLUE-Text for all classes is the same, and different performance statistics in different classes can be in part due to an incomplete or inaccurate knowledgebase. In this case a rather high Recall despite low Precision rate may be an indication of modeling problems originated from the semantic or syntactic knowledgebase. For example an insufficient or erroneous model (e.g. one that does not follow a consistent method in assigning class membership in a taxonomy, or is careless about multiple inheritance) may lead to ambiguity in extraction that may demonstrate itself as high sensitivity and low precision (high FP or Type I error).

As BLUE-Text uses UMLS-KS as the sole source of knowledge to inform processing Modifier concepts, a thorough and careful examination of UMLS-KS may reveal modeling problems that can in part explain the observed Errors in BLUE-Text performance. This is understandable and somehow foreseeable as UMLS-KS, being mainly focused on capturing biomedical concepts, is not intended (and does not claim) to be a reference model for Modifiers, and existence of these concepts in the UMLS-KS is recent and incidental to inclusion of such terms in some of the modern medical vocabularies (such as SNOMEDCT). If this observation is true, it indicates that using a more complete and customized knowledgebase specifically developed to model Modifiers may radically improve the BLUE-Text performance in this area.

An expectable consequence of having low accuracy in extracting Modifier concept is an even lower accuracy of extraction of facts related to the Modifier concepts. That is, it can be expected that relationships between Observation concepts and their Modifiers will not be accurately extracted. For accurate extraction of these relationships (Understanding) it is critically important that the Observations and Modifiers concepts are also extracted accurately. An error in extracting

any side of a relationship will result in an inaccurate (FP or FN) extraction. Effects of the high Error rates in extracting Modifier concepts will be discussed in the next section.

A6. Quality and Quantity of Observations:

Modifier concepts further refine, and describe of Observations within the text. The task of a text-understanding system in this case is to extract a semantically equivalent statement, making assertions about the observations extracted from the text and their modifiers.

The BLUE-Text demonstrates a modest accuracy (R: 78.7%; P: 51.3%; F: 62.1%; E: 55.0%) with rather high error for explicating relationships between *Modifier* and *Clinical Observation* concepts (Table 7).

As explained in the section A4, this is an expectable problem due to low accuracy of extraction algorithm in extraction of the Modifier concepts using the UMLS-KS as the source of knowledge. This performance may be improved by using a dedicated lexicon for Modifiers that does not have some of the modeling problems inherent from using UMLS-KS.

A7. Anatomical sites (Locus) of Clinical Observations:

Clinical Observations may be associated with some *Locus* (anatomical sites, organs or tissue, etc) within the clinical text. The task of a text-understanding system in this case is to extract the relationships between the observation and the body site as explicit statements that are semantically in agreement with what is presented in the text.

The BLUE-Text demonstrates an acceptable accuracy for explicating relationships between Locus and Clinical Observations concepts (Table 7). However a closer examination of the

subclasses of the Locus concept (Table 8) shows a slightly better performance in identifying the histological relationships ()(Figure 36).

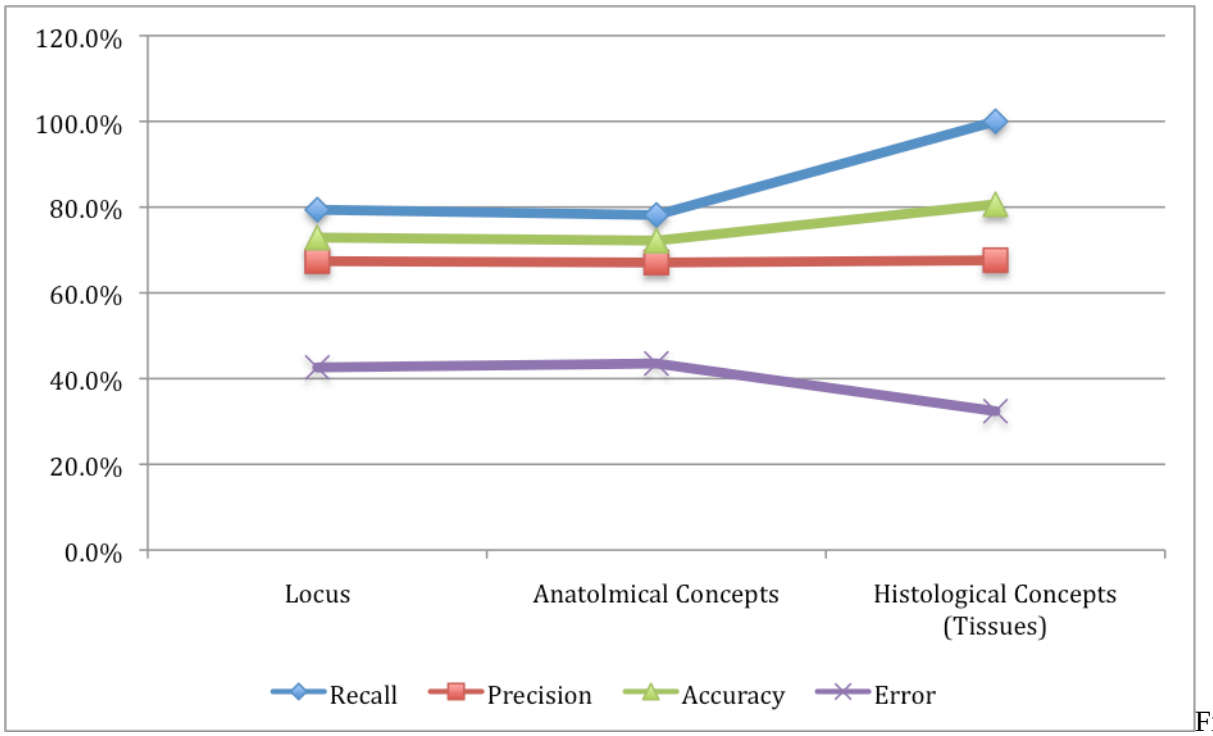


Figure 36: Locus (Anatomical and Histological concepts) related to Clinical Observation

A8. Negation of Observations:

Negation is basically an assertion about quality of a concept and BLUE-Text semantic knowledge models it as a subclass of a *Modifier* concept. However as BLUE-Text deals with Negation as special case the accuracy and validity measures for Modifier concepts does not include performance statistics for Negation. Negation is considered as a special case as UMLS-KS does not provide a rich knowledgebase of Negations and related terms that can be used for general and biomedical language understanding. Therefore we had to construct a custom made lexicon and syntactic knowledgebase to support BLUE-Text extraction algorithm with lexical expressions and knowledge related to Negations. However the extraction algorithm is identically the same used for extracting other types of facts from the text.

As mentioned in chapter 3, as negation is modeled as a Boolean statement in the BLUE-Text output (an Observation is either negated or is not), we have been able to also calculate Specificity metric along with other measures such as Accuracy and Fallout for the Negation of Clinical Observations (Table 7).

BLUE-Text demonstrates an excellent performance (R: 97.4%; S: 97.6%; A: 97.3%) for explicating negations of Clinical Observations, one that closely matches the human experts agreement in the gold-standard (observed inter-rater agreement: 98.2%, κ=83.5%).

A9. Modifier, Locus, Negation and The Effect of Knowledgebase:

Figure 37 compares the overall accuracy and error rates for the three category of assertions evaluated. It seems that the Modifiers of Clinical Observations are the only category of assertions that BLUE-Text performance is inconclusive and non-reliable. The accuracy of assertions about Locus and Negation of Clinical Observations has been acceptable or excellent. Figure 37 also suggests that the PSA rate -an indicator of the agreement between human experts and the quality of gold-standard- tallies the accuracy of extraction in all categories.

Although low PSA of the gold-standard (65%) for extraction of Modifiers makes any interpretation inconclusive, we have presented evidence that higher than expected error rate in this category is mainly due to issues pertaining to use of UMLS-KS as terminological knowledge for this category of assertions and are not related to the algorithm itself. For instance, although the extraction algorithm used for the 3 categories of assertions are the same, and accourding to the BLUE-Text semantic knowledge Negations are models as subclasses of Modifier concepts, that is, they inherit all features and properties of Modifier concepts, Negations demonstrate a near perfect extraction with high accuracy and very low error rates. As the lexicon and syntactic

model used for Negation is a custom made knowledgebase and the only variable that may explain the different performance observed between Negations and Modifiers.

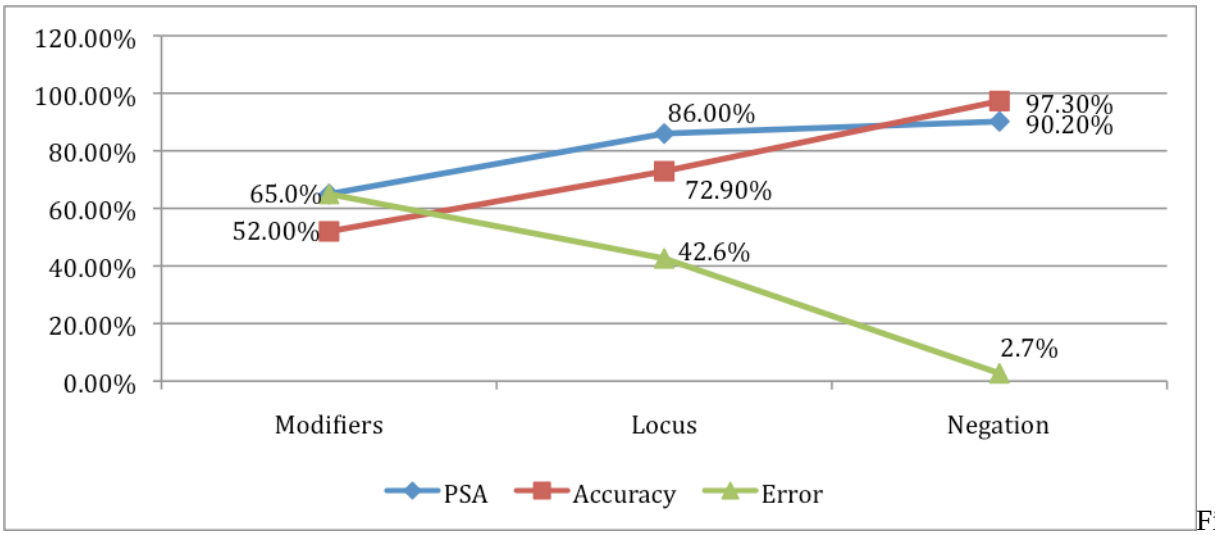


Figure 37: Extraction of Modifier, Locus and Negation of Clinical Observations

One may argue that use of Modifiers in the text may have increased the complexity of text, decreased its quality and contributed to the higher error rate in overall performance. That is, the low accuracy may have been the result of bad quality text and a low performance of BLUE-Text due to bad quality text. Although we will provide a more detailed discussion of the BLUE-Text performance by quality of text in next section, but a preliminary assessment (Figure 38) reveals that the BLUE-Text performance does not deteriorate as the quality of text decreases. In fact, BLUE-Text demonstrates a better performance (although not statistically significant) when the quality of the gold-standard has deteriorated.

Figure 38 shows that the performance of the BLUE-Text has been consistent not only in the Modifiers category, but also in extraction of Locus and Negations too. This is consistent with our expectation, that is, as the algorithm remains the same for all categories, and as the quality of text does not affect the accuracy and Error rates in BLUE-Text implementation, the observed difference between performance of BLUE-Text in Modifier, Locus and Negation categories are

attributable to quality (completeness, consistency and validity) of the knowledgebase available to the system (UMLS-KS).

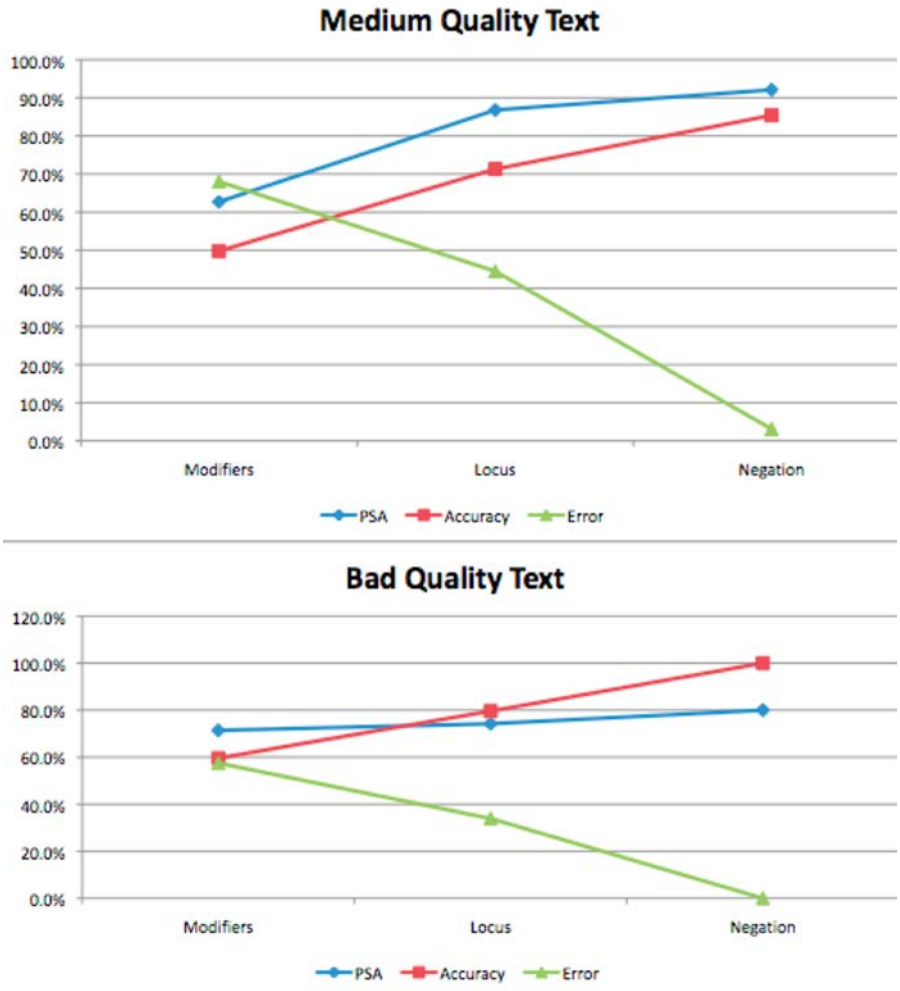


Figure 38. Quality of Text, Accuracy and Error [38]

A10. BLUE-Text and Good, Medium, and Bad Quality Text

Table 16 and Figure 39 compares the overall accuracy and error rates of the BLUE-Text Algorithm in extracting Clinical Observation concepts, Modifier statements, Negation statements and statements pertaining to the anatomical sites of Clinical Observations.

[38] Statistics pertaining to the Good Quality text is not shown here due to low (n=3) sample size (Table 4).

It is evident that BLUE-Text maintains a rather consistent performance regardless of the quality of text. All performance statistics discussed in the previous sections for validity of processing clinical text in this chapter is applicable to each of the 3 categories of quality of text in this evaluation.

Table 16: Quality of text and Validity of BLUE-Text

	Good Quality			Medium Quality			Bad Quality		
	Reliability	**Accuracy**	**Error**	**Reliability**	**Accuracy**	**Error**	**Reliability**	**Accuracy**	**Error**
Observation	93.0%	90.5%	17.4%	91.5%	84.7%	26.5%	88.9%	86.8%	23.3%
Modifier				78.6%	57.1%	60.0%	75.0%	75.6%	39.3%
Locus	100.0%	85.0%	26.1%	86.8%	71.3%	44.5%	74.2%	79.6%	33.9%
Negation	100.0%	100.0%	0.0%	92.1%	96.9%	3.1%	80.0%	100.0%	0.0%

The performance measures for BLUE-Text algorithm are observed for the Negation statements in all category of the quality of text. Clinical Observation concepts and statements regarding Locus and Modifier demonstrate relatively lower accuracy and higher error rates in all categories of quality of text. Although with a superficial review of statistics one may identify slightly better performance for the Good Quality and Bad Quality text than for the Medium Quality text, but the differences are not practically significant due to considerable overlaps between the confidence intervals observed in a relative categories. One explanation that may explain successful parsing of bad quality clinical text, comparable to a good quality text would be the richness and density of bad quality text in terms number of important facts and assertions that human experts could extract, regardless of irregularities and syntactic errors. It is seems that bad quality text seems to have shorter sentences or phrases than good quality text and frequently missing expressions are syntactic or structural elements that there absence was not considered critical for a human reader, or for a parser that does not have syntactic commitment to such structure. A Good Quality text is

also by definition one that does not have complex lexical expressions and patterns, frequently with short sentences structured for better readability.

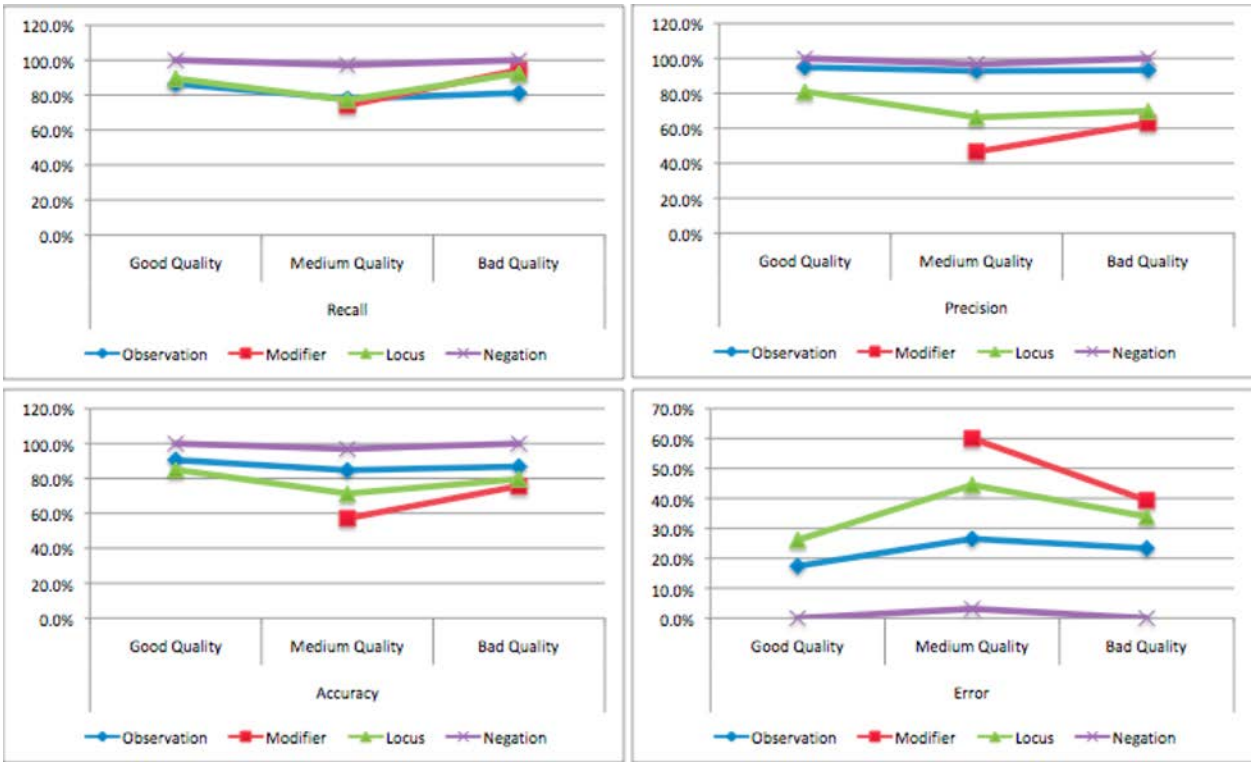


Figure 39: Quality of Text and Recall, Precision, Accuracy, and Error Rates

However, Medium Quality text frequently presents with longer sentences, and more complex lexical expressions. A minimal syntactic parser will probably make more errors in interpreting longer and more complex sentences than shorter and simpler sentences.

Section B: Syntactic Validity of the BLUE-Text Output Representation

Syntactic validity of text understanding system relates to construction and representation of an optimal output that can be easily interpreted and used by human or a computer program. An optimal output for a text understanding system may be one that uses an expressive and rich information representation framework to formally represent the information content of output,

clearly and unambiguously, and explicates details with proper granularity. When possible and in order to improve reusability and interoperability, mappings to the standards based controlled vocabularies should be provided. Output representation framework used for a text understanding system should enable contextualization and repurposing of the information content to support semantic interoperability. In a distributed environment such as one advocated by the CTSA program where communities of multidisciplinary research are loosely coupled through a network of just in time information sharing and information exchange, it is desirable to provide a self-descriptive output that can be immediately interpreted and integrated by other automated systems. Chapter one introduces detailed desiderata for an optimal output representation framework that will be used to scrutinize the syntactic characteristics and validity of the BLUE-Text output.

B1. Output Representation Framework

BLUE-Text uses information and knowledge representation languages from the Semantic Web technology platform (Berners-Lee 2000; W3C 2004)(Figure 13) to construct a formal (computer interpretable) output. BLUE-Text uses RDF(S) and OWL as the main information and knowledge representation languages to represent not only the output but also all mappings between concepts and relationships and their formal semantics within ontologies that are available to the system. This has been the key factor in conceptualization of a formal output representation framework that is unambiguously interpretable by computer programs, and is self-descriptive enough to enable repurposing and contextualization of information for both syntactic and semantic interoperability and integration.

Furthermore, use of formal knowledge representation languages to provide semantic and domain knowledge necessary for interpretation of BLUE-Text output enables use of computer reasoning

for knowledge discovery and computer reasoning, and enhances information retrieval, search and integration. Formal output representation also enables seamless extension and contextualization of the knowledgebases and output model to support new use cases or to be queried from a different perspective. It allows using ontologies from a collaborative network to enhance the BLUE-Text knowledgebases that enables easier customization and adaptation in new and novel environments.

The Semantic Web also provides a set of standards for information retrieval (SPARQL), reasoning and inference (Description Logic, SWRL) and interoperability online that complements its representation languages to develop a ubiquitous, distributed and yet integrated environment for information and knowledge sharing and reuse. That is, in an optimal setup, systems like BLUE-Text can use shared knowledge from internet to improve and contextualize their performance, and automated systems can invoke BLUE-Text and obtain a customized output that is immediately interpretable with least or no pre-coordination.

B2. Explicit (unambiguous) output representation:

Output of NLP system cannot be useful and 'understandable' by other systems if information is not explicitly and clearly defined in a way that allows two different computer systems to interpret them identically and precisely. A concept in one system may present with ambiguity and vagueness in another, unless definitions and semantic properties are available in a way that can be used by computer systems to automatically disambiguate terms and interpret them identically and in a proper context. This is probably one of the most important prerequisites for a information integration and interoperability and for contextualization and reuse of information by both human experts and automated agents.

In effect BLUE-Text output is a formal ontology that is populated with instances extracted from the input text, all linked to instances of concepts available from the domain, semantic and terminological knowledge. All facts (statements) in the output are asserted using constructs with formal semantics and precisely defined in an ontology. This representation not only makes information readily accessible but also defines all its content explicitly (unambiguously and precisely) for secondary users.

B3. Expressive Output with Detailed Granularity

To support effective information integration, BLUE-Text explicates an expressive and rich output that captures relevant detail extractable by the NLP system, and represents it regardless of the primary use cases and applications. For example, for every Clinical Observation not only the modifier or anatomical location are presented, but also facts such as presented by patient himself or observed by a care giver, historical context of the observation (Ketosis in a patient with history of DM), Background events contributing to the observation or event (Seizure while driving), causative context of the event (nausea and vomiting after drinking bad milk), temporal context of observations ("night time headaches"), and other contextual information are also presented. A comprehensive and extensible model of negation, and uncertainty in clinical text adds to the expressivity and granularity with which information are described in the output. This information can be used by any third party system to customize information retrieval, extraction and contextualization, as necessary for information sharing and reuse, integration and interoperability.

B4. UMLS Encoding

Controlled vocabularies and terminology systems establish an agreed point of reference between multiple systems when referring to the same medical concept (2003; The College of American Pathologists 2003), and help to explicate and disambiguate concepts.

BLUE-Text output is fully mapped and encoded to UMLS MTH concepts through a method of consistently mapping all extractions to a corresponding CUI, and a corresponding UMLS-SN Semantic Type. Using these mappings and the SKOS representation of the UMLS-KS, BLUE-Text output also makes the whole UMLS-KS available for inferences and queries for all semantic applications. That is, not only a corresponding UMLS code is provided for each extracted concept in the BLUE-Text, but also all knowledge (definitions, terminologies, semantic relationships, source specific codes and metadata, bi-directional mappings between sources) are also made available for retrieval or querying.

That is, BLUE-Text not only provides with a fully encoded NLP output, but also a robust taxonomy service along with it that can be used on demand to exploit UMLS-KS knowledgebase for information retrieval and integration.

B5. Knowledge based output representation

Frequently NLP systems draw inferences, based on some heuristics, assumptions, defaults and rules, in order to deal with complex real world use-cases and requirements (Friedman and Hripcsak 1998), however, this knowledge is omitted from the output based on the assumption that the systems utilizing the output share the same context and commit to the same heuristics, assumptions and presumptions. Absence of this information prevents from secondary use of NLP output or adoption of these systems in new environments as the output is not provable or cannot

be validated. An optimal output representation should be self-descriptive and enable information systems to access the relevant context, and the semantics of the information represented in the output, in a way that conclusions can be traced back to the associated evidence in the text and the logic and rules of inference in the knowledgebase.

BLUE-Text constructs a self-descriptive, formal and explicit output that can be used by automated agents for information retrieval, contextualization, and repurposing as links to the semantics of all concepts and relationships are explicated within the output and ontologies formalizing that semantic are made available by the system. Use of the Semantic Web to represent all knowledge, and information throughout the system is critical not only for providing a syntactically optimal output, but also for enabling users of the output to extend and reuse the system and its output on demand.

B6. Conceptual Graph Vs. Task Specific Output Construction

Figure 12 suggests that BLUE-Text process can in effect produce two different sets of outputs: 1) A Conceptual Graph that is basically a graph representation of the Parse Graph –the product of the minimal syntactic parsing (Figure 23)- after mapping to appropriate syntactic and terminological knowledge, including UMLS encoding (Figure 25), and to the domain ontology plus inferences made by the semantic parser about the syntactic relationships of the tokens of text based on their positional and semantic index (Figure 26); and 2) a customizable and task specific RDF output (Figure 27) that is based on the semantic model (Figure 17) and in effect populates instances of its classes with information extracted from the text and establishes relationships between them, and their semantics according to the semantic model and the domain ontology.

It is important to notice that the RDF output is by itself an extension and contextualization of the Conceptual Graph to construct a generic and application independent output for the BLUE-Text and Chapter 4 provided evaluation results for only one possible implementation of the RDF output. Different pragmatic use cases may be conceptualized that may require extension of the Conceptual Graph in different ways to support novel tasks and use cases. It is also possible to build upon the RDF output and extend it to further customize or reuse BLUE-Text output, without requiring reprocessing text.

Exa2mples of why and how Conceptual Graphs can be used to provide task specific customization of output would be a) construction of a transformation script that produces an XML message conformant to the HL7 and Clinical Document Architecture (CDA) from BLUE-Text output b) systems that only need a specific type of information, such as Medication Lists, Diagnoses Lists etc; c) Using BLUE-Text output in legacy applications that depend on keyword based search or regular expression based search patterns; d) using BLUE-Text output in systems that only require information extraction based on UMLS-Semantic Types or specific CUIs and similar situations where access to the syntactic structure of the original text is necessary to fulfill the requirements of the applications.

Use of the default RDF output generated by BLUE-Text may be suitable for scenarios that systems need to query and retrieve information based on abstractions. An example of such use case may be a search for identifying candidate cases for a clinical research based on inclusion criteria. It should be noted that these applications would also have access to the underlying

syntactic structure of the text via links and associations to the Conceptual Graph from RDF output.

Figure 40 demonstrates how different layers of models incorporated into the self-descriptive output of the BLUE-Text provide context to enable different types of query and information retrieval from the output.

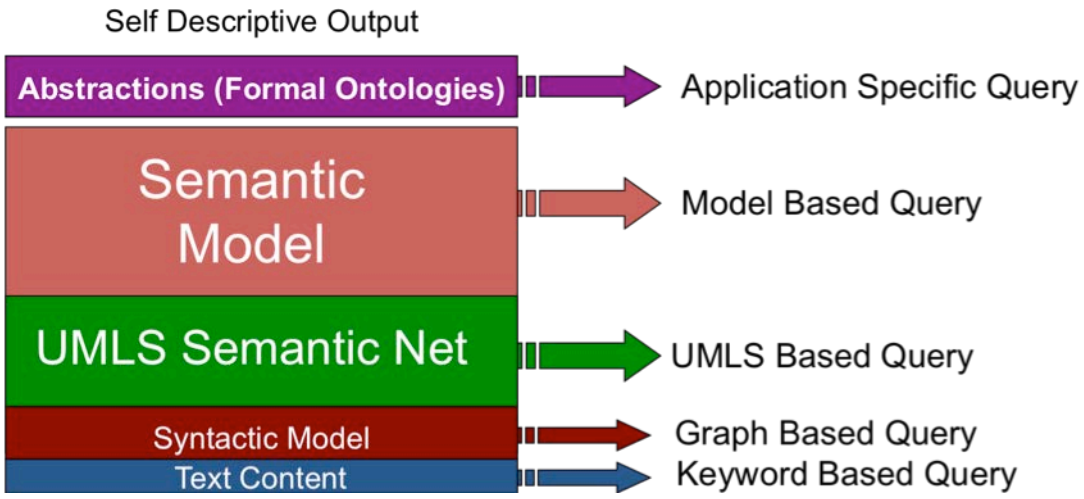


Figure 40. Layers of models in RDF output and the queries they support

The self descriptive RDF output at minimum includes all tokens associated with the original text that can be used for querying and extraction using simple string and expression matching techniques. The syntactic model enables querying not only using the tokens of text but also relationships such patterns happening before or after each other or belonging to lexical categories (negation for example). Next layer of concepts involves domain knowledge provided by the UMLS-SN (UMLS SKOS representation), which adds the ability to extract content also based on their membership in semantic types and their CUI encodings. The Next layer of models (the Semantic Model) allows extraction of concepts based on their ontological relationships defined by the Semantic Model. This layer can be extended and contextualized by mapping to specialized ontologies that introduce additional layers of abstraction for the interpretation of

output. In practice features of all layers can be combined into one single query to extract a targeted and precise retrieval.

Comparative analysis of the BLUE-Text validity

An objective and reliable comparison of the BLUE-Text performance statistics with other text-understanding systems such as MedLEE, or MedSynDikate is only possible if evaluation is based on a shared gold-standard and in a controlled environment where the system is the only variable of the evaluation. Due to technical and practical issues such comparison was not feasible at this time. However it is possible to use published performance statistics from similar systems to overview the overall performance of the systems in areas that they are validated for. For example overall validity of MedLEE and BLUE-Text in extraction of Clinical Observations may be informative in a sense that it provides a basis to set the expectations for the performance of systems if transported to a different environment. That is, a comparable overall performance statistics between MedLEE and BLUE-Text may suggest that the MedLEE system would operate similarly if ported to the environment within which the BLUE-Text has been tested for. However, we recognize that such an assumption could only be validated by evidence collected through an empirical examination of both systems in a controlled environment using a rigorous and systematic evaluation methodology.

Table 17 provides an overview of the performance statistics as published by the text-understanding systems cited in the chapter 2. All systems are compared using the standard Recall and Precision and all statistics are based on the overall performance of the system to extract concepts of importance for the use case (domain) in which the systems have been evaluated. That is, performance statistics provided here does not include Recall or Precision rates for explication of negation or anatomical relationships of the clinical observations.

Table 17. BLUE-Text and other text understanding systems

System	Primary Domain	Overall Performance	Syntax Independent	Standard Encoding	Formal Output	Shared Domain Ontology	Extensible Reusable Ontology	Self Descriptive, Extensible Output
Taira, et al.	Thoracic radiology	R: 79%-87% P: 88%	No	No	No	No	No	No
MedSynDikate	Gastro-Intestinal diseases	R: 80%-93% P: 81%-93%	No	UMLS	YES (KL-ONE)	No	No	No
Wang and Ong	ECG reports	R: 85% P:?	No	No	No	No	No	No
MPLUS	Head CT Scan Reports	R: 98% P: 85%	No	No	No	No	No	No
MedLEE	Radiology	R: 70%-85% P: 87%	No	UMLS	No	No	No	No
LSP	Asthma Control Quality Assurance	R: 82.1% P: 82.5%	No	No	No	No	No	No
BLUE-Text	General Purpose	R: 79%-86% P: 93%-95%	Yes	UMLS	YES (OWL)	Yes	Yes	Yes

The current prototype implementation of the BLUE-Text algorithm appears to be comparable in overall performance to the other cited text understanding systems. However this comparable performance statistics should be interpreted in the context within which these systems have been evaluated. BLUE-Text have been evaluated using a corpora from chief complaints and triage notes from emergency departments from 8 different hospitals. For this evaluation BLUE-Text algorithm and models have not been tuned or customized for any specific lexical pattern, syntactic structure or convention that may characterize clinical text from a clinical specialty or domain. That is the BLUE-Text performance was based on a domain and task independent (generic) evaluation. The expectation is to observe better performance when optimizing and tuning a system with a specific use case or domain in mind.

Comparative Analysis of the BLUE-Text Output

To illustrate the syntactic features of the BLUE-Text output we will compare the RDF output provided by the BLUE-Text (Appendix C) to the data structure (XML) constructed by the MedLEE system (Appendix B) using an identical sample text. The MedLEE is one of the most sophisticated, most cited, and widely adopted medical NLP tools in presence. We will highlight similarities as well as differences between two different approaches for the output representation and discuss their implications on extensibility, reusability, interoperability, and portability to a new domain or utility.

The following text was used by both systems to create the output that will be studied here:

“a 13 years old teenager with nausea and vomitting after drinking bad milk. has taken Reglan that made her drowsy and confused. no fever and headache. Feels tingling on finger tips and around her mouth. dry skin in observation.”

I. Structured (implicit) vs. Formal (explicit) Output

As the critique and discussion of shortcomings of the data structures used by the MedLEE is beyond the scope of this document we will stop at reviewing only few illustrative examples that can establish a basis for a fair and objective comparison.

At first glance outputs of both systems (Appendix B and C) seem to be conveniently structured as XML documents (Appendix B and C). MedLEE structured output follows a schema that provides a structured format (NLP output) and a parsed format (text parser output) for each sentence identified by system. XML tags such as `<sentence>, <structured …>, <tt>` are used as conventions to denote the content of each data element.

A closer look will reveal that all data elements in the MedLEE output (Appendix B) are ad hoc character strings with no immediate meaning that can be used by neither human users nor automated agents to be interpreted unambiguously, precisely and in the right context. (e.g., `<undef>milk</undef>, <problem v="nausea"…>).` For example the attribute “v” is used in all tags to present different types of vlaues based on the XML element. Values of the attribute ‘v’ are all string values refering to some unknown terminology. Some times a value ‘v’ is accompanied by a ‘code’ that seems to be a UMLS code for the ‘v’ (e.g. `<problem v='nausea' code='umls:C0027497_nausea' …>` but in other places ‘`v`’ itself refers to a UMLS code for its parent XML element (e.g. `<code v='umls:C0027497_nausea'`). Meanwhile the attribute ‘`code`’ is not declared anywhere in the output and it is only meaningfull to a human expert that is already familiar with the kind of concept identifiers used by UMLS. Representation of the UMLS codes in the ‘`code`’ attribute is a concocsion of different values without any declaration or instructions on what they are and how they should be interpretted. Most importantly the values

of the ‘code’ attribute needs another level of text processing before being able to identify, query or extract the code it refers to. However we interpret the meaning of the ‘code’ and ‘v’, it is apparent that an arbitrary string (‘code’) is used to implicitly suggest some undeclared relationships between two undefined string values (“umls:C0027497_nausea” and “nausea”).

Figure 41. Structure of the MedLEE and BLUE-Text Output

The problem with such approaches to information representation is two fold:

First, an identical interpretation of the output by systems and entities other than its constructor, or even by the constructor itself in different times can not be guaranteed. This is a big impediment to the reuse and contextualization of the content by secondary seystems.

For example, it is impossible to determine whether the term ‘Vomit’ in this output refers to the substances of a vomiting act or the act of vomiting itself (disambiguation), by looking neither at

the string value (‘Vomit’) nor at the implicitly proposed CUI = ‘C0042963’. Also in the UMLS-MTH the code ‘C0042963’ is associated with both terms ‘Vomit’ and ‘Vomiting’; the term ‘Vomiting’ is associated with CUIs (‘C0042963’ and ‘C1963281’), and the term ‘Vomit’ is associated with CUIs (‘C0042963’ and ‘C0042965’). That is, as is, the CUI code by itself is of no help for disambiguation of the extracted term. It is only possible for a human reader to resolve the ambiguity by using the original text.

Secondly access and retrieval of information from this output requires a deep understanding of the structures used to represented information (the XML Schema) and following strict syntactic guidelines to extract information. As information are presented as strings without a formal semantic (computer interpretable meaning), they have no properties that could be used to determine relevance of information to any query or search criteria, other than the superficial syntactic features of a string value. A secondary system to consume this output an extensive training and customization process is required for human developers to construct custom tailored interfaces and queries to retrieve task specific information from the output.

Furthermore, in order to enable effective information access and retrieval for human users (e.g., a navigation and exploration user interface) and computer programs a custom application needs to be developed that consistently and appropriately interpret implicit and ambiguous structure and

makes explicit assumptions and rules that are not defined and declared within the document itself (such as the relationships between `id`, `sid`, and `idref` attributes and values, or the meaning of `mode2`, `mode3`, `dem, code,` and `v`). However, slightest change in the structure or schema, or in the string representation of the extracted terms may require reprogramming the interfaces and queries to accommodate the change.

In conclusion, in regards to the information representation framework the MedLEE system functions as an algorithm to convert free form and unconstrained text into a strictly structured text. However, although it is possible to infer some semantics as implied by the schema and structure used to represent the output, the process of retrieval, reuse and repurposing such output is a resource intensive, cumbersome and inefficient.

A quick look into the BLUE-Text output may suggest some similarities in that both systems are using a form of XML structure to represent output (Fig 41). However, BLUE-Text RDF output is not a typical XML, rather an RDF document that is formatted as XML for convenience. In fact BLUE-Text output can be formatted in 3 other valid RDF formats (N3, N-Triple and Abbreviated/XML), all with identical meaning and content. Appendix D presents with the same content as in Appendix C, subject of this evaluation, but in N3 format.

This lack of commitment to a structure or syntactic form in output representation has immediate and important implications:

A) Knowledge of structure, syntactic conventions, and schema is not required for accessing and retrieving information from this output. That is, computer programs need not to be pre-coordinated in order to access, interpret or retrieve information from this output

B) As there are no structural conventions used to format the outputs, any generic RDF visualization and navigation tool can be used to explore through the output, and interact with information directly and immediately (Figure 42).

C) Changes in the content and its format do not imply change or require reprogramming of the interfaces and tools used to interact with the output. That is, interfaces established between BLUE-Text and other systems do not change if the content or format of the output changes.

D) The RDF output provided by the BLUE-Text follows a standards based information representation framework with a formal semantics that supports advanced information retrieval and search through a standards based query language (SPARQL). Hence it is extremely easy to use SPARQL to custom tailor search and retrieval queries based on the semantic relationships between extracted concepts, their syntactic features available from string processing (e.g., regular expressions) or a combination of both. In next few pages we will see examples of using SPARQL to retrieve or transform information (Figure 43).

Using a formal output representation rather than a syntactic convention then enables BLUE-Text to refer to uniquely identified and properly and explicitly defined concepts rather than character strings. For example, as shown in Figures 41 and 42, the BLUE-Text refers to the whole text using a unique URI that is used to describe the properties of the whole text being processed. The character string associated with the text using the `:hasTextForm` property is only one feather of that uniquely identified object, another property of the object is its type which is defined as `:ChiefComplaint` in the output. Similar to MedLEE, BLUE-Text has also identified 5 segments of text that can be interpreted independently here called `:Evidence (#EV_309412434, "#EV_1703472089", "#EV_1658750257", "#EV_161514994", "#EV_258386563")`, each

:Evidence is associated with a text form and its explicit interpretations, for example :Evidence EV_161514994 is associated with the text "a 13 years old teenager with nausea and vomitting after drinking bad milk." and related to two other objects (MAN_17141111145, MAN_174111234) typed as :Manifestation through an explicit relationship :hasManifestion.

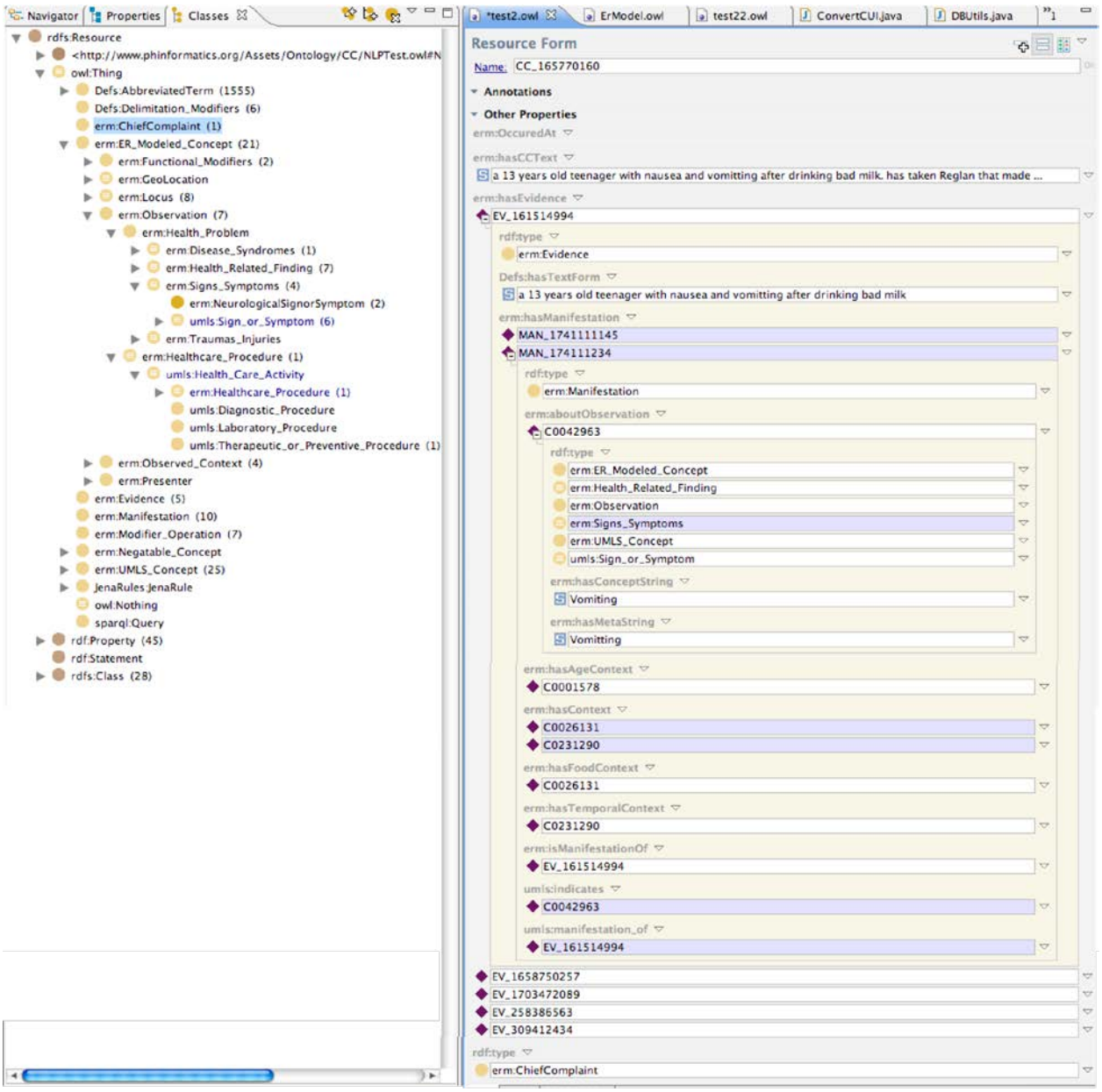


Figure 42. BLUE-Text output in a generic RDF/OWL editor software

The MAN_174111234 object is in turn explicitly associated with a UMLS CUI that not only

provides terminological data (string representations of the concept) but also with type and class membership information that can be used by a computer agent to disambiguate between terms such as ‘vomit’ and ‘vomiting’. For example an explicit type `umls:Signe_or_Symptom` excludes the possibility of the vomit or vomiting to be interpreted as a `umls:Substance` by either human or computational agents.

In BLUE-Text output representation uniquely identified, properly typed, and explicitly defined objects are related to each other and to literal values (such as character strings extracted from text). This is the key to enable computer programs to automatically interpret and compute logical entailment of the information content in the output. This formal and explicit representation not only provides a precise, unambiguous and self descriptive output that can establish an identical interpretation among contextually disparate systems and can be shared in a distributed network of collaborators, but also leverages computer reasoning for inferences, classifications, contextualization, enhanced retrieval and discoveries and semantic interoperability.

Richness, and Expressivity of the output

Another aspect of a proper output representation framework for text understanding is richness and expressivity of the output model to maintain and preserve granular detail that can be used to disambiguate, integrate, reuse, or sift through heterogeneous information found in the clinical text. We will use some simple but illustrative examples on how BLUE-Text output addresses this issue and how it compares with the MedLEE output.

Different systems A and B may need to extract information from a given NLP output with different requirements. System A requires extraction of all signs and symptoms of a patient using a keyword search in order to compare and match the results with a locally maintained

vocabulary. System A has no awareness of standard terminologies and their coding schemes. The system also requires anatomical and modifier information for each finding if present. This scenario represents a typical case for NLP interoperability when contextually different use cases need to extract a minimal set of information from clinical text to support a research questions, clinical hypothesis building, and data collection tasks. In these typical scenarios interoperability and conformance to standard terminologies are not required. However, it is possible that search criteria may change several times before finalized so interoperability and ad-hoc integration is of great importance.

To support System A using MedLEE output a custom script needs to be constructed that is specialized to make assumptions about what the intricate data structure would mean in the context of this particular query, navigate the data structure enclosing the information using multiple queries to different sections of the structure to access and extract the pertinent information and then to join the results of multiple queries into a unified result set (Table 18). The XML framework and the conventions used by MedLEE to construct its output does not allow to satisfy this task with a single query and search criteria.

Certainty	Finding/Problem	BodyLocation	Region
	Drowsiness		
High	Vomit		
High	Nausea		
Moderate	Tingling	Finger	Tip of Finger
High	Confusion		
	Dry	Skin	
no	Headache		
no	Fever		

Table 18: MedLEE extraction to satisfy the query for System A

System A using BLUE-Text output: An SPARQL query is required to express the question in formal terms without regards to the underlying syntactic structure used to express output. The query does not need to make any assumptions on the implied meaning or relevance of the underlying information to satisfy the query. More importantly we just need to issue one single and conceptually intuitive query to the output (Figure 43).

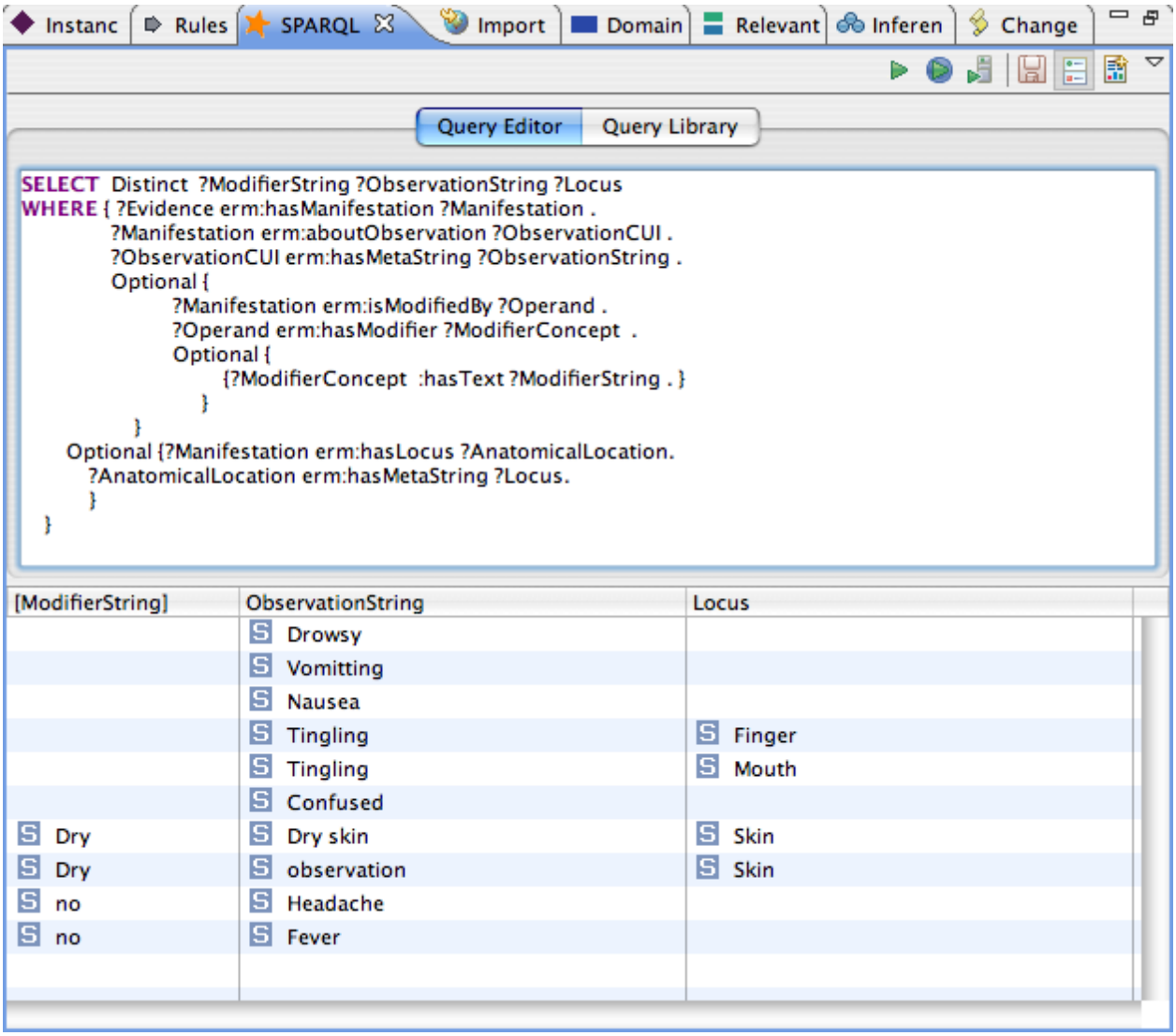


Figure 43. SPARQLquery to extract keywords and their associations from the text

Examples of the SPARQL query that can satisfy requirements of System A is presented at the Figure 43. The query follows the explicit relationships formally defined in the Semantic Model,

makes no assumptions about what they mean in the context of this task and even whether they apply here or not. It is the task of the query engine to infer that by computing the entailments of the query and the underlying RDF information.

While two systems provide comparable results, the differences are most revealing:

MedLEE fails to extract the anatomical relationship between ‘Mouth’ and ‘Tingling’, but BLUE-Text extracts and represents this relationship.

MedLEE also asserts that a finding ‘Dry’ is related to a body location ‘Skin’. The meaning and implications of such a representation is subject to different interpretations.

The BLUE-Text however extracts two different interpretations for the same expressions. One interpretation extracts a unique concept of “Dry Skin” associated with Locus ‘Skin’ and having the quality of ‘Dry’. It also indicates that there has been an ‘observation’ made at the Locus ‘Skin’ that appears to have a quality of ‘Dry’. Both interpretations have semantically similar (but not identical) interpretations, however the first interpretation is more specific that the later, both are true positive extractions but missing from MedLEE output.

Another issue with the ambiguity and the granularity of the MedLEE output is using ‘Certainty’ to indicate a subjective and vague certainty score or negation of a concept interchangeably. The question is if we assume that there is an spectrum of certainty from ‘no’ to ‘High’- ‘no’ being true negative observation, and ‘High’ being true positive observation- where ‘Tingling’ with a ‘moderate certainty’ stands?, what its ‘moderate certainty’ means for a retrieval query and for its future interpretations? The vagueness in representing “Negation of an extracted concept” and “Certainty of the extraction of the concept” make it difficult to reconcile differences and

integrate information with a system that conceptualizes extraction of Negation and extraction of a concept both as positive findings and subject to uncertainty.

BLUE-Text output presents with clear, deterministic and unambiguous assertions about every 'Negatable Concept'. As mentioned in the method section, current BLUE-Text Semantic Model conceptualizes 'Clinical Observation' and 'Locus' as 'Negatable' concepts, sanctioning statements such as 'rash on the face but not on the extremities, no hair-loss' where existence of rash on the extremities (a Locus) and hair-loss (a Clinical Observation) are negated. This means that the question of whether or not a concept is negated is only is valid for two classes of concepts and other concepts by definition can not be negated, and will be inferred as such by the reasoner. For the 'Negatable Concepts' however, an explicit assertion will deterministically indicate Negation (`<subject_observation> <:isNegatedBy> <negation_modifier>`), and will enable retrieval and querying of concepts based on this property.

Figure 44 provides another SPARQL query that illustrates richness and expressivity of the BLUE-Text output where it breaks ground with MedLEE and goes beyond. This example demonstrates how a detailed representation of concepts extracted from clinical text rich with explicit associations identified between them can facilitate complex retrieval tasks that would otherwise be cumbersome and resource intensive if not impossible.

Assume a System B is to use BLUE-Text output to determine health problems attributable to toxicity or poisoning. Although there are many ways of asking such an abstract question from an ontology driven system such as BLUE-Text, here we will look into one approach that exploits the detailed and granular information representation and explicit relationships to satisfy a complex query (Query based reasoning).

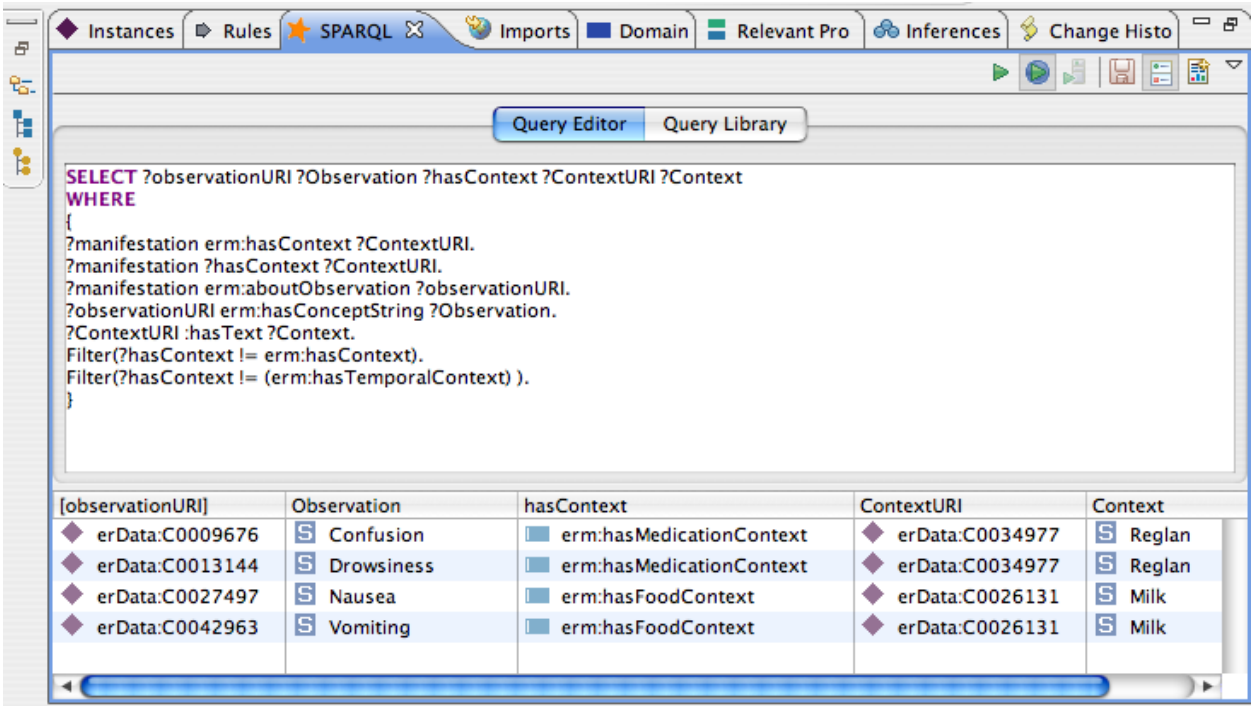


Figure 44. SPARQL query to extract possible evidence for poisoning or toxicity

In this example the query does not make any assumptions on what the meaning of poisoning or toxicity would be and how it can be determined using the output (we will see how it can be done as an example for the contextualization and repurposing BLUE-Text output). However it invokes a simple and generic inquiry to extract all observations that according to the text have a known context and background (other than a temporal background[45]).

The BLUE-Text output explicates :`hasFoodContext` relationships between 'Milk' (as a Food) and 'Nausea' and 'Vomiting' (as Clinical Observations), and :`hasMedicationContext` relationships between 'Regla' (as a Medicine) and 'Drowsiness' and 'Confusion' . It is now upon the System B to sift through the output and filter through :`hasFoodContext` and :`hasMedicationContext` of observed clinical symptoms and make proper assumptions and inferences according the task at hand. However the rich and expressive output presentation

[45] as mentioned in the method section, current prototype implementation of the BLUE-Text does not fully implement temporal reasoning and extraction of the temporal-context.

provided by BLUE-Text could satisfy in an effective and precise way a complex information extraction task that could not be completed by MedLEE output.

Support of Encoding to Standard Vocabularies:

Let's assume a System C that requires extracting all findings about the patient but only using standard terms and their corresponding CUI provided by UMLS, and regardless of how exactly those concepts are expressed in the original text. System C also requires CUIs associated with anatomical locations and their modifiers when possible. System C is a typical case of NLP interoperability where an enterprise system requires systematic processing of clinical text to support a set of operations and tasks that in turn involve interoperation or reporting to some tertiary systems and entities. Use of standard vocabularies in these multi-tier scenarios is required to enable disambiguation and interoperability.

Although MedLEE output provides mapping (encoding) to UMLS CUIs it cannot immediately and completely satisfy requirements of the System C. A custom application needs to be developed to further process MedLEE output by extracting terms and their associated 'code' from the MedLEE output (similar to the System A scenario). The results of this process needs to be further processed to extract the standard terms from the UMLS-MTH and construct the desirable output.

To use BLUE-Text output for this task a SPARQL query similar to the query used to support the System A (Figure 43) is required to express the question in formal terms, this time retrieving information from SKOS-UMLS perspective rather than the Semantic Model view. As these models are linked through explicit relationships, the query engine can successfully navigate the

RDF representation and extract all pertinent information. Similar to the previous scenario only a single SPARQL query can satisfy the task altogether (Figure 45).

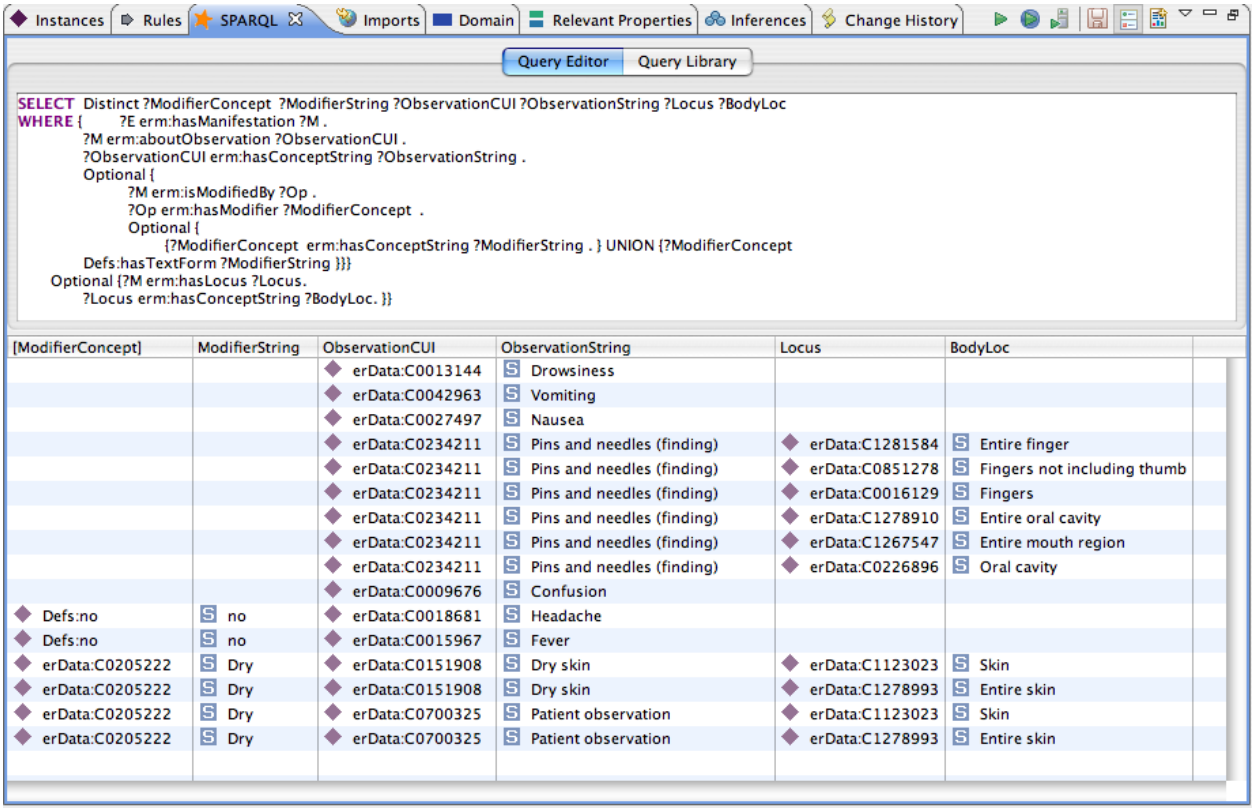


Figure 45. The SPARQL query to satisfy requirements of task C

Note that when it comes to the interpretation of the relationships between the ‘Tingling’ and the ‘Mouth’, MedLEE fails to extract the relationship completely due to a failure by CFG parser to extract ‘Mouth’ as a body location. However, the BLUE-Text has identified 3 different interpretations of ‘Mouth’ and explicated their relationships with the ‘Tingling’. This level of granularity along with a consistent encoding and vocabulary mapping is critical for supporting novel interoperability, customization and contextualization use cases ad-hoc and without requiring intensive interface design, implementation and maintenance activities.

Another dimension of the encoding and vocabulary support by BLUE-Text output representation is the immediate and seamless support of the information retrieval through cross-vocabulary

mappings. An example of this use case is a hypothetical System D that is operationally equivalent to the System C, but requires SNOMEDCT codes rather than UMLS-CUIs. This task can also be seamlessly supported by a single SPARQL query to the output, this time following a link between the Metathesaurus concepts codes associated with them originated from SNOMEDCT as the source vocabulary (Figure 46).

```
SELECT ?ModifierConcept  ?ModifierString ?SABCode ?ObservationString ?Locus
WHERE {
        ?Evidence   erm:hasManifestation   ?Manifestation .
        ?Manifestation   erm:aboutObservation   ?ObservationCUI .
        ?ObservationCUI umls:correspondsTo   ?SABCode.
        ?SABCode     skos:fromConceptSchema   umls:SNOMEDCT.
        ?SABCode erm:hasText ?ObservationString .
        OPTIONAL {
                ?Manifestation erm:isModifiedBy ?Operand .
                ?Operand erm:hasModifier ?ModifierConcept  .
                OPTIONAL {?ModifierConcept  erm:hasText ?ModifierString}
                }
        OPTIONAL {
                ? Manifestation   erm:hasLocus   ?bodyLoc.
                ?bodyLoc umls:correspondsTo ?LocusSABCode.
                ?LocusSABCode      skos:fromConceptSchema  umls:SNOMEDCT.
                ?LocusSABCode erm:hasText ?Locus .
                }
}
```

Figure 46. SPARQL query to demonstrate cross-vocabulary encoding and retrieval

Context and knowledge representation

Frequently users of NLP output need to interpret and draw inferences based on the meaning and implications of terms used in the text. However, the true meaning of a term in a text depends greatly on the other contextual and modifier information. For example, the term ‘Penicillin’ has different connotations in allergy, medication history, home medications, and prescriptions sections of a medical record. As we observed in the example of ‘Vomit’ in the previous

discussions, even encoding to a medical terminology system may not be enough for disambiguation of terms.

The problem can be remedied if the output provides explicit definitions that systems used to identify, extract and encode the terms from the original text, in a way that other systems or human experts can use those definitions to ensure an identical interpretation of the extracted terms. As explained in the beginning of this section, using uniquely identified and explicitly defined concepts rather than character strings to refer to 'terms' is the first step towards this, but it is not enough. An optimal text understanding system output should also provide with the domain and background knowledge such as terminology systems, Lexicons, rules and heuristics, that are being used by the system to precisely define or interpret concepts and associate them with extracted terms.

Frequently, this knowledge is omitted from the output representation based on the assumption that the systems utilizing the output share the same knowledge and context and commit to the same heuristics, and assumptions as the NLP system. Absence of this information can prevent from secondary use of NLP output as the output is not provable and cannot be validated. It also hinders information sharing and contextualization of the output for repurposing and and reuse in new environments.

An optimal output representation should be self-descriptive in a way that enables information systems to access the relevant context, and the semantics of the information represented in the output, in a way that conclusions can be traced back to the associated evidence in the text and the logic and rules of inference in the knowledgebase. For this to happen, not only a formal

interpretive model (ontology) should be available for the querying agents, but also all data (NLP output in this case) should be defined by and mapped to the ontologies.

As mentioned in the previous section, MedLEE output is entirely based on a syntactic convention using superficial features of character strings to conceptualize and information representation model for NLP output. As a result the clinical content within the data structure are depicted as character strings and 'terms' and the data structure providing an schema to embody and contain the clinical content is undeclared and void of any formal semantic that can be used by machine or human for its interpretation.

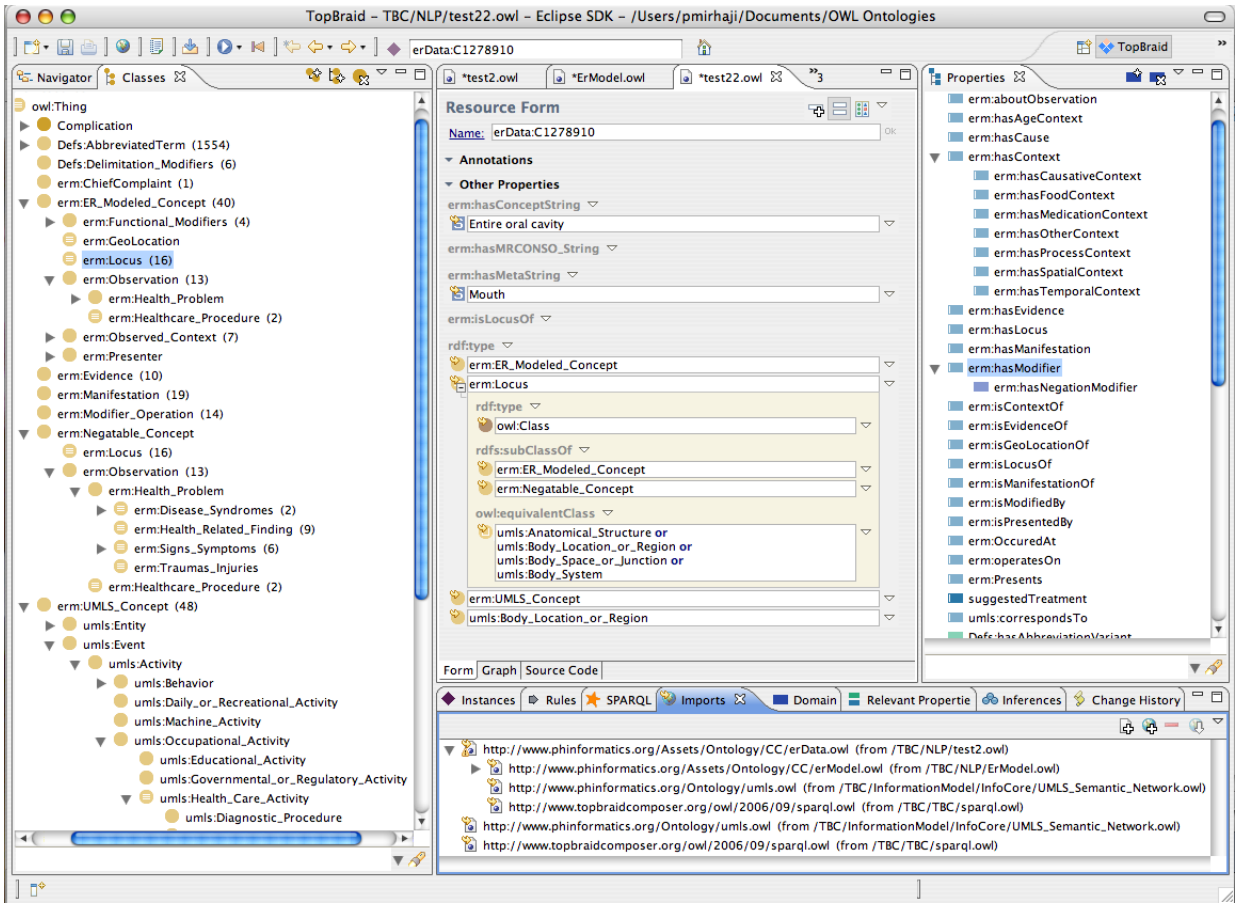


Figure 47. BLUE-Text output as a formal OWL ontology

BLUE-Text provides such ontology driven output representation where all interpretive models (ontologies) are shared and available to the users of the NLP output, and all clinical content

extracted as a result of the text processing is also defined by, and mapped to ontologies, making a self-descriptive model ready for interpretation, integration and contextualization (Figure 47). BLUE-Text output is in effect a formal ontology populated by the instances of concepts and their relationships extracted from the text (Figure 47). It associates every term or literal value in the model with a uniquely and formally defined object (Figure 42) and makes available all context, and knowledge available to the system to put the terms in a precise, unambiguous and explicit context. Figure 47 demonstrates how the term 'Mouth' (a false negative finding by MedLEE) is not only identified by a unique URI `(erData:C1267547)`, but also is defined using concepts such as `erm:Locus`, and `umls:Body_Location_or_Region`. Furthermore it describes the `erm:Locus` in terms of a logical union between several concepts drawn from UMLS-SN. An explicit URL that locates the ontology on the Web is provided (Figure 47, bottom frame) and used to make available the UMLS-SN as a whole and as a reusable ontology. In other words semantics to unambiguously interpret the term 'Mouth' is available through concepts in a formal ontology and explicit links to the location of other knowledge that may be needed to compute all its.

Information Integration and Information Contextualization (reusability)

Integration of heterogeneous and disparate data from multiple clinical databases and sources data is one of the biggest challenges of the modern day health informatics aiming at supporting the clinical translational sciences and research. An optimal information integration method should not only be able to use the syntactic and structural features of the information, but also should be make inferences draw conclusions based on their meaning in order to make associations that are immediately or explicitly available (semantic integration).

For example a physical examination note ‘`patient presents nausea, vomiting, diarrhea, tachycardia and orthostatic hypotension`’, a triage note ‘`patient has dry skin, a swollen tongue and complains from dizziness and drowsiness after several days of fasting`’ and the UMLS CUI ‘`C0011175`’ in a database field for diagnoses list, have common semantic interpretations and should be identified and retrieved by queries searching for cases of ‘Dehydration’.

For the semantic integration to happen several characteristics should be met:

I. First all data and information to be integrated should be clear and unambiguous

II. A rich and expressive representation framework should provide detail with proper granularity in order to support disambiguation and repurposing of data

III. A formal model (ontology) should be available to provide precise and explicit definitions for all information and a reference model for integration.

IV. All information should be mapped to the ontologies in an explicit and provable way, such that a query agent can use the definitions in the ontology and the relationships between concepts and their instances to compute entailments of data as a whole.

Once a system establishes the above criteria, it is expected that semantic applications can interpret, integrate and contextualize information effectively to meet new requirements.

In the past few pages we established that the BLUE-Text output meets all criteria for semantic integration. Here we will follow through a few example use case scenarios using the same output used to compare BLUE-Text output to the MedLEE output.

Let's consider a System E which is interested in extracting all manifestations of a Dehydration from BLUE-Text output. However Dehydration as a unique concept has not been incorporated in the BLUE-Text ontologies. That is the current implementation of BLUE-Text does not classify disorders as 'Dehydration' unless it is explicitly mentioned in the text. However as we observed in the case for the System A dehydration may have many different representations in the text, such as "Dry Skin", or an observation of Dryness located in the 'Skin', both indicating but not explicitly asserting that the patient was dehydrated.

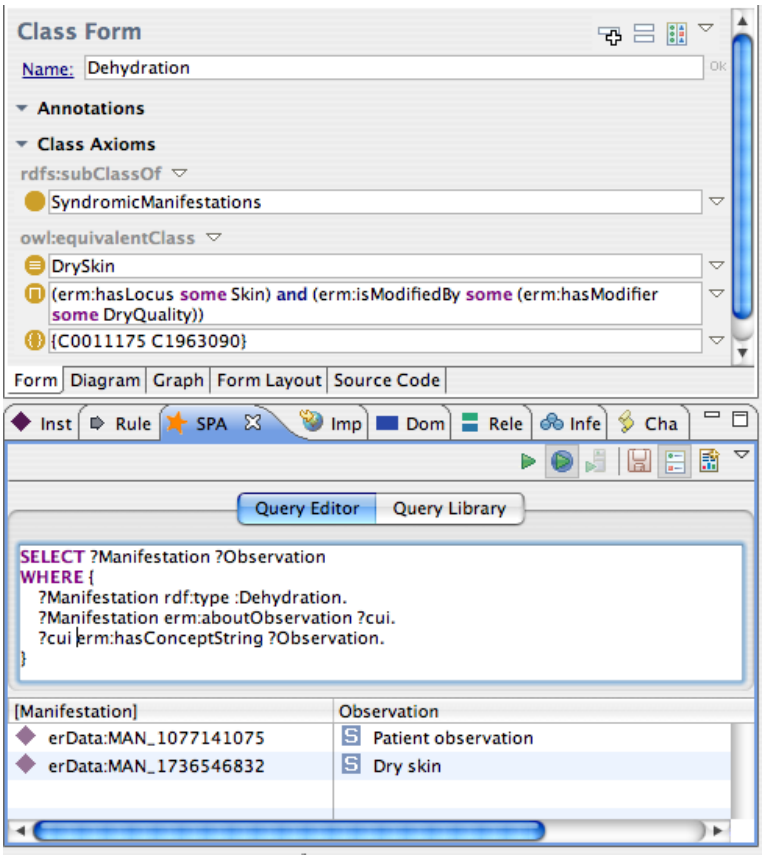


Figure 48: Extension of BLUE-Text output to detect Dehydration

Figure 48 demonstrates how BLUE-Text output can be extended and contextualized to support extraction of a novel concept such as 'Dehydration' that was not present at the time of text processing.

The new concept of “Dehydration” is explicitly defined in the output model as ‘any manifestation of Dry Quality if observed at Skin, or if Dry Skin is explicitly asserted, or if the Dehydration is explicitly asserted, or if CUIs associated with Dehydration are asserted). An SPARQL query then is issued to extract all manifestations of ‘Dehydration’ (?manifestation rdf:type :Dehydration). In figure 48 shows the two different manifestations of the dehydration that are extracted from the text.

To take the example a little bit further, lets consider a System F that requires extraction of cases attributable to food poisoning, and along with the evidence pertaining to the poisoning from the text. This task may requires intensive custom programming and scripting to extract such information effectively using outputs of other text understanding systems. However, using the BLUE-Text output, and because it provides with a formal ontology and a powerful information retrieval mechanism such as SPARQL this task can be handled in 3 easy and intuitive steps and by one single SPARQL query: first we formally define the concept of ‘Food Poisoning’ as in previous example, second we use a reasoner to compute the entailments of the output in light of the new knowledge added to the output, and last we issue an SPARQL query as demonstrated in Figure 49 to extract and present the data as required.

The extension of the BLUE-Text output model in this case first creates a unique concept that integrates different representations that may indicate food poisoning. For instance food poisoning can be directly asserted in the text, it can appear as manifestations with CUIs associated to the food poisoning, or it may be inferred by relationships between Gastrointestinal Syndromes with a known food context. Once the integration concept is defined (Figure 49), the reasoner will compute new relationships and classifications that will be added to the output (inferred statements). An example of such an inferred statement is that our test example is a food

poisoning case per new definition. Then the SPARQL query attempts to navigate the BLUE-Text output to first identify if the text is classified as food poisoning, and if so to find its manifestations using relationships between symptoms, their class membership as inferred by reasoner (Gastrointestinal Syndromes) and their known food context as asserted in the output already.

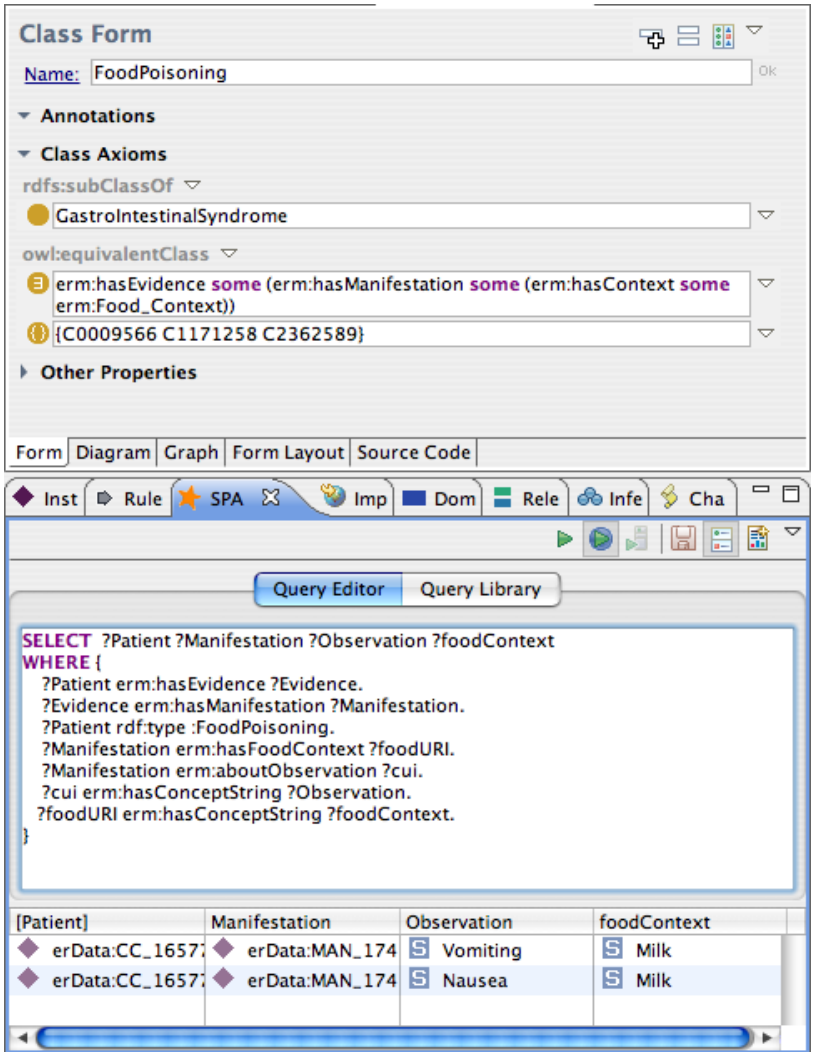


Figure 49. Extensibility of BLUE-Text output to extract Food Poisoning

The process to support tasks of System F could be implemented in less than 20 minutes, including the definitions, reasoning, construction and execution of the SPARQL query.

An interesting feature of such extensions in the output model is that to support new tasks and requirements as exemplified here, we did not actually reprocess the text again. That is, the output of the BLUE-Text and not the system itself was extended to support new uses of existing data. This has big implications in ability to reuse and repurpose legacy data that has been processed by the system once in the past, but still is useful for interpretations and processing for the unprecedented use cases. However the future iterations of the BLUE-Text process are now familiar with new concepts such as food poisoning and dehydration and will produce new output that conforms to and takes into account this new knowledge for future processes.

Is BLUE-Text a Decision Support Tool?

BLUE-Text is an instance of ontology driven (knowledgebased) expert systems conceptualized to use its knowledgebases for contextualization and interpretation of fragments of text extracted by the minimal syntactic parser in the context provided by ontologies available to it. The process if output construction has been designed to accommodate such contextualizations seamlessly and transparently. In thi s chapter we observed several examples of how BLUE-Text ontologies could be extended to interpret the original text in different contexts and for different use cases. With that said there are no limitations on how extensive one can modify BLUE-Text output model and what future use cases it can support. It is important to note that it is also possible to construct a completely different output constructor that formats output and based on a complete different conceptualization and output representation framework introduced here by using the Conceptual Graph as the input.

We will conclude this section with one last example of how the formal and ontology driven representation of the output of text understanding systems can be applied to use clinical text for problem solving and decision support in a clinical research setting. This example of course is

only a hypothetical demonstration and needs to be further refined for real world application. However it highlights several points raised throughout this chapter to communicate and illustrate the semantic and syntactic validity of BLUE-Text output, and evaluate its merits in comparison with other state of the art systems.

Consider System G requiring to find candidate patients eligible to a clinical research project using all data available including clinical notes. The research project needs to find patients that have a history of adverse reaction to 'Dopamine Receptor Antagonist' medications. However, System G also requires to provide proof and extract all evidence that resulted in classification of the patient as a candidate for the study. That is, conclusions of system should be able to be traced back to the evidence used to make the inferences and a black box method is not desirable.

Normally for a definition this broad one needs to use human experts to read and sift through patient information and clinical text, and use their medical expertise to identify indications of adverse reaction to the family of drugs that suppress Dopamine receptors. However, systems such as BLUE-Text provide with an expressive and formal ontology that can be easily extended to contextualize interpretation of text according to new definitions. In this case, all is required to go through a systematic process of defining the family of drugs the meet criteria for 'Dopamine Receptor Antagonist', and define concepts that can indicate adverse reaction to this family of drugs. For illustrative purposes, we also want to be able to provide customized treatment advice as we identify new cases of patients with adverse reaction. That is for each category of problems we provide information about appropriate therapeutic interventions. As a result, the new output model will include concepts such as NeurologicalSyndrome, DrugAdverseReaction, ReactionToDopaminAntagonists, etc. Once the model is extended reasoner needs to compute

entailments of new knowledge and discover new relationships that can be inferred using new knowledge and existing information.

Figure 50 illustrates the formal definition for a class that defines text that meets the above inclusion criteria in terms of what we know about drug adverse reactions, symptoms of reaction to Dopamine Receptor Antagonists, and their representation in clinical text (right hand panel).

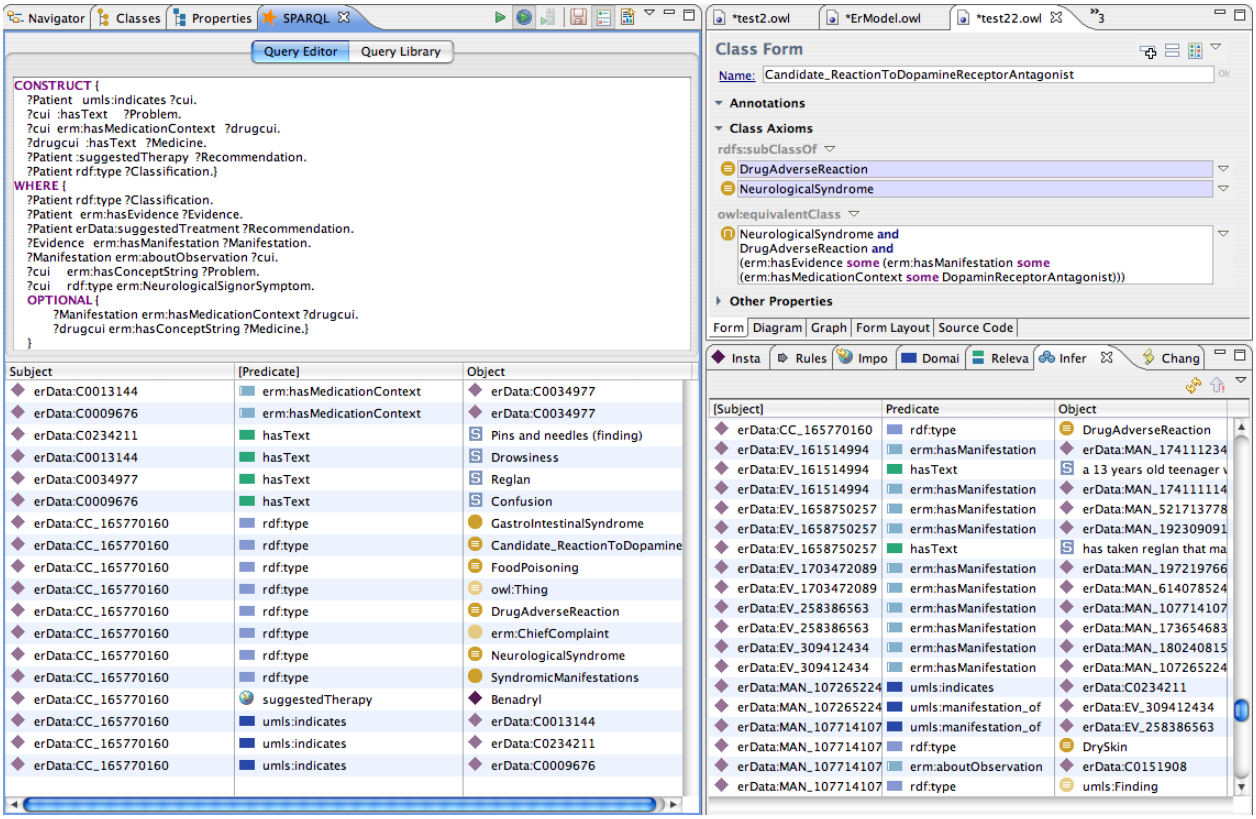


Figure 50. Formal definition and proof for inclusion criteria using BLUE-Text output

In left hand panel (Figure 50) a single SPARQL query is represented that leverages the computer reasoning to extract statements and evidence from the BLUE-Text output that once put together can prove the conclusions made about a given text regarding its appropriateness to the study. It is important to note that this time the query is not extracting some results, but extracting some facts and statements from output (in form of <subject><predicate><objects>) that can be used by any semantic application to draw further conclusions and inferences. In other words, it is possible to

map this output to secondary decision support ontologies and issue a second SPARQL Query that computes implications of facts extracted from the BLUE-Text output in a completely different context related to the protocols and guidelines of the research project.

Section C: UMLS as domain knowledge and terminology

Tables 12, 13, and 14 present with information regarding expectable validity of a text-understanding system that uses UMLS-KS as is, to provide a single source of terminological knowledge for text understanding.

It is important to notice that although the algorithm used for extraction of information related to different UMLS-Semantic Types is identical the performance statistics demonstrates a wide range of variation in both Recall and Precision rates. For example when extracting clinical observations such as `umls:Disease_or_Disorder`, `umls:Injury_or_Poisoning`, or `umls:Sign_or_Symptom` the Recall and Precision rates are excellent (R: 82%-90%; P:97%-100%). However the same algorithm demonstrates an undesirable performance for UMLS-Semantic Types such as `umls:Preventive_or_Therapeutic Procedure`, or `umls:Diagnostic_Procedure` (R: 67%; P: 69%-80%) . This might be attributable to a less than perfect coverage of terminologies associated with medical procedures and significant difference between terms used in the field and those in the biomedical dictionaries.

Another important observation that stands out is the significant reduction of performance in extraction of concepts associated with the `umls:Qualitative_Concept` and `umls:Quantitative_Concepts` (R: 68%; P: 42%). It is clear that UMLS-KS is not a complete and reliable source of knowledge for extraction of such concepts and a modern text

understanding system needs to supply complementary terms and a better classification scheme through a custom developed model and lexicon of modifiers and descriptors.

A more detailed and comprehensive analysis of the reliability of the UMLS for concept extraction and text understanding may require a more rigorous evaluation that controls other related variables such as accuracy of performance of UMLS mapping tools (MMTx in case of BLUE-Text) used to map tokens to UMLS-CUIs, domain specific expressions and patterns that may render use of UMLS in appropriate or insufficient, and similar confounders that were not controlled in this study.

Section D: Limitations of BLUE-Text

BLUE-Text algorithm is implemented as a prototype of semantic text understanding systems that use the Semantic Web as a basis for information and knowledge representation and aim at constructing an accurate representation of clinical text that is as interpretable to artifacts as it is for human experts. However during its implementation we had to make many choices or frequently we were constrained with having no option but to accept what was practically available to us. The quality and performance of the current prototype implementation is not only a result of its design and conceptualization but also the current state of its implementation. For example we explained the pros and cons and impact of deciding between using the SKOS-UMLS as a linked ontology or exposing SKOS-UMLS to the algorithm through an online ontology service. Many times both the theoretical design and implementation of the algorithm was revised to meet the practical realities of the developing Semantic Web applications and technology platforms available to us at the time. Hence discussion of the limitations of the BLUE-Text algorithm can not ignore the state of the affairs with the infrastructure, technology tools and development environment that supports development of Semantic Web applications in general.

In this section we will first start with impediments to implementation of the BLUE-Text algorithm due to the state of the underlying Semantic Web technology platform.

Maturity of the OWL as the Knowledge Representation Language

OWL 1.0 is a rather new and still maturing language, as a result many of the practically useful knowledge representation constructs such as property chaining and many other mathematical attributes of properties are not supported by the current version of the OWL. At the time of this writing OWL 2.0 specifications are under review by W3C and will be up for recommendation and adaptation as standard by mid 2009. The rapidly moving and maturing state of the knowledge representation framework has impacted the BLUE-Text design and implementation form many regards:

At the time of developing BLUE-Text there were no robust and reliable APIs for developing semantic applications using RDF/OWL. Integrated Ontology Editors and Visualization tools were scarce, poorly developed and lacked many practical features required for developing a real world application. Integrated Description Logic and Rules reasoning was not possible with conventional tools and posed extreme technical development challenges despite a theoretically intuitive and trivial design. Scalable and integrated RDF based databases that could serve large ontologies and triple stores were not (and still are not) available and reasoning in light of large ontologies had to be broken down into processes handled by external reasoners implemented accessed through the communication network.

When the technical infrastructure supporting developing Semantic Web applications matures to a point comparable to the conventional information and data processing technologies such as relational and multi-dimensional databases, and APIs and integrated and interactive application

development environments (IDE) start supporting Semantic Web suite of languages and frameworks, the BLUE-Text algorithm can be implemented in a much robust and optimal fashion. This may reduce the limitations and improve the performance of the BLUE-Text algorithm considerably.

D1. Computationally intensive process hinders performance

Due to the problems stated above inherited from the underlying technology platform, the current BLUE-Text algorithm is implemented as patchwork of disparate technologies (e.g., JAVA, .Net, Lisp and Prolog) used to develop applications wrapped, invoked and consumed in a services oriented architecture (SOA) and through a local area network. This adds an overwhelming overhead to the process and slows down the process. On the other hand the minimal syntactic parser algorithm utilized by the BLUE-Text is an iterative and process intensive algorithm that calls MMTx (another process intensive algorithm) as a service and several times. The semantic interpreter uses an external reasoner online, that is every statement added to the model and its entailments are communicated with the semantic interpreter over the network iteratively.

As a result current implementation of the BLUE-Text is a demanding and resource intensive one. It takes an average of 60-70 seconds to process a 25 words sentence one an average conventional server platform (dual 2.8 GHZ processors and 3GB RAM). As the performance statistics regarding computational resources required by other systems is not published and publically available it is not possible to provide a comparison with other systems such as MedLEE at this point. However as the technology platforms and richer more integrated IDEs and reasoners become available, opportunities to improve and mitigate the processing time are becoming more and more viable.

D2. Limited to Clinical Text (patient centric by design)

The current Semantic Model (ontology) utilized by the BLUE-Text is a patient centric model that conceptualizes a one-to-many relationship between a patient and many patient encounters and clinical observations. As a result the BLUE-Text algorithm can not possibly understand clinical content that is centered around a disease (e.g., journal articles related to influenza) or a population (case report of an influenza outbreak). However, in these scenarios the concept extraction algorithm should still be powerful since these types of biomedical content generally provide with a syntactically and structurally good quality text where the BLUE-Text concept extraction algorithm performs excellently.

D3. Single Language (English only so far).

Although the minimal syntactic parser is conceptualized as a language independent parser suitable for all languages with Indo-European typography, scribing form left to right, but all terminological and syntactic knowledge available at this time are constructed using English languages.

It is theoretically possible, however, to construct a new lexicon for any other or multiple indo-European languages at once, especially using UMLS-KS international languages such as Spanish, French and German. The actual validity of BLUE-Text using a language other than English is subject to more rigorous evaluation of the minimal syntactic parser in other languages and objective evaluation of the BLUE-Text algorithm with clinical content prepared using other languages.

D4. Word Sense Disambiguation is modeled as variation.

Capturing all valid variations of a lexical representation is an extremely useful and important feature for a text understanding system intended for a general purpose application and reuse, as demonstrated in the case for Dehydration. BLUE-Text represents such alternate representations as a variation in the output (as seen in the case for 'Dry Skin'). However, current implementation of the BLUE-Text does not implement a word sense disambiguation and normalization function for the output constructor.

Absence of these functions from current implementation has been a tactical decision mainly due to lack of time and resources to implement functions that were not essential for proof of BLUE-Text concept and were out of the scope of this project (BLUE-Text does not have a scientific contribution to make in improving those functions yet, hence we did not spend our time implementing them). The idea is to study existing functions and implement one that suits best with the BLUE-Text architecture.

However we acknowledge that the absence of these functions have impacted the performance statistics and evaluation results presented at Chapter 4. The impact of not having a word sense disambiguation function has been explicating alternate interpretations for the same lexical expression that do not clinically make sense, contribute to the false positive rate and adversely impact the precision of the BLUE-Text extraction algorithm. For example '`minute`' can be interpreted as having the meaning of the time unite 'minute' (60 seconds=1 minute) or it may mean '`small`'. In cases such as '`panic attack 2-3 times a day for 15-20 minutes`' BLUE-Text algorithm attempts to construct an interpretation for minute as '`small`', and may produce assertions meaning 'patient manifest with 15-20 small periods of panic attack'. The minimal syntactic algorithm that generally ignores the grammatical cues and information about

the sentence structure may compound the error rate due to word sense disambiguation. Most word sense disambiguation techniques use these cues and information for disambiguation. Other probabilistic techniques using a sample corpora from the same domain may be more appropriate for systems such as BLUE-Text.

In any case, and despite the fact that one of the critical features of a real world text understanding systems are not included in the current implementation of the BLUE-Text, its overall precision rate is very good and comparable to systems that do include the function.

The lack of normalization algorithm did not have a great negative impact on the performance of BLUE-Text. Normalization has been in fact left out partly by design as BLUE-Text conceptualization is interested in being able to represent as much different representations of the same concept possible. This is critical to support ad-hoc information integration and contextualization as we demonstrated in case for `Dry Skin`. However, it is much more useful if the algorithm could also state a relationship between the different alternatives. For example optimal system should assert that the two different notions of `Dry Skin` in the output are semantically equivalent (or even better if the system could infer that the `Dry Skin`, is a `skos:narrower` and more specialized case of the 'patient observation with dryness in the skin. This may reduce the redundancies and facilitate future extensions of the model without having to account for all possible representations, trusting the reasoner to be able to sift through them automatically.

D5. Complex narratives and BLUE-Text Performance

BLUE-Text performance is less than desirable for extraction of the semantic types associated with therapeutic, preventive or diagnostic procedures. The reason is partly attributable to the

quality and completeness of the UMLS-KS as the domain model and the terminology to describe procedures as we discussed in the previous sections of this chapter.

However a minimal syntactic parser that relies mainly on its domain and semantic ontology to make sense of the Parse Graph, without using grammatical cues and the sentence structute may be bound to failure and errors when the text presents with complex and specific lexical patterns not captured and described by system ontologies. Current semantic and domain ontologies available to the BLUE-Text algorithm are developed as simple prototypes capturing an information model that captures the basic relationships in the clinical text. More complex relationships such as narration of events in the course of a complex procedure (surgical procedures, invasive or non invasive therapeutic or diagnostic processes such as angioplasty, CABG, etc) are not modeled and hence not supported by current implementation of the BLUE-Text.

This by no means implies that BLUE-Text is unreliable for extraction of concepts and important relationships from complex clinical reports and narratives associated with procedures and processes. However, we are expecting that many of the part/whole relationships between a process and its sub-processes, temporal relationships (precedence, successions, etc) and longitudinal account of events (what happened exactly when) will not be captured, since they are not modeled through any of the system ontologies available to the BLUE-Text.

We are expecting that addition of a more robust and domain specific lexicon that includes acronyms, synonyms and shorthand forms specifically used for procedures and processes by itself can improve BLUE-Text performance significantly and to an acceptable level.

Chapter Summary: Semantic, Syntactic and Pragmatic Validity of BLUE-Text

This chapter aimed at interpreting the results of our formal evaluation (chapter 4) and discussing the significance and validity of the output representation framework introduced by the BLUE-Text algorithm, especially as it compares to the output generated by other state of the art text understanding systems.

We have followed a structured and systematic method to establish the validity and quality of the gold-standard and its representativeness for this project (Table 3) and introduced a paradigm rooted in semiology to systematically discuss the Semantic, Syntactic and Pragmatic validity of the BLUE-Text.

The Semantic validity of the BLUE-Text output was mainly discussed through Section A, by a detailed and exhaustive examination of the evaluation results put forward at chapter 4. BLUE-Text demonstrated an excellent overall semantic validity in extraction of biomedical concepts, their anatomical locations, and detecting negation of those concepts. However, due to problems rooted at using the UMLS-KS as the main source of the domain and terminological knowledge, and as a result of some design decisions made in implementing certain features such as word sense disambiguation and normalization BLUE-Text semantic validity needs improvement before being applicable in a real world clinical environment. The extraction and representation of modifier relationships, and clinical procedures were among areas that can be improved by building customized and specialized ontologies to replace or complement use of UMLS-KS in those areas. However an informal comparison of the published results from multiple text understanding systems with the BLUE-Text performance statistics demonstrates a rather comparably good performance in general.

The Syntactic validity of the BLUE-Text output was discussed by using the desiderata introduced in the chapter 2 that introduces characteristics of an optimal output representation framework for text understanding systems to improve information retrieval, information integration, interoperability, contextualization and reuse. We provided evidence that BLUE-Text formal output representation framework provides rich and expressive presentation of its content with rather high detail and with appropriate granularity. It expresses a clear, precise and unambiguous representation that also supports a consistent and effective encoding scheme for mapping between its content and standard vocabularies. Moreover, it is constructed as a formal ontology that incorporates a body of knowledge and contextual information including definitions, and relationships between concept and their relationships. Inclusion of all these detail and knowledge in the formal output enables computer reasoning for discoveries as well as more effective semantic integration and interoperability.

The Pragmatic validity of the BLUE-Text should be evaluated through its application and formal evaluation in the context of some real world practical applications and is beyond the scope of this work. However we provided a rather comprehensive comparison of the Semantic and Syntactic validity of the MedLEE output and the BLUE-Text output in the context of solving 9 hypothetical use case scenarios ranging from NLP Interoperability and information integration to information contextualization and decision support to highlight significance and contributions of BLUE-Text in providing practical, robust and efficient solutions for utilization of clinical content in solving practical clinical and healthcare problems.

Example use cases (Systems A, B, C, D, E, F, and G) provided in this section demonstrate how application of concepts from knowledge engineering, semantic natural language processing and cognitive engineering enables conceptualization of extensible, reusable, and dynamically flexible

text understanding systems that push and break boundaries of NLP beyond its traditional turf (information retrieval) and towards advanced decision support and problem solving using clinical text.

Our discussion also revealed several areas of shortcomings and discovered limitations that highlight need for further research and provide a roadmap for the future work. BLUE-Text implementation is impacted by the rather young age of its underlying technology platform (Semantic Web), and shares the same problems and pain with all other developments in this family of technologies. BLUE-Text current prototype implementation can be overhauled as new APIs and toolsets become available and as the Semantic Web community provides collective answers to its outstanding research and development problems.

We acknowledge that in areas such as explication of the relationships between modifier concepts and descriptors, or detection of the procedures and processes, especially in long and complex narratives BLUE-Text performance is compounded by problems rooted in its conceptualization of a shared model for clinical content. Fortunately this can be easily remedied through extending the semantic model to incorporate such concepts as temporal relationships, and temporal reasoning over clinical events, as well as part/whole relationships between processes and events.

Some underdeveloped aspects of the BLUE-Text algorithm, such as implementation of word sense disambiguation and normalization also require planning and further development in the future. As we will discuss some of the future works and planning in great detail and in the next (and last) chapter of this dissertation, we would like to conclude this chapter with highlighting the outstanding contributions of the BLUE-Text to improve the state of the art in conceptualizing future generation of context aware and knowledge driven text understanding systems.

CHAPTER 6: CONCLUSIONS

Why text-understanding?

Clinical text understanding deals with the conversion of patient health data spoken or recorded as unconstrained text, into formal representations readily interpretable (understandable) by computer programs. This is of interest to health informatics because important information in electronic health records are frequently represented as unconstrained text, and are used extensively by human experts to guide clinical practice, decision making, and to document delivery of care and health status. Furthermore, recent initiatives such as the CTSA program advocating for translational and clinical research call for informatics infrastructure that support semantic integration of all data regardless of their structure and format (including unconstrained text) and enable contextualization and repurposing of the clinical information from electronic health records systems and research databases for multidisciplinary research in a collaborative and distributed environments.

This vision mandates that next generation NLP technologies used in biomedical and clinical environments should be evaluated not only in terms of their validity and reliability in their intended environment, but also in light of their interoperability, and ability to support information sharing, integration and contextualization and reusability in a network of loosely coupled and dynamically changing information processing environment. This vision adds a new layer of information representation requirements for conceptualization of clinical text processing systems, and is the main trust of the BLUE-Text project.

What is the BLUE-Text?

The minimal syntactic, semantic process for clinical text understanding and extraction introduced in this documentation (BLUE-Text) sets out to provide a method of transforming unconstrained and free text information into a computer interpretable (formal) representation that can be used for automation, querying (search and retrieval), information integration, contextualization, reuse, and sharing in a distributed and multidisciplinary environment. . The conceptualization of the BLUE-Text algorithm intends to provide a self-descriptive output that is extensible on-demand, and is measurably resilient to the grammatical and structural irregularities frequently found in the clinical content.

The BLUE-Text employs a broad range of technological frameworks and standards from the Semantic Web community. Our method can also be categorized as an ontology driven method as it extensively uses formal ontologies to inform system behavior. A combination of Description Logic (DL) and rules reasoning is used by the semantic application for classification and inferences.

The minimal syntactic, semantic method comprises of the following main components:

1- A set of ontologies representing syntactic, terminological, semantic and domain knowledge.

2- A text-parsing algorithm for minimal syntactic analysis of clinical content.

3- A semantic interpreter algorithm that uses the above ontologies for ontology mapping, semantic indexing, and output representation (Figure 12).

The BLUE-Text Evaluation Paradigm

In order to describe distinctive features, and characteristics of a computational algorithm as "Text-Understanding" and to measure it objectively, one must present an unambiguous, and clear definition of "Text-understanding", and establish an objective, reproducible and reliable metric to evaluate it. However, the term itself is an ambiguous one and in most cases used interchangeably with other terms such as language processing or text parsing.

BLUE-Text establishes a definition and a corresponding evaluation paradigm using a semiological interpretation of the term 'Text-Understanding" and exploring its semantic, and syntactic implications.

The Semantic Validity of BLUE-Text

Semantically the BLUE-Text project defines the term "understanding" and sets the stage to evaluate it as the ability to identify the relationships between concepts, and between the concepts and entities they refer to. We evaluate the Semantic validity of the BLUE-Text by comparing the agreement between input text and the BLUE-Text output using human interpretation and validation of both the input and the output.

Chapter 4 and discussions of chapter 5 provide an extensive overview of the semantic validity of the BLUE-Text output not only in extracting important clinical concepts such as clinical observations, anatomical concepts, and body locations, but also in explicating meaningful relationships between such concepts such as site of a trauma, severity of a clinical finding, important negative evidences such as lack of a disease, rejection of therapeutic intervention, or stopping an ongoing process.

The evaluation of the BLUE-Text semantic validity demonstrates a strong performance in all categories of evaluation but extraction and explication of qualitative and quantitative concepts (modifiers). This signifies the need to further improvement in both the terminological and syntactic models provided to the BLUE-Text, as well as design and implementation of new complementary processes for the word sense disambiguation and normalization. We also expect that once these new functionalities are implemented, the semantic validity of the BLUE-Text in other areas will improve even further.

The Syntactic Validity of BLUE-Text

Syntactically the BLUE-Text defines the term "text-understanding" and sets the stage for its evaluation as the ability to explicate the output and the relationships between its entities in a formal and computationally interpretable way. The evaluation paradigm for Syntactic validity of the BLUE-Text is based on the desiderata we introduced at Chapter 2 for optimal output representation by text-understanding systems and describes use-cases that can demonstrate appropriateness and efficiencies of the BLUE-Text formal output and its structure to enable optimal information retrieval and information interpretation by computer programs.

We have submitted evidence that the BLUE-Text formal output representation framework provides rich and expressive presentation of its content with high detail and appropriate granularity. It expresses a clear, precise and unambiguous representation that also supports a consistent and effective encoding scheme for mapping between its content and standard vocabularies (UMLS-KS).

The Pragmatic validity of the BLUE-Text should be evaluated through its application in real world practical applications and is beyond the scope of this work. However, in order to highlight significance and contributions of BLUE-Text we also provided a rather comprehensive comparison of the Semantic and Syntactic validity of the MedLEE output (as an exemplary text understanding system with a syntactic approach to the output representation) and the BLUE-Text output (which uses a formal output representation framework) in the context of solving hypothetical use case scenarios ranging from NLP Interoperability and information integration to information contextualization and decision support (the illustrative use cases provided by Systems A, B, C, D, E, F, and G).

Future Works

Although BLUE-Text current state of implementation demonstrates a strong performance comparable to other state of the art concept extraction and text understanding systems such as MedLEE, we acknowledge that there still is a big gap between the desired functionality and performance and what we have observed through this evaluation. BLUE-Text implementation was an extremely valuable experience and learning opportunity to navigate the fine line between theoretical and practical considering constrains of the development environment and the technical infrastructure.

The formal evalutation of the BLUE-Text in the domain of triage notes and chief complaints from emergency department also provided valuable lessons on conceptualization and design of objective measures of validity that are generalizable and applicable in other contexts.

Out experience with BLUE-Text motivates us to plan for its future extensions, and improvements by tapping into lessons learned during the process and through feedback we are expecting to receive after disseminating the information and publishing our results.

Improve Modifier and Locus Algorithm

One immediate area that stands out in evaluation of the Semantic validity of BLUE-Text and presents with an opportunity to improve the current implementation is the overall extraction of Modifier concepts (Qualitative and Quantitative concepts) and their relationships with other concepts. We expect and immediate and dramatic improvement by replacing the terminological knowledge and domain knowledge pertaining to this class of concepts with a customized model that effectively addresses the current problems inherent from direct import of concepts from the UMLS-KS.

Word Sense Disambiguation and Normalization

Due to the constraints of time and resource and as the main thrust of the project has been to address problems of forma output representation results, the word sense disambiguation and Normalization problems have been uniformly implemented as ‘alternative interpretation’. As a result the current evaluation demonstrates a higher than expected False Positives decreasing the observed Precision and Accuracy. This will be remedied when implement a robust disambiguation and normalization feature that maintains only sensible alternative interpretations, and explicates there relative positional and semantic relationships. We expect this will increase both Recall and Precision rates in future iterations of the BLUE-Text implementation. We are also expecting that the Modifiers and Locus relationships will be

dramatically if the normalization and disambiguation processes are accompanied with a specialized terminological knowledge.

Multilingual support

Current UMLS-KS can be used to extract medical terms pertaining to several languages other than English. Although the coverage for languages other than English has been scarce and not updated as frequently as its English language version, this provides and opportunity to use the existing resource to support processing of clinical text in other languages and provide with and incentive to invest and support other languages in the UMLS-KS. It is important to note that the current implementation of BLUE-Text can operate with other indo-European languages only if UMLS-KS supports them (for Encoding and Extraction purposes) and if the current Lexicon (Syntactic Model) is updated with language specific expressions for text parsing and negation.

Support Clinical Document Architecture (CDA) for information exchange

Most health information systems used in healthcare industry are legacy systems incapable of processing formal and semantic output as is. However, industry standards such as HL7 and CDA has been proposed and used to facilitate information exchange across health information systems and are being adopted by most new implementations.

Although CDA and other information exchange frameworks based on HL7 Reference Information Model (RIM) are in the same category as syntactic and non semantic information representation models, but using certain standard conventions and guidelines

make them ‘understandable’ to the degree that information systems agree and commit to them.

On the other hand the formal RDF output and the Conceptual Graph generated by BLUE-Text further contextualization as well as transformation of information in a way that it can meet demands of contextually different systems. One useful extension to the current BLUE-Text output is a task specific constructor that converts Conceptual Graph directly to a CDA compliant XML message ready to be consumed by health information systems that have custom interfaces to receive and process such documents. This enables BLUE-Text to seamlessly present itself and interact with other information systems using CDA as the information model for information exchange.

REFERENCES

(2003). "Unified Medical Language System® (UMLS®).", from URL: http://www.nlm.nih.gov/research/umls/.

Allen, J. (1995). Natural Language Understanding, Benjamin/Cummings.

Anderson, J., C. Aydin, et al. (1994). Evaluating Health Care Information Systems. Thousand Oaks,CA: Sage (1994) by J Anderson, C Aydin, S Jay Thousand Oaks, CA, Sage

Atkinson, K. (2008). "GNU Aspell." from http://aspell.net/.

Berners-Lee, T. (2000). The Semantic Web. The World Wide Web Consortium (W3C) Conference on Semantic Web - XML2000, URL:http://www.w3.org/2000/Talks/1206-xml2k-tbl/.

Burgun, A. (2006). "Desiderata for domain reference ontologies in biomedicine." J Biomed Inform **39**(3): 307-13.

Carpenter, B. (2004). Natural Language Processing, Alias Inc.

Carpenter, B. (2007). "Introduction to Natural Language Processing." from http://nltk.sourceforge.net/lite/doc/en/introduction.html.

Center for Health Research (2006). "Potential Impact of Advanced Clinical Information Technology on Cancer Care in 2015 " Cancer Causes and Control **17**(6): 813-820.

Ceusters, W., B. Smith, et al. (2005). "A terminological and ontological analysis of the NCI Thesaurus." Methods Inf Med **44**(4): 498-507.

Chapman, W. W., J. N. Dowling, et al. (2004). Evaluating Natural Language Processing Applications Applied to Outbreak and Disease Surveillance, The RODS Laboratory.

Chomsky, N. (1956). "Three Models for the Description of Language." IRE Transactions on Information Theory **2**(2): 113–123.

Christensen, L., P. Haug, et al. (2002). MPLUS: A Probabilistic Medical Language Understanding System Workshop On Natural Language Processing In The Biomedical Domain.

Cimino, J. J. (2000). "From data to knowledge through concept-oriented terminologies: experience with the Medical Entities Dictionary." J Am Med Inform Assoc **7**(3): 288-97.

Cimino, J. J. (2006). "In defense of the Desiderata." J Biomed Inform **39**(3): 299-306.

Cook, D. L., J. L. Mejino, et al. (2004). "The foundational model of anatomy: a template for the symbolic representation of multi-scale physiological functions." Conf Proc IEEE Eng Med Biol Soc **7**: 5415-8.

Do Amaral, M. and Y. Satomura (1995). Associating semantic grammars with the SNOMED: Processing medical language and representing clinical facts into a language-independent frame. MEDInfo 95.

Dolin RH, A. L., Beebe C, Biron PV, Boyer SL, Essin D, et al. (2001). "The HL7 clinical document architecture." J Am Med Inform Assoc **8(6)**(552-69).

Friedman, C. (2005). "Natural Language Processing in Medicine: Opportunities & Challenges. Columbia University.", from http://clinicalinformatics.stanford.edu/scci_seminars/slides/stanford_2005.pdf.

Friedman, C., P. Alderson, et al. (1994). "A general natural language text processor for clinical radiology." JAMIA: 161-174.

Friedman, C. and G. Hripcsak (1998). " Evaluating natural language processors in the clinical domain." Methods Inf Med **37**(4-5): 334-44.
Friedman, C. and G. Hripcsak (1999). " Natural language processing and its future in medicine." Acad Med **74**: 890-5.
Friedman, C., G. Hripcsak, et al. (1995). "Natural language processing in an operational clinical information system." J of Nat Lang Eng **1**: 83-108.
Friedman C, J. S., Forman B, Starren J. Proc (1995). Architectural requirements for a multipurpose natural language processor in the clinical environment. Annu Symp Comput Appl Med Care.
Friedman, C., L. Shagina, et al. (2004). "Automated encoding of clinical documents based on Natural Language Processing." J Am Med Inform Assoc
Grishman, R. and R. Kitteredge (1986). Analyzing language in restricted domains: Sublanguage description and processing. NJ, Lawrence Erlbaum.
Grishman, R. and B. Sundheim (1995). Design of the muc-6 evaluation (1995) the Sixth Message Understanding Conference (MUC-6).
Grune, D. and C. H. Jacobs (1990). Parsing Techniques – A Practical Guide, Ellis Horwood. England, Ellis Horwood.
Hahn, U. and M. Romacker (1997). Text structures in medical text processing: empirical evidence and a text understanding prototype. AMIA Annual Fall Symposium 1997.
Hahn, U., M. Romacker, et al. (1999). "Discourse structures in medical reports-watch out! The generation of referentially coherent and valid text knowledge bases in the MEDSYNDIKATE system, ." International Journal of Medical Informatics **53**(1).
Hahn, U., M. Romacker, et al. (2000). "MedSynDiKATe--design considerations for an ontology-based medical text understanding system." Proc AMIA Symp: 330-4.
Hahn, U., M. Romacker, et al. (2002). "Creating knowledge repositories from biomedical reports: the MEDSYNDIKATE text mining system." Pac Symp Biocomput: 338-49.
Hahn, U., M. Romacker, et al. (2002). "MEDSYNDIKATE--a natural language system for the extraction of medical information from findings reports." Int J Med Inform **67**(1-3): 63-74.
Hahn, U. and K. Schnattinge (1997). Deep knowledge discovery from natural language texts., . the 3rd
Intl. Conf. on Knowledge Discovery and Data Mining, Newport Beach, California.
Hamm, R. M. (2002). "Contingency Table Calculator: Probabilities from Counts." from http://www.fammed.ouhsc.edu/robhamm/cdmjavascript/cntngtable.htm.
Harris, Z. (1968). Mathematical Structures of Language. NY, Wiley Interscience.
Heinze, D. T., M. L. Morsch, et al. (2008). "Medical i2b2 NLP smoking challenge: the A-Life system architecture and methodology." J Am Med Inform Assoc **15**(1): 40-3.
Hersh, W. (2005). "Controlled Terminologies in Biomedicine: Rationale, Challenges, and Limitations." from http://medir.ohsu.edu/~hersh/terminology.pdf.
Hobbs, J. R. and F. Pan. (2006). "Time Ontology in OWL (W3C Working Draft 27 September 2006)." from http://www.w3.org/TR/owl-time/.
Horst, H. J. (2004). Completeness, decidability and complexity of entailment for RDF Schema and a semantic extension involving the OWL vocabulary. the Third International Semantic Web Conference (ISWC2004).

Humphreys, B. L., D. A. B. Lindberg, et al. (1998). "Unified Medical Language System®. An Informatics Research Collaboration " Journal of the American Medical Informatics Association **5**: 1-11.

Institute of Medicine (2003). Report on Patient Safety, , Institute of Medicine.

Jain, N. and C. Friedman (1997). Identification of findings suspicious for breast cancer based on natural lan-guage processing of mammogram reports. AMIA 97 Annual Symposium.

Jain, N., C. Knirsch, et al. (1996). Identification of Suspected Tuberculosis Patients based on Natural Language Processing of Chest Radiograph Reports. AMIA 96 Annual Symposium., Philadelphia, Belfus & Hanley.

Kashyap, V. and A. Borgida (2003). Representing the UMLS® Semantic Network using OWL International Semantic Web Conference 2003.

Koehler, S. B. (1998). Symtext: a natural language understanding system for encoding free text medical data, The University of Utah

Li, J. and J. Fine (2004). "On sample size for sensitivity and specificity in prospective diagnostic accuracy studies." Stat Med **23**(16): 2537-50.

Lussier, Y. and C. Patel. (2004). "Re-representing Biomedical Ontologies using the Web Ontology Language. Department of Biomedical Informatics, Columbia University.", from www.sofg.org/meetings/sofg2004/Patel.pdf

Lyman, M., N. Sager, et al. (1991). "The application of natural-language processing to healthcare quality assessment." Med Decis Making. **11**: S65-S68.

Meystre, S. and P. Haug (2006). "Natural language processing to extract medical problems from electronic clinical documents: Performance evaluation." Journal of Biomedical Informatics **39**: 589-599.

Miles, A. and D. Brickley. (Nov 2005). "Simple Knowledge Organisation System (SKOS)." from http://www.w3.org/TR/2005/WD-swbp-skos-core-guide-20051102.

Miller, G. A. (1956). "The Magical Number Seven, Plus or Minus Two: Some Limits on our Capacity for Processing Information." Psychological Review **63**(2): 81-97.

Minsky, M. (1975). A framework for representing knowledge. The Psychology of Computer Vision. P. Winston. New York, McGraw-Hill**:** 211-277.

Mirhaji, P., S. R. Lillibridge, et al. (2004). "Semantic Approach to Public Health Situation Awareness - Design and Methodology." Morbidity and Mortality Weekly Report (MMWR). **53(Suppl)**: 252.

Mirhaji, P., R. L. Richesson, et al. (2004). Knowledge based Public Health Situation Awareness. Sensors, and Command, Control, Communications, and Intelligence (C3I) Technologies for Homeland Security and Homeland Defense III., Orlando, FL., SPIE.

Mirhaji, P., R. L. Richesson, et al. (2004). Public Health Surveillance; a Semantic Approach. Multisensor, Multisource Information Fusion: Architectures, Algorithms, and Applications, SPIE.

Mirhaji, P., J. Zhang, et al. (2005). Situational Awareness in Public Health Preparedness Settings. Sensors, and Command, Control, Communications, and Intelligence (C3I) Technologies for Homeland Security and Homeland Defense (5778).

Mirhaji, P., J. Zhang, et al. (2003). Informatics Critical to Public Health Surveillance. Sensors, and Command, Control, Communications, and Intelligence (C3I)Technologies for Homeland Defense and Law Enforcement II, Orlando, Florida USA., SPIE Publications.

Mirhaji, P., M. Zhu, et al. (2009). "Ontology Driven Integration Platform for Clinical and Translational Research." BMC Bioinformatics **10**(2(Suppl 2)).

Olszewski, R. (2003). Bayesian classification of triage diagnoses for the early detection of epidemics. Recent Advances in Artificial Intelligence. the Sixteenth International FLAIRS Conference.

Ong, K. and Q. H. Wang (1995). "An Object-Oriented Approach to the Semantic Interpretation of Medical Text." Expert Systems with Applications **9**(3): 333-346.

Pearl, J. (1988). Probabilistic inference in intelligent systems. Networks of plausible inference, Morgan Kaufmann.

Ranum, D. L. (1989). Sprus: a knowledge-based understanding system for radiology text, The University of Utah

Rector, A. L., J. E. Rogers, et al. (2003). "OpenGALEN: open source medical terminology and tools." AMIA Annu Symp Proc: 982.

Ricci, R. J. (2002). Future of Healthcare:2012, IBM Healthcare and Life Sciences**:** 1-28.

Sager, N. (1981). Natural Language Processing: A Computer Grammar of English and its Applications., Addison-Wesley.

Sager, N. (1986). Sublanguage: linguistic phenomenon, computational tool. In Grishman R, Kitteredge R, eds. Analyzing language in restricted domains: sublanguage description and processing. NJ, Lawrence Erlbaum.

Sager, N., C. Friedman, et al. (1986). " The analysis and processing of clinical narrative." Medinfo 1101-5.

Sager, N., M. Lyman, et al. (1994). "Natural language processing and the representation of clinical data." JAMIA(1): 142-160.

Sager, N., M. Lyman, et al. (1965). "Linguistic String Project." from http://cs.nyu.edu/cs/projects/lsp/index.html.

Schütze, C. D. M. a. H. (1999). Foundations of Statistical Natural Language Processing. Cambridge, Massachusetts., The MIT Press.

Seidenberg, J. and A. Rector (2006). Web Ontology Segmentation: Analysis, Classification and Use. World Wide Web Conference Committee (IW3C2), Edinburgh, Scotland.

Smith, B. (2006). "From concepts to clinical reality: an essay on the benchmarking of biomedical terminologies." J Biomed Inform **39**(3): 288-98.

Spyns, P. (1996). "Natural language processing in medicine: an overview." Meth Inform Med **35**: 285-301.

Sun Microsystems (2004). Web Services and Service-Oriented Architectures, Sun Microsystems.

Taira, R. K., S. G. Soderland, et al. (2001). "Automatic structuring of radiology free-text reports." Radiographics **21**(1): 237-45.

The College of American Pathologists. (2003). "SNOMED®: Systematized Nomenclature of Medicine.", from URL:http://www.snomed.org/main.html.

Voytovich, A. (1999). "Reduction of medical verbiage: fewer words, more meaning." Annals of Internal Medicine **131**: 146-147.
W3C (2004). Semantic Web emerges as commercial-grade infrastructure for sharing data on the Web.
Wikepedia-Free Online Encyclopedia "Principles of Natural Language Processing."

Appendix A: Definition of Terms

Affinity: the semantic affinity between the two words refers to any similarity, resemblance, likeness, kinship, relationship, association, link, analogy, similitude, or correspondence. 30

Chomsky Hierarchy: Within the field of computer science, specifically in the area of formal languages, the Chomsky hierarchy (occasionally referred to as Chomsky–Schützenberger hierarchy) is a containment hierarchy of classes of formal grammars. This hierarchy of grammars was described by Noam Chomsky in 1956. It is also named after Marcel-Paul Schützenberger who played a crucial role in the development of the theory of formal languages. 38

Conceptual Graph (CG): is a notation for logic based on the existential graphs of Charles Sanders Peirce and the semantic networks of artificial intelligence. In the first published paper on CGs, John F. Sowa used them to represent the conceptual schemas used in database systems. A linear notation, called the Conceptual Graph Interchange Format (CGIF), has been standardized in the ISO standard for Common Logic..... passim

Context Free Grammar: In formal language theory, a context-free grammar (CFG) is a grammar in which every production rule is of the form [V → w] where V is a single nonterminal symbol, and w is a string of terminals and/or nonterminals (possibly empty). Thus, the difference with arbitrary grammars is that the left hand side of a production rule is always a single nonterminal symbol rather than a string of terminal and/or nonterminal symbols. The term ‘context-free’ expresses the fact that nonterminals are rewritten without regard to the context in which they occur. A formal language is context-free if some context-free grammar generates it. These languages are exactly all languages that can be recognized by a non-deterministic pushdown automaton. Context-free grammars play a central role in the description and design of programming languages and compilers. They are also used for analyzing the syntax of natural languages..... 24, 38

Dependency grammar (DG): is a class of syntactic theories developed by Lucien Tesnière. It is distinct from phrase structure grammars, as it lacks phrasal nodes. Structure is determined by the relation between a word (a head) and its dependents. Dependency grammars are not defined by a specific word order, and are thus well suited to languages with freer word order, such as Czech. Algebraic syntax and Extensible Dependency Grammar are types of dependency grammar. Link grammar is similar to dependency grammar, but link grammar includes directionality in the relations between words, as well as lacking a head-dependent relationship. 33, 39, 85

Description logics (DL): are a family of knowledge representation languages which can be used to represent the concept definitions of an application domain (known as terminological knowledge) in a structured and formally well-understood way. The name description logic refers, on the one hand, to concept descriptions used to describe a domain

and, on the other hand, to the logic-based semantics which can be given by a translation into first-order predicate logic. Description logic was designed as an extension to frames and semantic networks, which were not equipped with formal logic-based semantics. They form a middle ground solution: including some more expressive operations than propositional logic and having decidable or more efficient decision problems than first order predicate logic. Description logic was given its current name in the 1980s. Previous to this it was called (chronologically): terminological systems, and concept languages. Today description logic has become a cornerstone of the Semantic Web for its use in the design of ontologies. The OWL-DL and OWL-Lite sub-languages of the W3C-endorsed Web Ontology Language (OWL) are based on a description logic. ..... passim

Frame language: is a metalanguage. It applies the frame concept to the structuring of language properties. Frame languages are usually software languages. Frame languages are rather focused on the recognition and description of objects and classes, and relations and interactions are considered as 'secondary'. In general, 'frame' in this context means 'something that can be/(has to be) fulfilled'. In such sense, for example: Object-oriented programming languages are frame languages, but also every grammar is a frame language. In specific contexts, the authors of computer languages use the term 'frame' arbitrarily and frequently intuitively, and in a metaphoric sense. 27, 43

Hypernymy: the semantic relation in which one word is the hypernym of another. Hypernymy, the relation words stand in when their extensions stand in the relation of class to subclass, should not be confused with holonymy which is the relation words stand in when the things that they denote stand in the relation of whole to part. 9

Hyponymy: In linguistics, a hyponym is a word or phrase whose semantic range is included within that of another word, its hypernym. For example, scarlet, vermilion, carmine, and crimson are all hyponyms of red (their hypernym), which is, in turn, a hyponym of colour.[1] Computer science often terms this relationship an is-a relationship. For example, the phrase Red is a colour can be used to describe the hyponymic relationship between red and colour. 9, 41, 69, 79

KL-ONE is a well known knowledge representation system in the tradition of semantic networks and frames; that is, it is a frame language. The system is an attempt to overcome semantic indistinctness in semantic network representations and to explicitly represent conceptual information as a structured inheritance network. There is a whole family of KL-ONE-like systems. Frames in KL-ONE are called concepts. These form hierarchies using subsume-relations; in the KL-ONE terminology a super class is said to subsume its subclasses. Multiple inheritance is allowed. Actually a concept is said to be well-formed only if it inherits from more than one other concept. All concepts, except the top concept Thing, must have at least one super class. 33, 56, 155

Lexeme: A lexeme is an abstract unit of morphological analysis in linguistics, that roughly corresponds to a set of forms taken by a single word. For example, in the English

language, run, runs, ran and running are forms of the same lexeme, conventionally written as RUN. A related concept is the lemma (or citation form), which is a particular form of a lexeme that is chosen by convention to represent a canonical form of a lexeme. Lemmas are used in dictionaries as the headwords, and other forms of a lexeme are often listed later in the entry if they are not common conjugations of that word. A lexeme belongs to a particular syntactic category, has a certain meaning (semantic value), and in inflecting languages, has a corresponding inflectional paradigm; that is, a lexeme in many languages will have many different forms..... 7, 65, 66

Lexicon: In linguistics, the lexicon (from the Greek: Λεξικόν) of a language is its vocabulary, including its words and expressions. More formally, it is a language's inventory of lexeme TA \l "Lexeme: A lexeme is an abstract unit of morphological analysis in linguistics, that roughly corresponds to a set of forms taken by a single word. For example, in the English language, run, runs, ran and running are forms of the same lexeme, conventionally written as RUN. A related concept is the lemma (or citation form), which is a particular form of a lexeme that is chosen by convention to represent a canonical form of a lexeme. Lemmas are used in dictionaries as the headwords, and other forms of a lexeme are often listed later in the entry if they are not common conjugations of that word. A lexeme belongs to a particular syntactic category, has a certain meaning (semantic value), and in inflecting languages, has a corresponding inflectional paradigm; that is, a lexeme in many languages will have many different forms." \s "lexeme" \c 1 s. The lexicon includes the lexemes used to actualize words. ..... passim

LOOM: is a knowledge representation language developed by researchers in the Artificial Intelligence research group at the University of Southern California's Information Sciences Institute. The Loom project's goal is the development and fielding of advanced tools for knowledge representation and reasoning in Artificial Intelligence. Loom is a language and environment for constructing intelligent applications. At its heart is a knowledge representation system that is used to provide deductive support for the declarative portion of the Loom language. Declarative knowledge in Loom consists of definitions, rules, facts, and default rules. A deductive engine called a classifier utilizes forward-chaining, semantic unification and object-oriented truth maintenance technologies in order to compile the declarative knowledge into a network designed to efficiently support on-line deductive query processing..... 56

Meronymy: (from the Greek words meros = part and onoma = name) is a semantic relation used in linguistics. A meronym denotes a constituent part of, or a member of something. That is, ... 9, 41

Ontology: In the context of computer and information sciences, an ontology defines a set of representational primitives with which to model a domain of knowledge or discourse. The representational primitives are typically classes (or sets), attributes (or properties), and relationships (or relations among class members). The definitions of the representational

primitives include information about their meaning and constraints on their logically consistent application (Tom Gruber). ..... passim

Parse Tree: A parse tree or concrete syntax tree is an (ordered, rooted) tree that represents the syntactic structure of a string according to some formal grammar. In a parse tree, the interior nodes are labeled by non-terminals of the grammar, while the leaf nodes are labeled by terminals of the grammar. A program that produces such trees is called a parser. Parse trees may be generated for sentences in natural languages (see natural language processing), as well as during processing of computer languages, such as programming languages. Parse trees are distinct from abstract syntax trees (also known simply as syntax trees) which are a related concept in compilers. 21

Phrase structure grammar: has several different common meanings: In mathematics and in the area of formal language theory, it is often used as a synonym for context-sensitive grammar, which uses phrase structure rules or rewrite rules. However, it is not a precise term, and may also be used to refer to other classes of grammar in the Chomsky hierarchy that are more powerful than context-free grammars. In linguistics, it refers to any one of several related theories for the parsing of natural language, including the head-driven phrase structure grammar, the lexical functional grammar and the generalised phrase structure grammar. The subject divides clauses into two constituents, the noun phrase (NP) and the verb phrase (VP), reminiscent of the more rudimentary linguistic syntactical analysis, that of subject and predicate. From these the subject works down to the individual words within the sentence, parenthesising phrases according to the clauses' words' form classes in a way universally applicable to the English language..... 27, 39

Polysemy: A polyseme is a word or phrase with multiple, related meanings. A word is judged to be polysemous if it has two senses of the word whose meanings are related. Since the vague concept of relatedness is the test for polysemy, judgments of polysemy can be very difficult to make. 34

Semantic Web: The Semantic Web is an evolving extension of the World Wide Web in which the semantics of information and services on the web is defined, making it possible for the web to understand and satisfy the requests of people and machines to use the web content. It derives from World Wide Web Consortium director Sir Tim Berners-Lee's vision of the Web as a universal medium for data, information, and knowledge exchange. At its core, the semantic web comprises a set of design principles, collaborative working groups, and a variety of enabling technologies. Some elements of the semantic web are expressed as prospective future possibilities that are yet to be implemented or realized. Other elements of the semantic web are expressed in formal specifications. Some of these include Resource Description Framework (RDF), a variety of data interchange formats (e.g. RDF/XML, N3, Turtle, N-Triples), and notations such as RDF Schema (RDFS) and the Web Ontology Language (OWL), all of which are intended to provide a formal description of concepts, terms, and relationships within a given knowledge domain. passim

SPARQL: (pronounced ‘sparkle’) is an RDF query language; its name is a recursive acronym that stands for SPARQL Protocol and RDF Query Language. It is standardized by the RDF Data Access Working Group (DAWG) of the World Wide Web Consortium, and is considered a component of the semantic web. Initially released as a Candidate Recommendation in April 2006, but returned to Working Draft status in October 2006, due to two open issues. In June 2007, SPARQL advanced to Candidate Recommendation once again. On 12 November 2007 the status of SPARQL changed into Proposed Recommendation. On 15 January 2008, SPARQL became an official W3C Recommendation. SPARQL allows for a query to consist of triple patterns, conjunctions, disjunctions, and optional patterns..... passim

SWRL (Semantic Web Rule Language): is a proposal for a Semantic Web rules-language, combining sublanguages of the OWL Web Ontology Language (OWL DL and Lite) with those of the Rule Markup Language (Unary/Binary Datalog). Rules are of the form of an implication between an antecedent (body) and consequent (head). The intended meaning can be read as: whenever the conditions specified in the antecedent hold, then the conditions specified in the consequent must also hold. 147

Synonymy: Synonyms are different words (or sometimes phrases) with identical or very similar meanings. Words that are synonyms are said to be synonymous, and the state of being a synonym is called synonymy. The word comes from Ancient Greek syn (σύν) (‘with’) and onoma (ὄνομα) (‘name'). The words car and automobile are synonyms. Similarly, if we talk about a long time or an extended time, long and extended become synonyms. ..... passim

The Resource Description Framework (RDF): is a family of World Wide Web Consortium (W3C) specifications originally designed as a metadata data model. It has come to be used as a general method for conceptual description or modeling, of information that is implemented in web resources; using a variety of syntax formats. RDF is based upon the idea of making statements about resources, in particular, Web resources, in the form of subject-predicate-object expressions. These expressions are known as triples in RDF terminology. The subject denotes the resource, and the predicate denotes traits or aspects of the resource and expresses a relationship between the subject and the object. ..... passim

Token: A token is any categorized block of text. The block of text corresponding to the token is known as a lexeme. A lexical analyzer processes lexemes to categorize them according to function, giving them meaning. This assignment of meaning is known as tokenization. A token can look like anything; it just needs to be a useful part of the structured text. ..... passim

Valence: the number of grammatical elements with which a particular word, esp. a verb, combines in a sentence. 30, 33, 41

Appendix B: Sample MedLEE Output

```
<MedLEE>
<sentence><structured form="xml">
<finding v="demo"><age v="13 year" idref="p4"/>
<parsemode v="mode3"/>
<sectname v="summary"/>
<sid idref="s1"/></finding>
<problem v="nausea" code="UMLS:C0027497_nausea" idref="p14"><certainty
v="high certainty" idref="p12"/>
<parsemode v="mode2"/>
<sectname v="summary"/>
<sid idref="s1"/><code v="UMLS:C0027497_nausea" idref="p14"/>
</problem>
<problem v="vomit" code="UMLS:C0042963_vomiting" idref="p18"><certainty
v="high certainty" idref="p12"/>
<parsemode v="mode2"/>
<sectname v="summary"/>
<sid idref="s1"/><code v="UMLS:C0042963_vomiting" idref="p18"/>
</problem>

</structured>
<tt><sent id="s1">a <phr id="p4">13 years old</phr> teenager <phr
id="p12">with</phr> <phr id="p14">nausea</phr> <phr id="p16">and</phr>
<phr id="p18">vomitting</phr> after <undef>drinking</undef>
<undef>bad</undef> <undef>milk</undef>.</sent></tt>
</sentence>
<sentence><structured form="xml">
<med v="reglan" code="UMLS:C0034977_Reglan" idref="p34"><certainty
v="high certainty" idref="p32"/>
<parsemode v="mode1"/>
<sectname v="summary"/>
<sid idref="s2"/><code v="UMLS:C0034977_Reglan" idref="p34"/>
</med>
<problem v="drowsiness" code="UMLS:C0013144_drowsiness"
idref="p42"><parsemode v="mode1"/>
<sectname v="summary"/>
<sid idref="s2"/><code v="UMLS:C0013144_drowsiness" idref="p42"/>
</problem>
<problem v="confusion" code="UMLS:C0009676_confusion"
idref="p46"><certainty v="high certainty" idref="p44"/>
<parsemode v="mode1"/>
<sectname v="summary"/>
<sid idref="s2"/><code v="UMLS:C0009676_confusion" idref="p46"/>
</problem>

</structured>
<tt><sent id="s2"> has <phr id="p32">taken</phr> <phr
id="p34">Reglan</phr> that <phr id="p38">made</phr> her <phr
```

```
id="p42">drowsy</phr> <phr id="p44">and</phr> <phr
id="p46">confused</phr>.</sent></tt>
</sentence>
<sentence><structured form="xml">
<problem v="fever" code="UMLS:C0015967_fever" idref="p52"><certainty
v="no" idref="p50"/>
<parsemode v="mode1"/>
<sectname v="summary"/>
<sid idref="s3"/><code v="UMLS:C0015967_fever" idref="p52"/>
</problem>
<problem v="headache" code="UMLS:C0018681_headache"
idref="p56"><certainty v="no" idref="p50"/>
<parsemode v="mode1"/>
<sectname v="summary"/>
<sid idref="s3"/><code v="UMLS:C0018681_headache" idref="p56"/>
</problem>

</structured>
<tt><sent id="s3"> <phr id="p50">no</phr> <phr id="p52">fever</phr>
<phr id="p54">and</phr> <phr id="p56">headache</phr>.</sent></tt>
</sentence>
<sentence><structured form="xml">
<problem v="tingling" code="UMLS:C0423572_pins and needles"
idref="p62"><bodyloc v="finger" code="UMLS:C0016129_finger"
idref="p66"><region v="tip" idref="p68"/>
<code v="UMLS:C0729895_tip of finger" idref="p66 p68"/>
</bodyloc>
<certainty v="moderate certainty" idref="p60"/>
<parsemode v="mode2"/>
<sectname v="summary"/>
<sid idref="s4"/><code v="UMLS:C0850630_tingling fingers" idref="p62
p66"/>
</problem>

</structured>
<tt><sent id="s4"> <phr id="p60">Feels</phr> <phr
id="p62">tingling</phr> on <phr id="p66">finger</phr> <phr
id="p68">tips</phr> and around her mouth.</sent></tt>
</sentence>
<sentence><structured form="xml">
<finding v="dry" idref="p80"><bodyloc v="skin"
code="UMLS:C1123023_skin" idref="p82"><code v="UMLS:C1123023_skin"
idref="p82"/>
</bodyloc>
<parsemode v="mode3"/>
<sectname v="summary"/>
<sid idref="s5"/><code v="UMLS:C0151908_skin dry" idref="p80 p82"/>
</finding>

</structured>
```

```
<tt><sent id="s5"> <phr id="p80">dry</phr> <phr id="p82">skin</phr> in
observation.</sent></tt>
</sentence>
</MedLEE>
```

Appendix C: Sample BLUE-Text RDF output (without imports)

```
<?xml version="1.0"?>
<rdf:RDF
    xmlns="http://www.phinformatics.org/erData.owl#"
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:xsd="xsd:"
    xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
    xmlns:owl="http://www.w3.org/2002/07/owl#"
    xmlns:umls="http://umls.nlm.nih.gov/"
    xmlns:Defs="http://www.phinformatics.org/ccNLP_Definitions.owl#"
    xmlns:daml="http://www.daml.org/2001/03/daml+oil#"
    xmlns:erm="http://www.phinformatics.org/erModel.owl#"
    xmlns:sparql="http://www.topbraidcomposer.org/owl/2006/09/sparql.owl#"
  xml:base="http://www.phinformatics.org/erData.owl" >
  <rdf:Description rdf:about="umls:C0042963">
    <rdf:type rdf:resource="erm:Observation"/>
    <rdf:type rdf:resource="erm:Health_Related_Finding"/>
    <rdf:type rdf:resource="erm:ER_Modeled_Concept"/>
    <erm:hasMetaString
rdf:datatype="xsd:string">Vomitting</erm:hasMetaString>
    <erm:hasConceptString
rdf:datatype="xsd:string">Vomiting</erm:hasConceptString>
    <rdf:type rdf:resource="erm:UMLS_Concept"/>
    <rdf:type rdf:resource="umls:Sign_or_Symptom"/>
  </rdf:Description>
  <rdf:Description rdf:about="#Mod_453904929">
    <erm:hasModifier
rdf:resource="http://www.phinformatics.org/ccNLP_Definitions.owl#no"/>
    <erm:operatesOn rdf:resource="umls:C0015967"/>
    <rdf:type rdf:resource="erm:Modifier_Operation"/>
  </rdf:Description>
  <rdf:Description rdf:about="#MAN_1072652244">
    <erm:hasLocus rdf:resource="umls:C0226896"/>
    <erm:hasLocus rdf:resource="umls:C1267547"/>
    <erm:hasLocus rdf:resource="umls:C1278910"/>
    <erm:hasLocus rdf:resource="umls:C0016129"/>
    <erm:hasLocus rdf:resource="umls:C0851278"/>
    <erm:hasLocus rdf:resource="umls:C1281584"/>
    <erm:aboutObservation rdf:resource="umls:C0234211"/>
    <erm:isManifestationOf rdf:resource="#EV_309412434"/>
    <rdf:type rdf:resource="erm:Manifestation"/>
  </rdf:Description>
  <rdf:Description rdf:about="umls:C0026131">
    <rdf:type rdf:resource="erm:Observed_Context"/>
    <rdf:type rdf:resource="erm:Food_Context"/>
    <rdf:type rdf:resource="erm:ER_Modeled_Concept"/>
```

```
    <erm:hasConceptString
rdf:datatype="xsd:string">Milk</erm:hasConceptString>
    <erm:hasMetaString rdf:datatype="xsd:string">Milk</erm:hasMetaString>
    <rdf:type rdf:resource="umls:Body_Substance"/>
    <rdf:type rdf:resource="erm:UMLS_Concept"/>
    <rdf:type rdf:resource="umls:Food"/>
  </rdf:Description>
  <rdf:Description rdf:about="umls:C0009676">
    <rdf:type rdf:resource="erm:Observation"/>
    <rdf:type rdf:resource="erm:Disease_Syndromes"/>
    <rdf:type rdf:resource="erm:ER_Modeled_Concept"/>
    <erm:hasConceptString
rdf:datatype="xsd:string">Confusion</erm:hasConceptString>
    <erm:hasMetaString
rdf:datatype="xsd:string">Confused</erm:hasMetaString>
    <rdf:type rdf:resource="erm:UMLS_Concept"/>
    <rdf:type rdf:resource="umls:Mental_or_Behavioral_Dysfunction"/>
  </rdf:Description>
  <rdf:Description rdf:about="#EV_1658750257">
    <rdf:type rdf:resource="erm:Evidence"/>
  </rdf:Description>
  </rdf:Description>
  <rdf:Description rdf:about="">
    <owl:imports rdf:resource="http://www.phinformatics.org/erModel.owl"/>
    <owl:imports
rdf:resource="http://www.topbraidcomposer.org/owl/2006/09/sparql.owl"/>
    <owl:imports rdf:resource="umls:"/>
    <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#Ontology"/>
  </rdf:Description>
  <rdf:Description rdf:about="umls:location_of">
    <owl:inverseOf rdf:resource="erm:hasLocus"/>
  </rdf:Description>
  <rdf:Description rdf:about="#MAN_521713778">
    <erm:hasMedicationContext rdf:resource="umls:C0034977"/>
    <erm:aboutObservation rdf:resource="umls:C0013144"/>
    <erm:isManifestationOf rdf:resource="#EV_1658750257"/>
    <rdf:type rdf:resource="erm:Manifestation"/>
  </rdf:Description>
  <rdf:Description rdf:about="umls:C0205169">
    <rdf:type rdf:resource="erm:Functional_Modifiers"/>
    <rdf:type rdf:resource="erm:Qualitative_Modifier"/>
    <rdf:type rdf:resource="erm:ER_Modeled_Concept"/>
    <erm:hasConceptString rdf:datatype="xsd:string">Bad (qualifier
value)</erm:hasConceptString>
    <erm:hasMetaString rdf:datatype="xsd:string">Bad</erm:hasMetaString>
    <rdf:type rdf:resource="erm:UMLS_Concept"/>
    <rdf:type rdf:resource="umls:Qualitative_Concept"/>
  </rdf:Description>
  <rdf:Description rdf:about="umls:C0231290">
    <rdf:type rdf:resource="erm:Observed_Context"/>
    <rdf:type rdf:resource="erm:Temporal_Context"/>
    <rdf:type rdf:resource="erm:ER_Modeled_Concept"/>
```

```
    <erm:hasConceptString rdf:datatype="xsd:string">Status post
(contextual qualifier) (qualifier value)</erm:hasConceptString>
    <erm:hasMetaString rdf:datatype="xsd:string">After</erm:hasMetaString>
    <rdf:type rdf:resource="erm:UMLS_Concept"/>
    <rdf:type rdf:resource="umls:Temporal_Concept"/>
  </rdf:Description>
  <rdf:Description rdf:about="#MAN_1923090911">
    <erm:hasMedicationContext rdf:resource="umls:C0034977"/>
    <erm:aboutObservation rdf:resource="umls:C0009676"/>
    <erm:isManifestationOf rdf:resource="#EV_1658750257"/>
    <rdf:type rdf:resource="erm:Manifestation"/>
  </rdf:Description>
  <rdf:Description rdf:about="umls:C1123023">
    <rdf:type rdf:resource="erm:Locus"/>
    <rdf:type rdf:resource="erm:ER_Modeled_Concept"/>
    <erm:hasConceptString
rdf:datatype="xsd:string">Skin</erm:hasConceptString>
    <erm:hasMetaString rdf:datatype="xsd:string">Skin</erm:hasMetaString>
    <rdf:type rdf:resource="erm:UMLS_Concept"/>
    <rdf:type rdf:resource="umls:Body_Part__Organ__or_Organ_Component"/>
  </rdf:Description>
  <rdf:Description rdf:about="umls:C0013123">
    <rdf:type rdf:resource="erm:UMLS_Concept"/>
    <rdf:type rdf:resource="umls:Organism_Function"/>
  </rdf:Description>
  <rdf:Description rdf:about="#MAN_174111234">
    <erm:aboutObservation rdf:resource="umls:C0042963"/>
    <erm:isModifiedBy rdf:resource="#Mod_2053891702"/>
    <erm:hasTemporalContext rdf:resource="umls:C0231290"/>
    <erm:isManifestationOf rdf:resource="#EV_161514994"/>
    <erm:hasFoodContext rdf:resource="umls:C0026131"/>
    <erm:hasAgeContext rdf:resource="umls:C0001578"/>
    <rdf:type rdf:resource="erm:Manifestation"/>
  </rdf:Description>
  <rdf:Description rdf:about="umls:C1278910">
    <rdf:type rdf:resource="erm:Locus"/>
    <rdf:type rdf:resource="erm:ER_Modeled_Concept"/>
    <erm:hasConceptString rdf:datatype="xsd:string">Entire oral
cavity</erm:hasConceptString>
    <erm:hasMetaString rdf:datatype="xsd:string">Mouth</erm:hasMetaString>
    <rdf:type rdf:resource="erm:UMLS_Concept"/>
    <rdf:type rdf:resource="umls:Body_Location_or_Region"/>
  </rdf:Description>
  <rdf:Description rdf:about="#MAN_1972197664">
    <erm:isModifiedBy rdf:resource="#Mod_453904929"/>
    <erm:aboutObservation rdf:resource="umls:C0015967"/>
    <erm:isManifestationOf rdf:resource="#EV_1703472089"/>
    <rdf:type rdf:resource="erm:Manifestation"/>
  </rdf:Description>
  <rdf:Description rdf:about="erm:NeurologicalSignorSymptom">
    <rdfs:subClassOf rdf:resource="erm:Signs_Symptoms"/>
    <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#Class"/>
  </rdf:Description>
```

```
  <rdf:Description rdf:about="umls:C0034977">
    <rdf:type rdf:resource="erm:Observed_Context"/>
    <rdf:type rdf:resource="erm:Medication_Context"/>
    <rdf:type rdf:resource="erm:ER_Modeled_Concept"/>
    <erm:hasConceptString
rdf:datatype="xsd:string">Reglan</erm:hasConceptString>
    <erm:hasMetaString
rdf:datatype="xsd:string">Reglan</erm:hasMetaString>
    <rdf:type rdf:resource="umls:Organic_Chemical"/>
    <rdf:type rdf:resource="erm:UMLS_Concept"/>
    <rdf:type rdf:resource="umls:Pharmacologic_Substance"/>
  </rdf:Description>
  <rdf:Description rdf:about="umls:C0302523">
    <rdf:type rdf:resource="erm:UMLS_Concept"/>
    <rdf:type rdf:resource="umls:Research_Activity"/>
  </rdf:Description>
  <rdf:Description rdf:about="#Mod_202233976">
    <erm:hasModifier rdf:resource="umls:C0205222"/>
    <erm:operatesOn rdf:resource="umls:C1278993"/>
    <rdf:type rdf:resource="erm:Modifier_Operation"/>
  </rdf:Description>
  <rdf:Description rdf:about="umls:C1278993">
    <rdf:type rdf:resource="erm:Locus"/>
    <rdf:type rdf:resource="erm:ER_Modeled_Concept"/>
    <erm:hasConceptString rdf:datatype="xsd:string">Entire
skin</erm:hasConceptString>
    <erm:hasMetaString rdf:datatype="xsd:string">Skin</erm:hasMetaString>
    <rdf:type rdf:resource="erm:UMLS_Concept"/>
    <rdf:type rdf:resource="umls:Body_System"/>
  </rdf:Description>
  <rdf:Description rdf:about="umls:C0234211">
    <erm:hasConceptString rdf:datatype="xsd:string">Pins and needles
(finding)</erm:hasConceptString>
    <erm:hasMetaString
rdf:datatype="xsd:string">Tingling</erm:hasMetaString>
    <rdf:type rdf:resource="erm:NeurologicalSignorSymptom"/>
  </rdf:Description>
  <rdf:Description rdf:about="umls:C1267547">
    <rdf:type rdf:resource="erm:Locus"/>
    <rdf:type rdf:resource="erm:ER_Modeled_Concept"/>
    <erm:hasConceptString rdf:datatype="xsd:string">Entire mouth
region</erm:hasConceptString>
    <erm:hasMetaString rdf:datatype="xsd:string">Mouth</erm:hasMetaString>
    <rdf:type rdf:resource="erm:UMLS_Concept"/>
    <rdf:type rdf:resource="umls:Body_Location_or_Region"/>
  </rdf:Description>
  <rdf:Description rdf:about="umls:C0205222">
    <rdf:type rdf:resource="erm:Functional_Modifiers"/>
    <rdf:type rdf:resource="erm:Qualitative_Modifier"/>
    <rdf:type rdf:resource="erm:ER_Modeled_Concept"/>
    <erm:hasConceptString
rdf:datatype="xsd:string">Dry</erm:hasConceptString>
    <erm:hasMetaString rdf:datatype="xsd:string">Dry</erm:hasMetaString>
```

```
    <rdf:type rdf:resource="erm:UMLS_Concept"/>
    <rdf:type rdf:resource="umls:Qualitative_Concept"/>
  </rdf:Description>
  <rdf:Description rdf:about="#MAN_1741111145">
    <erm:hasAgeContext rdf:resource="umls:C0001578"/>
    <erm:hasFoodContext rdf:resource="umls:C0026131"/>
    <erm:isManifestationOf rdf:resource="#EV_161514994"/>
    <erm:aboutObservation rdf:resource="umls:C0027497"/>
    <erm:isModifiedBy rdf:resource="#Mod_2053891702"/>
    <erm:hasTemporalContext rdf:resource="umls:C0231290"/>
    <rdf:type rdf:resource="erm:Manifestation"/>
  </rdf:Description>
  <rdf:Description rdf:about="#CC_165770160">
    <erm:hasCCText rdf:datatype="xsd:string">a 13 years old teenager with
nausea and vomitting after drinking bad milk. has taken Reglan that made
her drowsy and confused. no fever and headache. Feels tingling on finger
tips and around his mouth. dry skin in observation.</erm:hasCCText>
    <erm:hasEvidence rdf:resource="#EV_309412434"/>
    <erm:hasEvidence rdf:resource="#EV_1703472089"/>
    <erm:hasEvidence rdf:resource="#EV_1658750257"/>
    <erm:hasEvidence rdf:resource="#EV_161514994"/>
    <erm:hasEvidence rdf:resource="#EV_258386563"/>
    <rdf:type rdf:resource="erm:ChiefComplaint"/>
  </rdf:Description>
  <rdf:Description rdf:about="#EV_258386563">
    <rdf:type rdf:resource="erm:Evidence"/>
  </rdf:Description>
  <rdf:Description rdf:about="#Mod_2053891702">
    <erm:hasModifier rdf:resource="umls:C0205169"/>
    <erm:operatesOn rdf:resource="umls:C0026131"/>
    <rdf:type rdf:resource="erm:Modifier_Operation"/>
  </rdf:Description>
  <rdf:Description rdf:about="#Mod_1682377644">
    <erm:hasModifier rdf:resource="umls:C0205222"/>
    <erm:operatesOn rdf:resource="umls:C1278993"/>
    <rdf:type rdf:resource="erm:Modifier_Operation"/>
  </rdf:Description>
  <rdf:Description rdf:about="#EV_1703472089">
    <rdf:type rdf:resource="erm:Evidence"/>
  </rdf:Description>
  <rdf:Description rdf:about="umls:C0700325">
    <rdf:type rdf:resource="erm:Observation"/>
    <rdf:type rdf:resource="erm:Healthcare_Procedure"/>
    <rdf:type rdf:resource="erm:ER_Modeled_Concept"/>
    <erm:hasConceptString rdf:datatype="xsd:string">Patient
observation</erm:hasConceptString>
    <erm:hasMetaString
rdf:datatype="xsd:string">observation</erm:hasMetaString>
    <rdf:type rdf:resource="erm:UMLS_Concept"/>
    <rdf:type rdf:resource="umls:Therapeutic_or_Preventive_Procedure"/>
  </rdf:Description>
  <rdf:Description rdf:about="umls:C0013144">
```

```
    <erm:hasConceptString
rdf:datatype="xsd:string">Drowsiness</erm:hasConceptString>
    <erm:hasMetaString
rdf:datatype="xsd:string">Drowsy</erm:hasMetaString>
    <rdf:type rdf:resource="erm:NeurologicalSignorSymptom"/>
  </rdf:Description>
  <rdf:Description rdf:about="umls:C0226896">
    <rdf:type rdf:resource="erm:Locus"/>
    <rdf:type rdf:resource="erm:ER_Modeled_Concept"/>
    <erm:hasConceptString rdf:datatype="xsd:string">Oral
cavity</erm:hasConceptString>
    <erm:hasMetaString rdf:datatype="xsd:string">Mouth</erm:hasMetaString>
    <rdf:type rdf:resource="erm:UMLS_Concept"/>
    <rdf:type rdf:resource="umls:Body_Part__Organ__or_Organ_Component"/>
  </rdf:Description>
  <rdf:Description rdf:about="umls:C0015967">
    <rdf:type rdf:resource="erm:Health_Related_Finding"/>
    <rdf:type rdf:resource="erm:Observation"/>
    <rdf:type rdf:resource="erm:Signs_Symptoms"/>
    <rdf:type rdf:resource="erm:ER_Modeled_Concept"/>
    <erm:hasConceptString
rdf:datatype="xsd:string">Fever</erm:hasConceptString>
    <erm:hasMetaString rdf:datatype="xsd:string">Fever</erm:hasMetaString>
    <rdf:type rdf:resource="erm:UMLS_Concept"/>
    <rdf:type rdf:resource="umls:Sign_or_Symptom"/>
  </rdf:Description>
  <rdf:Description rdf:about="umls:manifestation_of">
    <owl:equivalentProperty rdf:resource="erm:isManifestationOf"/>
    <owl:inverseOf rdf:resource="erm:hasManifestation"/>
  </rdf:Description>
  <rdf:Description rdf:about="umls:C0851278">
    <rdf:type rdf:resource="erm:Locus"/>
    <rdf:type rdf:resource="erm:ER_Modeled_Concept"/>
    <erm:hasConceptString rdf:datatype="xsd:string">Fingers not including
thumb</erm:hasConceptString>
    <erm:hasMetaString
rdf:datatype="xsd:string">Finger</erm:hasMetaString>
    <rdf:type rdf:resource="erm:UMLS_Concept"/>
    <rdf:type rdf:resource="umls:Body_Part__Organ__or_Organ_Component"/>
  </rdf:Description>
  <rdf:Description rdf:about="#suggestedTreatment">
    <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#ObjectProperty"/>
  </rdf:Description>
  <rdf:Description rdf:about="#Mod_1301669563">
    <erm:hasModifier
rdf:resource="http://www.phinformatics.org/ccNLP_Definitions.owl#no"/>
    <erm:operatesOn rdf:resource="umls:C0018681"/>
    <rdf:type rdf:resource="erm:Modifier_Operation"/>
  </rdf:Description>
  <rdf:Description rdf:about="#EV_309412434">
    <rdf:type rdf:resource="erm:Evidence"/>
  </rdf:Description>
```

```
  <rdf:Description rdf:about="#MAN_614078524">
    <erm:isModifiedBy rdf:resource="#Mod_1301669563"/>
    <erm:aboutObservation rdf:resource="umls:C0018681"/>
    <erm:isManifestationOf rdf:resource="#EV_1703472089"/>
    <rdf:type rdf:resource="erm:Manifestation"/>
  </rdf:Description>
  <rdf:Description rdf:about="umls:C0684271">
    <rdf:type rdf:resource="erm:UMLS_Concept"/>
    <rdf:type rdf:resource="umls:Organism_Function"/>
  </rdf:Description>
  <rdf:Description rdf:about="umls:indicates">
    <owl:equivalentProperty rdf:resource="erm:aboutObservation"/>
  </rdf:Description>
  <rdf:Description rdf:about="umls:C0027497">
    <rdf:type rdf:resource="erm:Observation"/>
    <rdf:type rdf:resource="erm:Health_Related_Finding"/>
    <rdf:type rdf:resource="erm:ER_Modeled_Concept"/>
    <erm:hasConceptString
rdf:datatype="xsd:string">Nausea</erm:hasConceptString>
    <erm:hasMetaString
rdf:datatype="xsd:string">Nausea</erm:hasMetaString>
    <rdf:type rdf:resource="erm:UMLS_Concept"/>
    <rdf:type rdf:resource="umls:Finding"/>
  </rdf:Description>
  <rdf:Description rdf:about="umls:C0001578">
    <rdf:type rdf:resource="erm:Observed_Context"/>
    <rdf:type rdf:resource="erm:Age_Context"/>
    <rdf:type rdf:resource="erm:ER_Modeled_Concept"/>
    <erm:hasConceptString
rdf:datatype="xsd:string">Adolescence</erm:hasConceptString>
    <erm:hasMetaString
rdf:datatype="xsd:string">teenager</erm:hasMetaString>
    <rdf:type rdf:resource="erm:UMLS_Concept"/>
    <rdf:type rdf:resource="umls:Age_Group"/>
  </rdf:Description>
  <rdf:Description rdf:about="#Mod_739632424">
    <erm:hasModifier rdf:resource="umls:C0205222"/>
    <erm:operatesOn rdf:resource="umls:C1123023"/>
    <rdf:type rdf:resource="erm:Modifier_Operation"/>
  </rdf:Description>
  <rdf:Description rdf:about="umls:C0001948">
    <rdf:type rdf:resource="erm:UMLS_Concept"/>
    <rdf:type rdf:resource="umls:Individual_Behavior"/>
  </rdf:Description>
  <rdf:Description rdf:about="umls:C1281584">
    <rdf:type rdf:resource="erm:Locus"/>
    <rdf:type rdf:resource="erm:ER_Modeled_Concept"/>
    <erm:hasConceptString rdf:datatype="xsd:string">Entire
finger</erm:hasConceptString>
    <erm:hasMetaString
rdf:datatype="xsd:string">Finger</erm:hasMetaString>
    <rdf:type rdf:resource="erm:UMLS_Concept"/>
    <rdf:type rdf:resource="umls:Body_Part__Organ__or_Organ_Component"/>
```

```
  </rdf:Description>
  <rdf:Description rdf:about="umls:affects">
    <owl:inverseOf rdf:resource="erm:hasModifier"/>
  </rdf:Description>
  <rdf:Description rdf:about="#EV_161514994">
    <Defs:hasTextForm rdf:datatype="xsd:string">a 13 years old teenager
with nausea and vomitting after drinking bad milk</Defs:hasTextForm>
    <rdf:type rdf:resource="erm:Evidence"/>
  </rdf:Description>
  <rdf:Description rdf:about="#MAN_1077141075">
    <erm:hasLocus rdf:resource="umls:C1278993"/>
    <erm:isModifiedBy rdf:resource="#Mod_202233976"/>
    <erm:hasLocus rdf:resource="umls:C1123023"/>
    <erm:isModifiedBy rdf:resource="#Mod_739632424"/>
    <erm:aboutObservation rdf:resource="umls:C0700325"/>
    <erm:isManifestationOf rdf:resource="#EV_258386563"/>
    <rdf:type rdf:resource="erm:Manifestation"/>
  </rdf:Description>
  <rdf:Description rdf:about="#MAN_1802408159">
    <erm:hasLocus rdf:resource="umls:C0226896"/>
    <erm:hasLocus rdf:resource="umls:C1267547"/>
    <erm:hasLocus rdf:resource="umls:C1278910"/>
    <erm:hasLocus rdf:resource="umls:C0016129"/>
    <erm:hasLocus rdf:resource="umls:C0851278"/>
    <erm:hasLocus rdf:resource="umls:C1281584"/>
    <erm:isManifestationOf rdf:resource="#EV_309412434"/>
    <rdf:type rdf:resource="erm:Manifestation"/>
  </rdf:Description>
  <rdf:Description rdf:about="#Mod_1178085293">
    <erm:hasModifier rdf:resource="umls:C0205222"/>
    <erm:operatesOn rdf:resource="umls:C1123023"/>
    <rdf:type rdf:resource="erm:Modifier_Operation"/>
  </rdf:Description>
  <rdf:Description rdf:about="umls:C0016129">
    <rdf:type rdf:resource="erm:Locus"/>
    <rdf:type rdf:resource="erm:ER_Modeled_Concept"/>
    <erm:hasConceptString
rdf:datatype="xsd:string">Fingers</erm:hasConceptString>
    <erm:hasMetaString
rdf:datatype="xsd:string">Finger</erm:hasMetaString>
    <rdf:type rdf:resource="erm:UMLS_Concept"/>
    <rdf:type rdf:resource="umls:Body_Part__Organ__or_Organ_Component"/>
  </rdf:Description>
  <rdf:Description rdf:about="umls:C0151908">
    <rdf:type rdf:resource="erm:Health_Related_Finding"/>
    <rdf:type rdf:resource="erm:Observation"/>
    <rdf:type rdf:resource="erm:Signs_Symptoms"/>
    <rdf:type rdf:resource="erm:ER_Modeled_Concept"/>
    <erm:hasConceptString rdf:datatype="xsd:string">Dry
skin</erm:hasConceptString>
    <erm:hasMetaString rdf:datatype="xsd:string">Dry
skin</erm:hasMetaString>
    <rdf:type rdf:resource="erm:UMLS_Concept"/>
```

```
    <rdf:type rdf:resource="umls:Sign_or_Symptom"/>
  </rdf:Description>
  <rdf:Description rdf:about="#MAN_1736546832">
    <erm:hasLocus rdf:resource="umls:C1278993"/>
    <erm:isModifiedBy rdf:resource="#Mod_1682377644"/>
    <erm:hasLocus rdf:resource="umls:C1123023"/>
    <erm:isModifiedBy rdf:resource="#Mod_1178085293"/>
    <erm:aboutObservation rdf:resource="umls:C0151908"/>
    <erm:isManifestationOf rdf:resource="#EV_258386563"/>
    <rdf:type rdf:resource="erm:Manifestation"/>
  </rdf:Description>
  <rdf:Description rdf:about="umls:C0018681">
    <rdf:type rdf:resource="erm:Health_Related_Finding"/>
    <rdf:type rdf:resource="erm:Observation"/>
    <rdf:type rdf:resource="erm:Signs_Symptoms"/>
    <rdf:type rdf:resource="erm:ER_Modeled_Concept"/>
    <erm:hasConceptString
rdf:datatype="xsd:string">Headache</erm:hasConceptString>
    <erm:hasMetaString
rdf:datatype="xsd:string">Headache</erm:hasMetaString>
    <rdf:type rdf:resource="erm:UMLS_Concept"/>
    <rdf:type rdf:resource="umls:Sign_or_Symptom"/>
  </rdf:Description>
</rdf:RDF>
```

Appendix D BLUE-Text RDF Output (N3 Format)

```
# Saved by TopBraid on Tue Jan 27 01:49:15 CST 2009
# baseURI: http://www.phinformatics.org/Assets/Ontology/CC/erData.owl
# imports: http://www.topbraidcomposer.org/owl/2006/09/sparql.owl
# imports: http://www.phinformatics.org/Assets/Ontology/CC/erModel.owl
# imports: http://www.phinformatics.org/Ontology/umls.owl

@prefix erm:
<http://www.phinformatics.org/Assets/Ontology/CC/erModel.owl#> .
@prefix umls:    <http://www.phinformatics.org/Ontology/umls.owl#> .
@prefix xsd:     <http://www.w3.org/2001/XMLSchema#> .
@prefix Defs:
<http://www.phinformatics.org/Assets/Ontology/CC/ccNLP_Definitions.owl#> .
@prefix sparql:  <http://www.topbraidcomposer.org/owl/2006/09/sparql.owl#>
.
@prefix rdfs:    <http://www.w3.org/2000/01/rdf-schema#> .
@prefix daml:    <http://www.daml.org/2001/03/daml+oil#> .
@prefix rdf:     <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix owl:     <http://www.w3.org/2002/07/owl#> .
@prefix :
<http://www.phinformatics.org/Assets/Ontology/CC/erData.owl#> .

:C0042963
      a       erm:Observation , erm:UMLS_Concept , erm:ER_Modeled_Concept
, erm:Health_Related_Finding , umls:Sign_or_Symptom ;
      erm:hasConceptString
              "Vomiting"^^xsd:string ;
      erm:hasMetaString "Vomiting"^^xsd:string .

:Mod_453904929
      a       erm:Modifier_Operation ;
      erm:hasModifier Defs:no ;
      erm:operatesOn :C0015967 .

:MAN_1072652244
      a       erm:Manifestation ;
      erm:aboutObservation
              :C0234211 ;
      erm:hasLocus :C1267547 , :C1278910 , :C0016129 , :C0226896 ,
:C0851278 , :C1281584 ;
      erm:isManifestationOf
              :EV_309412434 .

:C0026131
      a       erm:Observed_Context , erm:UMLS_Concept ,
erm:ER_Modeled_Concept , erm:Food_Context , umls:Body_Substance ,
umls:Food ;
      erm:hasConceptString
              "Milk"^^xsd:string ;
      erm:hasMetaString "Milk"^^xsd:string .
```

```
:C0009676
      a       erm:Observation , erm:UMLS_Concept , erm:ER_Modeled_Concept
, erm:Disease_Syndromes , umls:Mental_or_Behavioral_Dysfunction ;
      erm:hasConceptString
              "Confusion"^^xsd:string ;
      erm:hasMetaString "Confused"^^xsd:string .

:EV_1658750257
      a       erm:Evidence ;
      Defs:hasTextForm "has taken reglan that made her drowsy and
confused"^^xsd:string .

owl:Thing
      sparql:query """Select
?Obs ?observationName ?tc  ?ttext ?fc ?ftext
where {
?s ?tc ?to.
?s ?fc ?fo.
?to rdf:type erm:Temporal_Context.
?to erm:hasMetaString ?ttext.
?fo rdf:type erm:Food_Context.
?fo erm:hasMetaString ?ftext.
?s erm:aboutObservation ?Obs.
?Obs erm:hasMetaString ?observationName.
filter(?tc != erm:hasContext).
filter(?fc != erm:hasContext).
}"""^^xsd:string .

<http://www.phinformatics.org/Assets/Ontology/CC/erData.owl>
      a       owl:Ontology ;
      owl:imports
<http://www.phinformatics.org/Assets/Ontology/CC/erModel.owl> ,
<http://www.topbraidcomposer.org/owl/2006/09/sparql.owl> ,
<http://www.phinformatics.org/Ontology/umls.owl> .

umls:location_of
      owl:inverseOf erm:hasLocus .

:MAN_521713778
      a       erm:Manifestation ;
      erm:aboutObservation
              :C0013144 ;
      erm:hasMedicationContext
              :C0034977 ;
      erm:isManifestationOf
              :EV_1658750257 .

:C0205169
      a       erm:UMLS_Concept , erm:ER_Modeled_Concept ,
erm:Functional_Modifiers , erm:Qualitative_Modifier ,
umls:Qualitative_Concept ;
      erm:hasConceptString
```

```
                "Bad (qualifier value)"^^xsd:string ;
        erm:hasMetaString "Bad"^^xsd:string .

:C0231290
        a       erm:Observed_Context , erm:Temporal_Context ,
erm:UMLS_Concept , erm:ER_Modeled_Concept , umls:Temporal_Concept ;
        erm:hasConceptString
                "Status post (contextual qualifier) (qualifier
value)"^^xsd:string ;
        erm:hasMetaString "After"^^xsd:string .

:MAN_1923090911
        a       erm:Manifestation ;
        erm:aboutObservation
                :C0009676 ;
        erm:hasMedicationContext
                :C0034977 ;
        erm:isManifestationOf
                :EV_1658750257 .

:C1123023
        a       erm:UMLS_Concept , erm:ER_Modeled_Concept , erm:Locus ,
umls:Body_Part__Organ__or_Organ_Component ;
        erm:hasConceptString
                "Skin"^^xsd:string ;
        erm:hasMetaString "Skin"^^xsd:string .

:C0013123
        a       erm:UMLS_Concept , umls:Organism_Function .

:MAN_174111234
        a       erm:Manifestation ;
        erm:aboutObservation
                :C0042963 ;
        erm:hasAgeContext :C0001578 ;
        erm:hasFoodContext :C0026131 ;
        erm:hasTemporalContext
                :C0231290 ;
        erm:isManifestationOf
                :EV_161514994 .

:C1278910
        a       erm:UMLS_Concept , erm:ER_Modeled_Concept , erm:Locus ,
umls:Body_Location_or_Region ;
        erm:hasConceptString
                "Entire oral cavity"^^xsd:string ;
        erm:hasMetaString "Mouth"^^xsd:string .

:MAN_1972197664
        a       erm:Manifestation ;
        erm:aboutObservation
                :C0015967 ;
        erm:isManifestationOf
```

```
              :EV_1703472089 ;
       erm:isModifiedBy :Mod_453904929 .

erm:NeurologicalSignorSymptom
       a       owl:Class ;
       rdfs:subClassOf erm:Signs_Symptoms .

:C0034977
       a       erm:Observed_Context , erm:UMLS_Concept ,
erm:ER_Modeled_Concept , umls:Pharmacologic_Substance ,
umls:Organic_Chemical , erm:Medication_Context ;
       erm:hasConceptString
              "Reglan"^^xsd:string ;
       erm:hasMetaString "Reglan"^^xsd:string .

:C0302523
       a       erm:UMLS_Concept , umls:Research_Activity .

:Mod_202233976
       a       erm:Modifier_Operation ;
       erm:hasModifier :C0205222 ;
       erm:operatesOn :C1278993 .

:C1278993
       a       erm:UMLS_Concept , umls:Body_System , erm:ER_Modeled_Concept
, erm:Locus ;
       erm:hasConceptString
              "Entire skin"^^xsd:string ;
       erm:hasMetaString "Skin"^^xsd:string .

:C0234211
       a       erm:NeurologicalSignorSymptom ;
       erm:hasConceptString
              "Pins and needles (finding)"^^xsd:string ;
       erm:hasMetaString "Tingling"^^xsd:string .

:C1267547
       a       erm:UMLS_Concept , erm:ER_Modeled_Concept , erm:Locus ,
umls:Body_Location_or_Region ;
       erm:hasConceptString
              "Entire mouth region"^^xsd:string ;
       erm:hasMetaString "Mouth"^^xsd:string .

:C0205222
       a       erm:UMLS_Concept , erm:ER_Modeled_Concept ,
erm:Functional_Modifiers , erm:Qualitative_Modifier ,
umls:Qualitative_Concept ;
       erm:hasConceptString
              "Dry"^^xsd:string ;
       erm:hasMetaString "Dry"^^xsd:string .

:MAN_1741111145
       a       erm:Manifestation ;
```

```
      erm:aboutObservation
              :C0027497 ;
      erm:hasAgeContext :C0001578 ;
      erm:hasFoodContext :C0026131 ;
      erm:hasTemporalContext
              :C0231290 ;
      erm:isManifestationOf
              :EV_161514994 .

:CC_165770160
      a       erm:ChiefComplaint ;
      erm:hasCCText "a 13 years old teenager with nausea and vomitting
after drinking bad milk. has taken Reglan that made ..."^^xsd:string ;
      erm:hasEvidence :EV_1703472089 , :EV_258386563 , :EV_1658750257 ,
:EV_161514994 , :EV_309412434 .

:EV_258386563
      a       erm:Evidence .

:Mod_2053891702
      a       erm:Modifier_Operation ;
      erm:hasModifier :C0205169 ;
      erm:operatesOn :C0026131 .

:Mod_1682377644
      a       erm:Modifier_Operation ;
      erm:hasModifier :C0205222 ;
      erm:operatesOn :C1278993 .

:EV_1703472089
      a       erm:Evidence .

:C0700325
      a       erm:Observation , erm:UMLS_Concept , erm:ER_Modeled_Concept
, erm:Healthcare_Procedure , umls:Therapeutic_or_Preventive_Procedure ;
      erm:hasConceptString
              "Patient observation"^^xsd:string ;
      erm:hasMetaString "observation"^^xsd:string .

:C0013144
      a       erm:NeurologicalSignorSymptom ;
      erm:hasConceptString
              "Drowsiness"^^xsd:string ;
      erm:hasMetaString "Drowsy"^^xsd:string .

:C0226896
      a       erm:UMLS_Concept , erm:ER_Modeled_Concept , erm:Locus ,
umls:Body_Part__Organ__or_Organ_Component ;
      erm:hasConceptString
              "Oral cavity"^^xsd:string ;
      erm:hasMetaString "Mouth"^^xsd:string .

:C0015967
```

```
      a       erm:Observation , erm:UMLS_Concept , erm:ER_Modeled_Concept
, erm:Health_Related_Finding , erm:Signs_Symptoms , umls:Sign_or_Symptom ;
      erm:hasConceptString
              "Fever"^^xsd:string ;
      erm:hasMetaString "Fever"^^xsd:string .

umls:manifestation_of
      owl:equivalentProperty
              erm:isManifestationOf ;
      owl:inverseOf erm:hasManifestation .

:C0851278
      a       erm:UMLS_Concept , erm:ER_Modeled_Concept , erm:Locus ,
umls:Body_Part__Organ__or_Organ_Component ;
      erm:hasConceptString
              "Fingers not including thumb"^^xsd:string ;
      erm:hasMetaString "Finger"^^xsd:string .

:suggestedTreatment
      a       owl:ObjectProperty .

:Mod_1301669563
      a       erm:Modifier_Operation ;
      erm:hasModifier Defs:no ;
      erm:operatesOn :C0018681 .

:EV_309412434
      a       erm:Evidence .

:MAN_614078524
      a       erm:Manifestation ;
      erm:aboutObservation
              :C0018681 ;
      erm:isManifestationOf
              :EV_1703472089 ;
      erm:isModifiedBy :Mod_1301669563 .

:C0684271
      a       erm:UMLS_Concept , umls:Organism_Function .

umls:indicates
      owl:equivalentProperty
              erm:aboutObservation .

:C0027497
      a       erm:Observation , erm:UMLS_Concept , erm:ER_Modeled_Concept
, erm:Health_Related_Finding , umls:Finding ;
      erm:hasConceptString
              "Nausea"^^xsd:string ;
      erm:hasMetaString "Nausea"^^xsd:string .

:C0001578
      a       erm:Observed_Context , umls:Age_Group , erm:Age_Context ,
```

```
erm:UMLS_Concept , erm:ER_Modeled_Concept ;
      erm:hasConceptString
              "Adolescence"^^xsd:string ;
      erm:hasMetaString "teenager"^^xsd:string .

:Mod_739632424
      a       erm:Modifier_Operation ;
      erm:hasModifier :C0205222 ;
      erm:operatesOn :C1123023 .

:C0001948
      a       erm:UMLS_Concept , umls:Individual_Behavior .

:C1281584
      a       erm:UMLS_Concept , erm:ER_Modeled_Concept , erm:Locus ,
umls:Body_Part__Organ__or_Organ_Component ;
      erm:hasConceptString
              "Entire finger"^^xsd:string ;
      erm:hasMetaString "Finger"^^xsd:string .

umls:affects
      owl:inverseOf erm:hasModifier .

:EV_161514994
      a       erm:Evidence ;
      Defs:hasTextForm "a 13 years old teenager with nausea and vomitting
after drinking bad milk"^^xsd:string .

:MAN_1077141075
      a       erm:Manifestation ;
      erm:aboutObservation
              :C0700325 ;
      erm:hasLocus :C1278993 , :C1123023 ;
      erm:isManifestationOf
              :EV_258386563 ;
      erm:isModifiedBy :Mod_739632424 , :Mod_202233976 .

:MAN_1802408159
      a       erm:Manifestation ;
      erm:hasLocus :C1267547 , :C1278910 , :C0016129 , :C0226896 ,
:C0851278 , :C1281584 ;
      erm:isManifestationOf
              :EV_309412434 .

:Mod_1178085293
      a       erm:Modifier_Operation ;
      erm:hasModifier :C0205222 ;
      erm:operatesOn :C1123023 .

:C0016129
      a       erm:UMLS_Concept , erm:ER_Modeled_Concept , erm:Locus ,
umls:Body_Part__Organ__or_Organ_Component ;
      erm:hasConceptString
```

```
            "Fingers"^^xsd:string ;
      erm:hasMetaString "Finger"^^xsd:string .

:C0151908
      a       erm:Observation , erm:UMLS_Concept , erm:ER_Modeled_Concept
, erm:Health_Related_Finding , erm:Signs_Symptoms , umls:Sign_or_Symptom ;
      erm:hasConceptString
            "Dry skin"^^xsd:string ;
      erm:hasMetaString "Dry skin"^^xsd:string .

:MAN_1736546832
      a       erm:Manifestation ;
      erm:aboutObservation
            :C0151908 ;
      erm:hasLocus :C1278993 , :C1123023 ;
      erm:isManifestationOf
            :EV_258386563 ;
      erm:isModifiedBy :Mod_1682377644 , :Mod_1178085293 .

:C0018681
      a       erm:Observation , erm:UMLS_Concept , erm:ER_Modeled_Concept
, erm:Health_Related_Finding , erm:Signs_Symptoms , umls:Sign_or_Symptom ;
      erm:hasConceptString
            "Headache"^^xsd:string ;
      erm:hasMetaString "Headache"^^xsd:string .
```